

Список доступных элементов программы

№ п/п	Наименование	Идентификатор	Коментарий
	Доступные Функции		
	Бюджетные функции		
1	Очистить таблицу бюджетов	CLEAR_BDJREPEX (КОД_БЮДЖЕТА)	Очистить данные по бюджету в таблице BdjRepEx
2	Сформировать таблицу бюджетов	FORM_BDJREPEX (КОД_БЮДЖЕТА)	Сформировать данные по бюджету в таблицу BdjRepEx
	Группы категорий		
1	Код группы категорий	GRK_CD(КодКатегории)	Код группы категорий по переданному коду категории.
2	Наименование группы категорий	GRK_NAME(КодГруппы)	Наименование группы категорий по переданному коду группы категорий.
	Данные графиков работы		
1	График-норма	GRF_NORMA(КодГрафика)	Возвращает код графика-нормы по переданному коду графика
2	Заполнение таблиц "Данные графиков работы"	GRF_TBL_MON(ДатаРП)	Заполнение таблиц "Данные графиков работы по месяцам и дням" ДатаРП - дата расчетного периода Результат: РауGrfM - таблица содержит данные графиков работы за месяц РауGrfD - таблица содержит данные графиков работы по дням
3	Количество выходных дней в интервале месяца	GRF_WEEKEND(Грф, Фаза,ДатаРП,ДеньН, ДеньК)	Количество выходных дней по графику в расчетном периоде. Без учета праздничных дней.
4	Количество праздничных дней в интервале месяца	GRF_HOLIDAY(Грф,Ф аза,ДатаРП,ДеньН, ДеньК)	Количество праздничных дней по графику в расчетном периоде.

5	Количество рабочих дней в интервале месяца	GRF_WRK_DAY(Грф, Фаза,ДатаРП,ДеньН,ДеньК,Смена)	Количество рабочих дней по графику в расчетном периоде.
6	Количество рабочих часов в интервале месяца	GRF_WRK_HRS(Грф, Фаза,ДатаРП,ДеньН,ДеньК,Смена)	Количество рабочих часов по графику в расчетном периоде.
7	Наименование графика	GRF_NAME(КодГрафика)	Возвращает наименование графика работы по переданному коду
Данные документа зарплаты			
1	Получить назначение для выплаты по ПВ	PAY_PVD_GET_NazPV(pRcd)	Функция получения текста назначение для выплаты по платежной ведомости pRcd - Rcd платежной ведомости
2	Получить назначение для выплаты по РО	PAY_PVD_GET_NazRO(pRcd)	Функция получения текста назначение для выплаты по расходному ордеру pRcd - Rcd платежной ведомости
3	Читать данные документа	DC_GET(ТипДокумента, КодДокумента-CA)	Зачитывает данные документа (поля DC_INFO...) Тип документа: 0 - Стандартный документ 1 - Стандартная платежная ведомость 2 - Стандартный список на перечисление 4 - Универсальный документ 100 - Путевой лист
Данные справочников для расчёта зарплаты			
1	Дать первую запись таблицы "Ставки рефинансирования"	STV_REF_FIRST()	Дать первую запись таблицы "Ставки рефинансирования". Функция возвращает: 1 если запись найдена, заполняет поля STV_REF... 0 если запись не найдена, очищает поля STV_REF...

2	Дать следующую запись таблицы "Ставки рефинансиров	STV_REF_NEXT()	Дать следующую запись таблицы "Ставки рефинансирования". Функция возвращает: 1 если запись найдена, заполняет поля STV_REF... 0 если запись не найдена, очищает поля STV_REF...
3	Размер минимальной зарплаты на дату	SUM_MIN_ZP (Дата)	Размер минимальной зарплаты на указанную дату.
4	Размер прожиточного минимума на дату	SUM_PMIN(Дата)	Размер прожиточного минимума на указанную дату.
5	Размер прожиточного минимума на дату от возраста	SUM_PMIN_AGE(Дата, Возраст)	Размер прожиточного минимума на указанную дату. Возраст - количество лет.
6	Ставка рефинансирования	STV_REF(Дата)	Ставка рефинансирования на дату. Заполняются поля STV_REF...
7	Тариф по коду тарифной сетки и разряду	STS_RAZ_TAR(Сетка, Разряд)	Размер тарифа по коду тарифной сетки и разряду
8	Тариф по коду тарифной сетки и разряду на дату	STS_RAZ_TAR_D(Сетка,Разряд,Дата)	Размер тарифа по коду тарифной сетки и разряду на указанную дату
Дополнительные свойства видов оплат			
1	Код преамбулы для вида оплаты	KDR_VO_CD_PAM("Код вида оплаты")	Возвращает код преамбулы определенной для вида оплаты по переданному коду вида оплаты. Полученный код используется для получения наименования или текста преамбулы из справочника преамбул. См.раздел "Справочник преамбул".

2	Комментарий к виду оплаты	KDR_VO_COMM("Код вида оплаты")	Возвращает комментарий к виду оплаты по переданному коду вида оплаты. Комментарий может использоваться как альтернативное наименование вида оплаты для формирования текста приказа.
Доступ к данным Лицевого счета работника			
1	Получить дополнительный адрес работника	LS_GET_ADDR(Код_адреса)	Читает дополнительный адрес работника. Заполняет поля LS_ADA_PLC_... Параметры: Код_адреса: Код дополнительного адреса В случае успеха функция возвращает 1. Если запись лицевого счета работника не прочитана, или для работника не определен дополнительный адрес, функция возвращает 0.
2	Получить расширение файла фотографии работника	LS_GET_EXT_FILE_FOTO()	Возвращает расширение файла фотографии работника. Если фотографии нет или тип неопределен возвращает пусто.
3	Получить расширение файла фотографии работника	LS_CRD_BALL(Дата 0)	Средний балл студента на указанную дату

4	Получить стаж работника на дату	LS_GET_STG(Тип_Стажа, Код_Стажа, Дата, Дата_Начала, Дней, Месяцев, Лет)	<p>Возвращает данные стажа работника на дату.</p> <p>Параметры:</p> <p>Тип_стажа: 0-основной,1-дополнительный</p> <p>Код_стажа: Для основных стажей:</p> <ul style="list-style-type: none"> - 0-общий - 1-непрерывный(RU)/страховой(UA) - 2-отраслевой - 3-в организации - 4-госслужащего - 5-непрерывный госслужащего - 6-страховой <p>Для дополнительных стажей:</p> <ul style="list-style-type: none"> - Код стажа <p>Дата : Дата, на которую рассчитывается стаж</p> <p>Дата_начала: Дата нача</p>
5	Сохранить фотографию работника в файл	LS_FILE_FOTO("Имя_файла")	<p>Сохраняет фотографию работника во временный (TMP) каталог БЭСТ-ПРО с именем "Имя_файла" и расширением, соответствующим формату изображения.</p> <p>Возвращает 1 при успешном сохранении, иначе - 0.</p> <p>Применяется для отображения фотографии в Win-отчете.</p>
6	Читать интервал дополнительного стажа по коду	LS_AST_FIRSTCD(Код стажа)	<p>Читает 1-ю запись интервалов дополнительных стажей работника по коду стажа.</p> <p>Заполняет поля LS_AST_...</p> <p>Если запись прочитана, возвращает 1, иначе 0</p>

7	Читать наименования записи списка приказов	LS_PRKZ_NM()	Читает наименования записи списка приказов работника. Заполняет поля LS_PRK_... Если наименования прочитаны, возвращает 1, иначе 0
8	Читать наименования записи списка удержаний	LS_UDR_NM()	Читает наименования записи списка удержаний работника. Заполняет поля LS_UDR_...
9	Читать первое дополнительное удостоверение	LS_APS_FIRST()	Читает 1-ю запись списка дополнительных удостоверений работника. Заполняет поля LS_APS_... Если запись прочитана, возвращает 1, иначе 0
10	Читать первую запись списка льгот	LS_LG_FIRST()	Читает 1-ю запись списка льгот работника. Заполняет поля LS_PDX_LG_... Если запись прочитана, возвращает 1, иначе 0
11	Читать первую запись списка начислений	LS_NCH_FIRST()	Читает 1-ю запись списка начислений работника. Заполняет поля LS_NCH_... Если запись прочитана, возвращает 1, иначе 0
12	Читать первую запись списка приказов	LS_PRKZ_FIRST()	Читает 1-ю запись списка приказов работника. Заполняет поля LS_PRK_... Если запись прочитана, возвращает 1, иначе 0
13	Читать первую запись списка приказов (коды)	LS_PRKZ_CD_FST()	Читает 1-ю запись списка приказов работника. Заполняет только кодовые поля LS_PRK_... Если запись прочитана, возвращает 1, иначе 0

14	Читать первую запись списка удержаний	LS_UDR_FIRST()	Читает 1-ю запись списка удержаний работника. Заполняет поля LS_UDR_... Если запись прочитана, возвращает 1, иначе 0
15	Читать первую запись трудовой деятельности	LS_TD_FIRST()	Читает 1-ю запись трудовой деятельности работника. Заполняет поля LS_TD_... Если запись прочитана, возвращает 1, иначе 0
16	Читать первую запись устаревшего списка льгот	LS_LG_FIRSTOLD()	Читает 1-ю запись устаревшего списка льгот работника. Заполняет поля LS_PDX_LG_... Если запись прочитана, возвращает 1, иначе 0
17	Читать первую строку комментария	LS_CMT_FIRST(Строка)	Читает 1-ю строку комментария работника. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0
18	Читать первый интервал дополнительного стажа	LS_AST_FIRST()	Читает 1-ю запись интервалов дополнительных стажей работника. Заполняет поля LS_AST_... Если запись прочитана, возвращает 1, иначе 0
19	Читать следующее дополнительное удостоверение	LS_APS_NEXT()	Читает следующую запись списка дополнительных удостоверений работника. Заполняет поля LS_APS_... Если запись прочитана, возвращает 1, иначе 0
20	Читать следующий интервал дополнительного стажа	LS_AST_NEXT()	Читает следующую запись интервалов дополнительных стажей работника. Заполняет поля LS_AST_... Если запись прочитана, возвращает 1, иначе 0

21	Читать следующую запись списка льгот	LS_LG_NEXT()	Читает следующую запись списка льгот работника. Заполняет поля LS_PDX_LG_... Если запись прочитана, возвращает 1, иначе 0
22	Читать следующую запись списка начислений	LS_NCH_NEXT()	Читает следующую запись списка начислений работника. Заполняет поля LS_NCH_... Если запись прочитана, возвращает 1, иначе 0
23	Читать следующую запись списка приказов	LS_PRKZ_NEXT()	Читает следующую запись списка приказов работника. Заполняет поля LS_PRK_... Если запись прочитана, возвращает 1, иначе 0
24	Читать следующую запись списка приказов (коды)	LS_PRKZ_CD_NXT()	Читает следующую запись списка приказов работника. Заполняет только кодовые поля LS_PRK_... Если запись прочитана, возвращает 1, иначе 0
25	Читать следующую запись списка удержаний	LS_UDR_NEXT()	Читает следующую запись списка удержаний работника. Заполняет поля LS_UDR_... Если запись прочитана, возвращает 1, иначе 0
26	Читать следующую запись трудовой деятельности	LS_TD_NEXT()	Читает следующую запись трудовой деятельности работника. Заполняет поля LS_TD_... Если запись прочитана, возвращает 1, иначе 0
27	Читать следующую запись устаревшего списка льгот	LS_LG_NEXTOLD()	Читает следующую запись устаревшего списка льгот работника. Заполняет поля LS_PDX_LG_... Если запись прочитана, возвращает 1, иначе 0

28	Читать следующую строку комментария	LS_CMT_NEXT(Строка)	Читает следующую строку комментария работника. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0
29	Читать табельный номер первой записи совместителя	LS_SVM_FIRST()	Читает табельный номер первой записи совместителя Если запись прочитана, возвращает табельный номер внутреннего совместителя, иначе 0
30	Читать табельный номер следующей записи совместителя	LS_SVM_NEXT()	Читает табельный номер следующей записи внутреннего совместителя Если запись прочитана, возвращает табельный номер внутреннего совместителя, иначе 0
Доступ к Лицевым счетам работников			
1	Код справочника процентов начислений	LS_SPNCd(LS_KPU_RCD, дата)	Код справочника процентов начислений на указанную дату Определяется по штатной позиции Штатного расписания LS_KPU_RCD - ID записи картотеки работников дата - дата на которую определяется код
2	Проверка "чужого" Лицевого счета	LS_ALIEN()	Проверяет текущую запись Реестра работников на предмет возможности работы с ней. Код возврата: 0 - с записью можно работать, 1 - не все поля заполнены, 2 - запись удалена логически 3 - не установлена текущая запись Реестра работников

3	Проверка отмеченного Лицевого счета	LS_MARKED()	Проверяет текущую запись Реестра работников, установлена ли для нее отметка. Код возврата: 0 - не установлена, 1 - установлена Может использоваться только в Реестре работников.
4	Проверка прав доступа пользователя к лиц. счёту	LS_ACCESS()	Проверка доступности лицевого счета согласно прав доступа пользователя 1 - запись доступна 0 - запись не доступна
5	Установить сортировку Реестра работников	LS_SET_SORT(Порядок_сортировки)	Устанавливает порядок сортировки Реестра работников. Функции LS_FIRST/LS_NEXT работают по установленной сортировке: 0-Без сортировки, 1-Таб.номер, 2-Фамилия, 3-Подразделение,Таб.номер, 4-Подразделение,Фамилия
6	Читать запись Реестра работников	LS_BY_TN(Табельный_номер)	Читает запись Реестра работников по табельному номеру. Заполняет поля LS_... Если запись прочитана, возвращает 1, иначе 0
7	Читать наименования Лицевого счета	LS_NM()	Читает наименования Лицевого счета работника. Заполняет поля LS_...
8	Читать первую запись Реестра работников	LS_FIRST()	Читает 1-ю запись Реестра работников. Заполняет поля LS_... Если запись прочитана, возвращает 1, иначе 0

9	Читать первую отмеченную запись Реестра	LS_FIRST_MARK()	Читает 1-ю отмеченную запись Реестра работников. Заполняет поля LS_... Если запись прочитана, возвращает 1, иначе 0 Может использоваться только в Реестре работников.
10	Читать следующую запись Реестра работников	LS_NEXT()	Читает следующую запись Реестра работников. Заполняет поля LS_... Если запись прочитана, возвращает 1, иначе 0
11	Читать следующую отмеченную запись Реестра	LS_NEXT_MARK()	Читает следующую отмеченную запись Реестра работников. Заполняет поля LS_... Если запись прочитана, возвращает 1, иначе 0 Может использоваться только в Реестре работников.
Доступ к Расчетным листам работника			
1	Читать данные табеля записи Расчетного листа	RL_TB()	Читает данные поденного табеля записи расчетного листа. Запись должна быть прочитана функцией RL_RP_FIRST/RL_UP_FIRST/RL_NEXT Заполняет поля VO_TB_MSKO, VO_TB_MSKZ, VO_TB_xx_HRS Если запись не была прочитана, возвращает 0, иначе 1
2	Читать наименования записи Расчетного листа	RL_NM()	Читает наименования записи расчетного листа. Запись должна быть прочитана функцией RL_RP_FIRST/RL_UP_FIRST/RL_NEXT Заполняет поля VO_... Если запись не была прочитана, возвращает 0, иначе 1

3	Читать наименования ШПЗ записи Расчетного листа	RL_SPZ_NM()	<p>Читает наименования записи ШПЗ расчетного листа. Запись должна быть прочитана функцией RL_SPZ_FIRST/RL_SPZ_NEXT Заполняет поля VO_...</p> <p>Если запись не прочитана, возвращает 0, иначе 1</p>
4	Читать первую запись данных расчета по среднему	RL_SRD_FIRST()	<p>Читает 1-ю запись данных для расчета по среднему. Заполняет поля VO_SRD_...</p> <p>Если запись прочитана, возвращает 1, иначе 0. Может применяться в программах расчета средних.</p>
5	Читать первую запись Расчетного периода	RL_RP_FIRST(Расчетный_период 0)	<p>Читает 1-ю запись указанного расчетного периода (0-первого имеющегося расчетного периода) Заполняет поля VO_... Может перейти на следующий период. Если запись прочитана, возвращает 1, иначе 0</p>
6	Читать первую запись Учетного периода	RL_UP_FIRST(Учетный_период 0)	<p>Читает 1-ю запись указанного учетного периода (0-первого имеющегося учетного периода) Заполняет поля VO_... Может перейти на следующий период. Если запись прочитана, возвращает 1, иначе 0</p>
7	Читать первую запись ШПЗ	RL_SPZ_FIRST()	<p>Читает 1-ю запись шифра производственных затрат. Заполняет поля VO_SCH, VO_ZAK, VO_PDRF, VO_FK_SM Если запись прочитана, возвращает 1, иначе 0</p>

8	Читать следующую запись	RL_NEXT()	<p>Читает следующую запись расчетного или учетного периода (в зависимости от вызова RL_GET_FIRST...) Заполняет поля VO_... Может перейти на следующий период. Если запись прочитана, возвращает 1, иначе 0</p>
9	Читать следующую запись данных расчета по среднему	RL_SRD_NEXT()	<p>Читает следующую запись данных для расчета по среднему. Заполняет поля VO_SRD_... Если запись прочитана, возвращает 1, иначе 0. Может применяться в программах расчета средних.</p>
10	Читать следующую запись ШПЗ	RL_SPZ_NEXT()	<p>Читает следующую запись шифра производственных затрат. Заполняет поля VO_SCH, VO_ZAK, VO_PDRF, VO_FK_SM Если запись прочитана, возвращает 1, иначе 0</p>
Доступ к Списку работников			
1	Найти запись Списка работников по ID	WRK_BY_RCD(ID работника)	<p>Читает запись Списка работников по ID. Заполняет поля WRK_... Если запись прочитана, возвращает 1, иначе - 0</p>
2	Найти запись Списка работников по Таб.номеру	WRK_BY_TN(Табельный_номер)	<p>Читает запись Списка работников по табельному номеру. Заполняет поля WRK_... Если запись прочитана, возвращает 1, иначе - 0</p>

3	Установить сортировку Списка работников	WRK_SET_SORT(Порядок_сортировки)	Устанавливает порядок сортировки Списка работников. Функции WRK_FIRST/WRK_NEXT работают по установленной сортировке: 0-Без сортировки, 1-Таб.номер, 2-Фамилия, 3-Подразделение,Таб.номер, 4-Подразделение,Фамилия
4	Читать первую запись Списка работников	WRK_FIRST()	Читает 1-ю запись Списка работников. Заполняет поля WRK_... Если запись прочитана, возвращает 1, иначе - 0
5	Читать следующую запись Списка работников	WRK_NEXT()	Читает следующую запись Списка работников. Заполняет поля WRK_... Если запись прочитана, возвращает 1, иначе - 0
Доступ к Справочнику контрагентов			
1	Обновить реквизиты контрагента из ГНАУ	PTN_UPDRKZ_GNAU ("ОКПО", "ИНН")	Обновление реквизитов контрагента из ГНАУ по ОКПО или ИНН
2	Чтение записи контрагента по внутреннему коду	PTN_READ_INT(Внутренний_код_контрагента)	Читает запись Справочника контрагентов по внутреннему коду ("собственному адресу"). Заполняет поля раздела "Реквизиты контрагента". Если запись прочитана, возвращает 1, иначе 0
3	Чтение записи контрагента по коду	PTN_READ_CD(Код_контрагента)	Читает запись Справочника контрагентов по коду. Заполняет поля раздела "Реквизиты контрагента". Если запись прочитана, возвращает 1, иначе 0

4	Чтение информации о группе по внутреннему коду	PTN_GRP_RD_INT(Внутренний_Код_группы)	<p>Читает запись из базы групп контрагентов по внутреннему (Rcd) коду.</p> <p>Заполняет поля</p> <p>PtnGr_CdPath - объединенный код групп</p> <p>PtnGr_Cd01, PtnGr_Nm01 - код, название группы 1-го уровня</p> <p>PtnGr_Cd02, PtnGr_Nm02 - код, название группы 2-го уровня</p> <p>PtnGr_Cd03, PtnGr_Nm03 - код, название группы 3-го уровня</p> <p>PtnGr_Cd04, PtnGr_Nm04 - код, название группы 4-го уровня</p> <p>PtnGr_Cd05, PtnGr_Nm05 - код, название группы 5-го уровня</p> <p>PtnGr_Cd06, PtnGr_Nm06 - код, название группы 6-го уровня</p> <p>PtnGr_Cd</p>
---	--	---	---

5	Чтение информации о группе по коду	PTN_GRP_RD_CD(Код_группы)	<p>Читает запись из базы групп контрагентов по коду.</p> <p>Заполняет поля</p> <p>PtnGr_CdPath - объединенный код групп</p> <p>PtnGr_Cd01, PtnGr_Nm01 - код, название группы 1-го уровня</p> <p>PtnGr_Cd02, PtnGr_Nm02 - код, название группы 2-го уровня</p> <p>PtnGr_Cd03, PtnGr_Nm03 - код, название группы 3-го уровня</p> <p>PtnGr_Cd04, PtnGr_Nm04 - код, название группы 4-го уровня</p> <p>PtnGr_Cd05, PtnGr_Nm05 - код, название группы 5-го уровня</p> <p>PtnGr_Cd06, PtnGr_Nm06 - код, название группы 6-го уровня</p> <p>PtnGr_Cd07, PtnGr_Nm07 - к</p>
6	Чтение кода "родительской" группы	PTN_GRP_PAR (Код-группы)	<p>Возвращает код родительской группы, для группы с кодом "Код-группы".</p> <p>Если группа с кодом "Код-группы" не найдена, или нет родительской группы, то возвращает пустую строку.</p>
7	Чтение первой записи списка адресов контрагента	PTN_ADR_FIRST()	<p>Читает 1-ю запись списка адресов контрагента.</p> <p>Заполняет поля FilR_...</p> <p>Если запись прочитана, возвращает 1, иначе 0</p>
8	Чтение первой записи списка счетов контрагента	PTN_SCH_FIRST(КОД_КОНТРАГЕНТА)	<p>Читает 1-ю запись списка счетов контрагента.</p> <p>Заполняет поля PtnR_Sch...,PtnRBnk_...</p> <p>Если запись прочитана, возвращает 1, иначе 0</p>

9	Чтение первой строки истории контрагента	PTN_HIST_FIRST()	Читает первую запись истории контрагента. Заполняет поля PtnRHst_... Если запись прочитана возвращает 1, иначе 0
10	Чтение первой строки комментария контрагента	PTN_CMT_FIRST(Строка)	Читает 1-ю строку комментария контрагента. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0
11	Чтение следующей записи адресов котрагента	PTN_ADR_NEXT ()	Читает следующую запись списка адресов контрагента. Заполняет поля FilR_... Если запись прочитана, возвращает 1, иначе 0
12	Чтение следующей записи счетов котрагента	PTN_SCH_NEXT ()	Читает следующую запись списка счетов контрагента. Заполняет поля PtnR_Sch...,PtnRBnk_... Если запись прочитана, возвращает 1, иначе 0
13	Чтение следующей строки истории контрагента	PTN_HIST_NEXT()	Читает следующую запись истории контрагента. Заполняет поля PtnRHst_... Если запись прочитана возвращает 1, иначе 0
14	Чтение следующей строки комментария контрагента	PTN_CMT_NEXT(Строка)	Читает следующую строку комментария контрагента. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0
15	Чтение тарифной ставки контрагента	PTN_READ_TRF ("Код контр",Дата,"Шифр")	Функция возвращает тариф контрагента на дату по заданному шифру
Доступ к справочнику ОТН			
1	Найти запись справочника ОТН по rcd операции	OTN_GET_RCD(rcd_операции)	Читает запись справочника ОТН по rcd операции. Заполняет поля OTN_... Если запись прочитана, возвращает 1, иначе - 0

2	Найти запись справочника ОТН по коду операции	OTN_GET_CD("Код операции")	Читает запись справочника ОТН по коду операции. Заполняет поля OTN_... Если запись прочитана, возвращает 1, иначе - 0
Доступ к Справочнику подразделений			
1	Вернуть код организационной единицы	PDR_ORG_U_CD("Код подразделения", Дата)	Возвращает код организационной единицы Дата: 0-текущая дата
2	Вернуть код родительского подразделения	PDR_GET_PARENT("Код подразделения", Дата)	Возвращает код родительского подразделения. Дата: 0-текущая дата
3	Вернуть код типа организационной единицы	PDR_ORG_U_NMCD ("Код подразделения", Дата)	Возвращает код типа организационной единицы Дата: 0-текущая дата
4	Вернуть наименование организационной единицы	PDR_ORG_UNITNM("Код подразделения", Дата)	Возвращает наименование организационной единицы Дата: 0-текущая дата
5	Вернуть наименование подразделения	PDR_GET("Код подразделения", Дата)	Возвращает наименование подразделения. Дата: 0-текущая дата
6	Вернуть направление деятельности	PDR_ACTIVITY("Код подразделения", Дата)	Возвращает направление деятельности. Дата: 0-текущая дата
7	Вернуть номер уровня подразделения по коду	PDR_GET_LEVEL("Код подразделения", Дата)	Возвращает номер уровня подразделения по коду Дата: 0-текущая дата
8	Вернуть полное наименование подразделения	PDR_GET_FULL("Код подразделения", Падеж, Дата)	Возвращает полное наименование подразделения с указанием падежа. Падеж: 0-именительный,1-родительный,2-дательный Дата: 0-текущая дата
9	Вернуть признак структурной единицы подразделения	PDR_GET_ISSTE("Код подразделения", Дата)	Возвращает признак структурной единицы подразделения по коду(0/1) Дата: 0-текущая дата

10	Вернуть тип организационной единицы	PDR_ORG_UNIT("Код подразделения", Падеж, Дата)	Возвращает тип организационной единицы с указанием падежа. Падеж: 0-именительный,1-родительный,2-дательный Дата: 0-текущая дата
11	Вернуть тип подразделения по коду	PDR_GET_TYPE("Код подразделения", Дата)	Возвращает тип подразделения по коду Дата: 0-текущая дата
12	Диалог выбора подразделения с фильтром доступа	PDR_SELDLG_FLT("Имя поля кода", "Имя таблицы доступа", Множественный_выбор, Отме	Отобразить диалог выбора подразделения с фильтром доступа ВХОД : Имя поля кода - имя поля кода подразделения в таблице доступа Имя таблицы доступа - имя таблицы с доступными подразделениями(NULL - все доступно) Множественный_выбор - признак отметки нескольких записей.Если 0 - то только одной Отмечать_подчиненные - признак отметки подчиненных подразделений, при отметке родителя ВОЗВРАТ: Код выбранного подразделения(подразделений через запятую), или NULL при
13	Зачитать данные по подразделению(заполн.реквизиты)	READ_SPRPDR("Код_подразделения", Дата)	Зачитать данные по подразделению, заполняет реквизиты подразделения ВОЗВРАТ: 0-подразделение отсутствует/1-подразделение найдено

14	Получение кода подразделения по структурной единице	PDR_GET_BY_STE("Код структ.ед", Дата)	Получить код подразделения(сформированного на основании СТЕ) по коду СТЕ Дата: 0-текущая дата Возврат : код подразделения(сформированный на основании СТЕ)
Износ ОС			
1	Баланс.стоим. группы ОНУ на начало периода	BSTM_ONU_DAT(Дата периода,Код группы)	Возвращает для группы ОНУ, заданной кодом, сумму балансовой стоимости на начало периода. Период определяется датой начала.
2	Дата достиж.налог.бал.стоимостью суммы	STM_SM_DAT_N(Сумма)	Возвращает дату наступления налоговой балансовой стоимостью указанной суммы.
3	Дата достижения бал.стоимостью суммы	STM_SM_DAT(Сумма)	Возвращает дату наступления балансовой стоимостью указанной суммы.
4	Начисл. по предприятию износ за период	CLC_SM_VA(Дата)	Начисленный по предприятию износ за период
5	Начисленная НА за период	CLC_SMN(Дата)	Начисленная налоговая амортизация за период, указанный датой.
6	Начисленная НА между датами	CLC_SMN_PER(Дата_1, Дата_2)	Начисленная по карточке ОС налоговая амортизация между датами Дата_1 и Дата_2
7	Начисленная НА на дату	CLC_SMN_ALL(Дата)	Начисленная налоговая амортизация на дату
8	Начисленная НА с начала года	CLC_SMN_YEAR(Дата)	Начисленная по карточке ОС налоговая амортизация с начала года
9	Начисленная управл.аморизация за период	CLC_SMU(Дата)	Начисленная управленческая амортизация за период, указанный датой.

10	Начисленная управл.амортизация между датами	CLC_SMU_PER(Дата_1, Дата_2)	Начисленная по карточке ОС управленческая амортизация между датами Дата_1 и Дата_2
11	Начисленная управл.амортизация на дату	CLC_SMU_ALL(Дата)	Начисленная управленческая амортизация на дату
12	Начисленная управл.амортизация с начала года	CLC_SMU_YEAR(Дата)	Начисленная по карточке ОС управленческая амортизация с начала года
13	Начисленный износ за период	CLC_SM(Дата)	Начисленный износ за период, в который входит указанная дата.
14	Начисленный износ между датами	CLC_SM_PER(Дата_1, Дата_2)	Начисленный по карточке ОС износ между датами Дата_1 и Дата_2
15	Начисленный износ на дату	CLC_SM_ALL(Дата)	Начисленный по карточке ОС износ на дату
16	Начисленный износ с начала года	CLC_SM_YEAR(Дата)	Начисленный по карточке ОС износ с начала года
17	Пробег за период	CLC_PRB(Дата)	Пробег по карточке ОС за период, указанный датой
18	Пробег на дату	CLC_PRB_ALL(Дата)	Пробег по карточке ОС всего на дату
19	Сумма начисл. НА по группе ОНУ за период	IZNN_ONU_PER(Дата начала,Дата окончания,Код группы)	Возвращает для группы ОНУ, заданной кодом, сумму налоговой амортизации, начисленной за период. Период определяется датами начала и окончания.
	Интервалы изменения Лицевого счета		
1	Наименования текущего интервала	LS_GET_NM()	Заполняет поля наименований LS_* значениями последнего зачитанного интервала изменения основных сведений Лицевого счета

2	Основные сведения на дату	LS_GET_ON_DATE(Дата)	Заполняет поля LS_* значениями интервала изменения основных сведений Лицевого счета соответствующего указанной дате с точки зрения текущего учётного периода.
3	Основные сведения на дату (с указанием УП)	LS_GET_ON_D_UP(Дата, ДатаУП)	Дата - дата ДатаУП - учётный период Заполняет поля LS_* значениями интервала изменения основных сведений Лицевого счета соответствующего указанной дате с точки зрения указанного учётного периода.
4	Первый интервал	LS_GET_FIRST(ДатаРП)	Заполняет поля LS_* значениями первого интервала изменения основных сведений Лицевого счета для указанного расчетного периода с точки зрения текущего учётного периода.
5	Первый интервал (с точки зрения указанного УП)	LS_GET_FST_UP(ДатаРП, ДатаУП)	ДатаРП - расчётный период ДатаУП - учётный период Заполняет поля LS_* значениями первого интервала изменения основных сведений Лицевого счета для указанного расчетного периода с точки зрения указанного учётного периода.
6	Последний интервал	LS_GET_LAST(ДатаРП)	Заполняет поля LS_* значениями последнего интервала изменения основных сведений Лицевого счета для указанного расчетного периода с точки зрения текущего учётного периода.

7	Последний интервал (с точки зрения указанного УП)	LS_GET_LAST_UP(ДатаРП, ДатаУП)	ДатаРП - расчётный период ДатаУП - учётный период Заполняет поля LS_* значениями последнего интервала изменения основных сведений Лицевого счета для указанного расчетного периода с точки зрения указанного учётного периода.
8	Предыдущий интервал	LS_GET_PRIOR()	Заполняет поля LS_* значениями предыдущего интервала изменения основных сведений Лицевого счета в пределах расчетного периода
9	Следующий интервал	LS_GET_NEXT()	Заполняет поля LS_* значениями следующего интервала изменения основных сведений Лицевого счета в пределах расчетного периода
	Исходные суммы для расчёта зарплаты		
1	Вхождение вида оплаты в исходную сумму	VO_PL_SM_VO("КодВО1", "КодВО2", Дата)	Вхождение вида оплаты 2 в исходную сумму вида оплаты 1 на указанную дату.
2	Вхождение метода расчета в исходную сумму	VO_PL_SM_MET("КодВО", КодМР, Дата)	Вхождение метода расчета (КодМР) в исходную сумму вида оплаты (КодВО) на указанную дату.
3	План.сумма в текущ.расч.периоде по списку ВО	SUM_PL_VO("СписокВО" КодСпискаВО)	Плановая сумма в текущем Расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.

4	План.сумма в указ.расч.периоде по списку ВО	SUM_PL_RP_VO(ДатаРП,"СписокВО" КодСпискаВО)	Плановая сумма в указанном Расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
5	План.сумма на указ.дату по списку видов оплат	SUM_PL_DAT(Дата,"СписокВО" КодСпискаВО)	Плановая сумма на указанную дату по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
6	Факт.сумма в расч.периоде по списку ВО и интервале	SUM_FK_RP_VO_I(ДатаРП,"СписокВО" КодСпискаВО,ДатаН,ДатаК)	Фактическая сумма в указанном расчетном периоде по списку видов оплат и в интервале дат Список видов оплат содержит коды видов оплат разделенные пробелом. ДатаН - дата начала или 0 ДатаК - дата конца или 0
7	Факт.сумма в тек.УП по ВО внутр.совместителя	SUM_FK_VO_SVM("Список видов оплат" КодСпискаВО)	Факт.сумма в тек.учетном периоде по списку видов оплат внутр.совместителя. Список видов оплат содержит коды видов оплат разделенные пробелом.
8	Факт.сумма в текущ.расч.периоде по списку ВО	SUM_FK_VO_R("СписокВО" КодСпискаВО)	Фактическая сумма в текущем расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
9	Факт.сумма в текущ.учетн.периоде по списку ВО	SUM_FK_VO("СписокВО" КодСпискаВО)	Фактическая сумма в текущем Учетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.

10	Факт.сумма в указ.расч.периоде по списку ВО	SUM_FK_RP_VO_R (ДатаРП, "СписокВО" КодСпискаВО)	Фактическая сумма в указанном расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
11	Факт.сумма в указ.учетн.периоде по списку ВО	SUM_FK_UP_VO(ДатаУП,"СписокВО" КодСпискаВО)	Фактическая сумма в указанном Учетном периоде по Списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
12	Факт.сумма по РП и списку ВО + записи совместителя	SUM_FK_RP_VO_S (ДатаРП, "СписокВО" КодСпискаВО,ДатаН,ДатаК)	Фактическая сумма в указанном расчетном периоде по списку видов оплат и в интервале дат + суммы записей совместителя Список видов оплат содержит коды видов оплат разделенные пробелом. ДатаН - дата начала или 0 ДатаК - дата конца или 0
Итоги Расчетного листа			
1	Начислено в текущем учетном периоде	SRL_NCH ()	Сумма начислений в текущем учетном периоде.
2	Начислено в указанном учетном периоде	SRL_NCH_UP (ДатаУП)	Сумма начислений в указанном учетном периоде.

3	Начислено по указанному ИФ и ЭКР	SRL_NCH_SE (ДатаУП,ИФ,ЭКР)	<p>Сумма начислений по:</p> <ul style="list-style-type: none"> - учетному периоду "ДатаУП" - источнику финансирования "ИФ" - экономическому классификатору расходов "ЭКР" <p>Если переменные "ИФ" или "ЭКР" имеют пустое значение (""), то выбираются записи с неопределенным ИФ или ЭКР. Если переменные "ИФ" или "ЭКР" имеют значение "Все", то выбираются записи с любым ИФ или ЭКР.</p>
4	Сальдо входящее по указанному ИФ и ЭКР	SRL_SLD_INP_SE (ДатаУП,ИФ,ЭКР)	<p>Сальдо входящее по:</p> <ul style="list-style-type: none"> - учетному периоду "ДатаУП" - источнику финансирования "ИФ" - экономическому классификатору расходов "ЭКР" <p>Если переменные "ИФ" или "ЭКР" имеют пустое значение (""), то выбираются записи с неопределенным ИФ или ЭКР. Если переменные "ИФ" или "ЭКР" имеют значение "Все", то выбираются записи с любым ИФ или ЭКР.</p>
5	Сальдо входящее текущего учетного периода	SRL_SLD_INP ()	Сумма входящего сальдо текущего учетного периода.
6	Сальдо входящее указанного учетного периода	SRL_SLD_INP_UP (ДатаУП)	Сумма входящего сальдо указанного учетного периода.

7	Сальдо исходящее по указанному ИФ и ЭКР	SRL_SLD_OUT_SE (ДатаУП,ИФ,ЭКР)	Сальдо исходящее по: - учетному периоду "ДатаУП" - источнику финансирования "ИФ" - экономическому классификатору расходов "ЭКР" Если переменные "ИФ" или "ЭКР" имеют пустое значение (""), то выбираются записи с неопределенным ИФ или ЭКР. Если переменные "ИФ" или "ЭКР" имеют значение "Все", то выбираются записи с любым ИФ или ЭКР.
8	Сальдо исходящее текущего учетного периода	SRL_SLD_OUT ()	Сальдо исходящее текущего учетного периода.
9	Сальдо исходящее указанного учетного периода	SRL_SLD_OUT_UP (ДатаУП)	Сальдо исходящее указанного учетного периода.
10	Удержано по указанному ИФ и ЭКР	SRL_UDR_SE (ДатаУП,ИФ,ЭКР)	Сумма удержаний по: - учетному периоду "ДатаУП" - источнику финансирования "ИФ" - экономическому классификатору расходов "ЭКР" Если переменные "ИФ" или "ЭКР" имеют пустое значение (""), то выбираются записи с неопределенным ИФ или ЭКР. Если переменные "ИФ" или "ЭКР" имеют значение "Все", то выбираются записи с любым ИФ или ЭКР.
11	Удержание в Пенс.фонд 1% в текущем учетном периоде	SRL_PF_1P ()	Сумма удержания в Пенсионный фонд 1% в текущем учетном периоде.

12	Удержание в Пенс.фонд 1% в указанном учетн.периоде	SRL_PF_1P_UP (ДатаУП)	Сумма удержания в Пенсионный фонд 1% в указанном учетном периоде.
13	Удержание в Пенс.фонд 2% в текущем учетном периоде	SRL_PF_2P ()	Сумма удержания в Пенсионный фонд 2% в текущем учетном периоде.
14	Удержание в Пенс.фонд 2% в указанном учетн.периоде	SRL_PF_2P_UP (ДатаУП)	Сумма удержания в Пенсионный фонд 2% в указанном учетном периоде.
15	Удержано в текущем учетном периоде	SRL_UDR ()	Сумма удержаний в текущем учетном периоде.
16	Удержано в указанном учетном периоде	SRL_UDR_UP (ДатаУП)	Сумма удержаний в указанном учетном периоде.
	Навигация по аналитике документа		
1	Добавить сумму по аналитической карточке	FANL_SET_SM(ИД карточки, "Сумма")	Добавить (изменить) сумму по аналитической карточке
2	Получить первую запись карточки	FANL_FIRST()	Получить первую запись аналитической карточки Возвращает ID записи
3	Получить следующую запись карточки	FANL_NEXT()	Получить следующую запись аналитической карточки Возвращает ID записи
4	Получить сумму карточки	FANL_GET_SM(ID карточки)	Получить сумму аналитической карточки
5	Удалить текущую запись по карточке	FANL_CUR_DEL()	Удаляет текущую карточку, установленную FANL_GET_FIRST/FANL_GET_NEXT. Возвращает Ид следующей карточки или 0.
	Навигация по шифрам банковских документов		
1	Функция добавления суммы по шифру	FS_BNKSETSUM("Тип шифра", "Код шифра", Сумма)	Добавить сумму по шифру. Тип шифра: 0 - Поступление, 1 - Платеж

2	Функция получения первого шифра	FS_BNKFIRST()	Функция возвращает первый шифр банковского документа
3	Функция получения следующего шифра	FS_BNKNEXT()	Функция возвращает следующий шифр банковского документа
4	Функция получения суммы по текущему шифру	FS_BNKGETSUM()	Функция возвращает сумму текущего шифра банковского документа
5	Функция получения суммы по шифру	FS_BNKCODE("Шифр")	Функция возвращает сумму по шифру банковского документа
6	Функция удаления разноски по шифрам	FS_BNKDELSH("Код шифра")	Удалить разноску по шифрам ПП. Если "Код шифра" пустой "", то удалит все шифры
Навигация по шифрам денежных операций			
1	Получить первый шифр	FS_SHFIRST()	Функция возвращает код первого шифра для текущего платежа.
2	Получить следующий шифр	FS_SHNEXT()	Функция возвращает следующий код шифра для текущего платежа.
3	Получить сумму по шифру	FS_SHSUM()	Функция возвращает сумму по текущему шифру
4	Функции получения первого шифра	FS_EXPFIRST()	Функция возвращает первый код шифра платежа/поступления.
5	Функции получения следующего шифра	FS_EXPNEXT()	Функция возвращает следующий код шифра платежа/поступления.
6	Функции получения суммы по шифру	FS_EXPCODE("Шифр")	Функция возвращает сумму по шифру платежа/поступления.
7	Функция добавления суммы по шифру	FS_EXPSETSUM("Тип шифра", "Код шифра", Сумма)	Добавить сумму по шифру. Тип шифра: Поступление - 0, Платеж - 1.

8	Функция получения суммы по текущему шифру	FS_EXPGETSUM()	Получить сумму по текущей строке
9	Функция удаления разности по шифру	FS_EXPDELSH("Код шифра")	Удалить разность по шифру. Код шифра - если "" - то текущий
	Общесистемные функции		
	Встроенные функции		
1	Абсолютная величина числа	ABS("выражение")	Функция возвращает абсолютное значение выражения.
2	Вернуть длину строки	LEN("строка")	Функция возвращает количество символов в строке.
3	Вернуть код символа	ASC("строка")	Возвращает ASCII-код первого символа строки.
4	Вернуть левую подстроку	LEFT("строка", N)	Возвращает N левых символов строки.
5	Вернуть позицию вхождения	BP_INSTR("строка1", "строка2")	Возвращает позицию вхождения строки2 в строку1. Если функция вернула -1, то вхождение не обнаружено.
6	Вернуть позицию вхождения с начальной позиции	INSTR1(start, "строка1", "строка2")	Возвращает позицию вхождения строки2 в строку1. Параметр start определяет позицию начала поиска. Если функция вернула -1, то вхождение не обнаружено.
7	Вернуть последовательность символов	REPLICATE("символ", количество)	Функция возвращает строку, состоящую из указанного количества символов.
8	Вернуть правую подстроку	RIGHT("строка", N)	Функция возвращает N правых символов строки.
9	Вернуть символ по коду	CHR(ASCII-код)	Функция возвращает символ, соответствующий ASCII-коду, переданому в качестве параметра.

10	Вернуть фрагмент строки	GETSUBST("строка", номер, "разделитель")	Функция возвращает фрагмент строки с указанным номером. Разделителем фрагментов служит указанный набор символов. Пример: ST = "Иванов::Петров::Сидоров" GETSUBST(ST, 2, "::") вернет "Петров"
11	Возвращает восьмиричную строку	OCT("выражение")	Функция возвращает строку восьмиричных символов, соответствующую арифметическому выражению, округляя его до целого.
12	Возвращает наибольшее целое	INT("выражение")	Возвращает наибольшее целое, равное или меньшее специфицированного арифметического выражения.
13	Возвращает шестнадцатиричную строку	HEX("выражение")	Возвращает строку шестнадцатиричных символов, соответствующую арифметическому выражению, округляя его до целого.
14	Дополнить строку символами по краям	PADC("строка", количество, "символ")	Функция дополняет строку по краям символами до указанного количества.
15	Дополнить строку символами слева	PADL("строка", количество, "символ")	Функция дополняет строку слева символами до указанного количества.
16	Дополнить строку символами справа	PADR("строка", количество, "символ")	Функция дополняет строку справа символами до указанного количества.
17	Значение переменной окружения	ENVIRON("строка")	Функция возвращает значение переменной окружения DOS, заданной строкой-параметром.

18	Конвертировать в FLOAT	CSNG("выражение")	Конвертирует арифметическое выражение в FLOAT-значение.
19	Округлить выражение в длинное целое	CLNG("выражение")	Функция округляет арифметическое выражение в длинное целое число (в диапазоне от -2147483648 до 2147483647).
20	Округлить выражение в целое	CINT("выражение")	Функция округляет арифметическое выражение в целое число (в диапазоне от -32768 до 32767).
21	Отсечение дробной части	FIX("выражение")	Возвращает целую часть арифметического выражения.
22	Преобразование в верхний регистр	UCASE("строка")	Функция конвертирует все символы строки в верхний регистр.
23	Преобразование в нижний регистр	LCASE("строка")	Конвертирует все символы строки в нижний регистр.
24	Преобразование строки в FLOAT-число	CVS("числовая строка")	Конвертирует числовую строку в FLOAT-значение.
25	Преобразование строки в длинное целое	CVL("числовая строка")	Конвертирует числовую строку в длинное целое.
26	Преобразование строки в целое число	CVI("числовая строка")	Конвертирует числовую строку в целое значение.
27	Преобразование строки в число	CVD("числовая строка")	Конвертирует числовую строку в DOUBLE-значение.
28	Системная дата	DATE	Функция возвращает системную дату в виде встроенного типа даты. (Встроенная ф-ция VBScript)
29	Системное время	TIME	Функция возвращает системное время в виде встроенного типа времени. (Встроенная ф-ция VBScript)

30	Удалить ведущие пробелы	LTRIM("строка")	Функция удаляет ведущие пробелы в строке.
31	Установить язык для строковых функций	SET_LANGUAGE(<Наименование языка>)	Устанавливает наименование языка для использования в строковых функциях Возможные значения "RU" - русский, "UA" - украинский
Интерфейсные функции			
1	Вставить вложение для документа	INSERT_ATTACH (ТИП_ДОКУМЕНТА, ID_ДОКУМЕНТА)	Возвращает 0 - если не успешно, 1 - если удаление выполнено
2	Выбрать отчет	pchar REP_SELECT(uint1 Mode,uint2 CdForm) выбрать отчет	Mode - режим выбора (REP_REPORT,REP_PRN_DOC,REP_PRN_RST) CdForm- Код экранной формы Возвращает полный путь у файлу описания выбранного отчета
3	Завершение индикатора	INDIC_TERM()	Функция завершает индикатор.
4	Зажечь окно ожидания	COFFEE_START("Наименование процесса")	Функция зажигает окно с наименованием процесса.
5	Запустить отчет на выполнение	uint2 REP_RUN(pchar)	Выполняет заданный отчет REP_RUN("R225_001.UPF")
6	Инициализация индикатора	INDIC_INIT("Наименование процесса", кол-во)	Функция зажигает окно индицирования длительного процесса с количеством итераций равным "кол-ву".

7	Отправить e-mail	MAIL_SEND('Сервер', 'Логин', 'Пароль', порт сервера, 0, 'Имя отправителя', 'e-mail')	MAIL_SEND_NOMSG('Сервер', 'Логин', 'Пароль', порт сервера, 0, 'Имя отправителя', 'e-mail отправителя', 'тема', 'e-mail получателя', 'заголовок', 'текст', 'путь к файлу вложения') Функции отправляет электронное письмо. MAIL_SEND_NOMSG - подавляются сообщения об ошибках
8	Отправить e-mail с подавлением сообщений	MAIL_SEND_NOMSG('Сервер', 'Логин', 'Пароль', порт сервера, 0, 'Имя отправителя', '')	Функции отправляет электронное письмо. подавляются сообщения об ошибках
9	Перерисовка индикатора	INDIC_REDRAW(итерация)	Функция перерисовывает индикатор, проинициализированный функцией INDIC_INIT().
10	Погасить окно ожидания	COFFEE_STOP()	Функция гасит окно с наименованием процесса.

11	Скопировать ссылки на вложения	COPY_ATTACH (ТИП_ДОКУМЕНТА, ID_ДОКУМЕНТА, ТИП_ДОКУМВВхода , ID_ДОКУМВхода)	<p>1. "ТИП_ДОКУМЕНТА" - значения типа документа из которого будет происходить копирование вложения</p> <p>2. "ID_ДОКУМЕНТА" - rcd документа из которого будет происходить копирование вложения</p> <p>3. "ТИП_ДОКУМВВхода" - значения типа документа в который будет происходить копирование вложения</p> <p>4. ID_ДОКУМВхода - rcd документа в который будет происходить копирование вложения</p> <p>если "ТИП_ДОКУМВВхода" равен нулю то копируются ссылки на "ТИП_ДОКУМЕНТА"</p> <p>Возвращает количество скопированных ссылок</p>
Модули пользователя			
1	Запустить модуль пользователя	int RUNUSERMODULE (uint4 CdMd)	Запустить модуль пользователя с кодом модуля CdMd
2	Запустить реестр пользователя	int RUNUSERRST(uint2 CdRst)	Запустить реестр пользователя с кодом реестра CdRst
3	Обновить запись в реестре	REFRESH1REC ()	Обновить текущую запись в пользовательском реестре
4	Обновить реестр	REFRESHUSERRST ()	Обновить пользовательский реестр
5	Открыть документ ИС-ПРО	uint4 OPEN_USE_DOCUM ENT(ID документа , ID типа документа)	Возвращает 1 если документ открыт успешно, иначе 0
6	Отметить/снять отметку записи реестра	uint4 MARKUNMARKID(ui nt4 Id)	Отметить/снять отметку записи реестра

7	Очистить все марки реестра	void CLEARALLMARK()	Очистить все марки реестра
8	Получить ID первой отмеченной записи реестра	int GETFIRSTMARKID()	Возвращает ID первой отмеченной записи в реестре. Если ни одна запись не выбрана, то функция вернет 0.
9	Получить ID следующей отмеченной записи реестра	int GETNEXTMARKID()	Возвращает ID следующей отмеченной записи в реестре. Если произведен полный проход по отмеченным записям, то функция вернет 0.
10	Получить значение указанной ячейки в реестре	GETFIELDVALUE("ИМЯ_КОЛОНКИ")	Функция возвращает значение ячейки текущей установленной записи в реестре по колонке "ИМЯ_КОЛОНКИ".
Прочие функции			
1	Вывести на экран сообщение об ошибке	BP_ERROR("Сообщение")	Функция выводит на экран сообщение об ошибке.
2	Вывести сообщение на экран	MESSAGE("Сообщение")	Функция выводит сообщение на экран.
3	Выдача на экран диалога пользователя	QUERY("Сообщение ")	Функция выдает на экран сообщение в диалоговом окне с кнопками "Да" "Нет"
4	запрашивать ли параметры XML при обновлении	ISASK_REFRESH_XML()	запрашивать ли параметры XML при обновлении 1 - запрашивать 0 - нет
5	Настроить сумму на валюту	SET_CURRENCY(Сумма , "Валюта")	Функция проставляет место точки для суммы по указанной валюте.
6	Получить курс на дату	CURRENCY_CURS("Валюта", Дата, "Котировка")	Функция возвращает курс валюты на дату из справочника курсов валют.

	Функции ввода значений полей		
1	Ввод даты в режиме диалога	INPUT_D("Заголовок", "Описание значения")	Функция производит ввод даты в режиме диалога.
2	Ввод скрытой строки в режиме диалога	INPUT_PASS ("Заголовок", "Описание значения")	Функция производит ввод скрытой строки (пароль).
3	Ввод строки в режиме диалога	INPUT_S("Заголовок", "Описание значения")	Функция производит ввод строки в режиме диалога.
4	Ввод числа в режиме диалога	INPUT_N("Заголовок", "Описание", Место точки)	Функция производит ввод действительного числа в режиме диалога.
5	Диалог редактирования полей	PICTURE_READ()	Диалог редактирования полей, добавленных PICTURE_ADD. Возвращает признак успешного редактирования.
6	Диалог редактирования полей с заголовком	PICTURE_READH(Строка_заголовка)	Диалог редактирования полей, добавленных PICTURE_ADD. Возвращает признак успешного редактирования. Параметр "Строка_заголовка" определяет заголовок окна.
7	Добавление редактируемого поля	PICTURE_ADDFLD("Описание", Поле)	Функция производит добавление редак. поля с такими же реквизитами как "Поле" и присваивает значение по умолчанию как у "Поля".
8	Добавление редактируемого поля	PICTURE_ADD("Описание", Тип, Размер)	Функция производит добавление редак. поля. Тип: "S" "N" "D" "T". Размер: длина поля точка.
9	Получение введенного значения	PICTURE_GET()	Функция возвращает значение поля после редак. Вызов в том же порядке, что и вызовы PICTURE_ADD

10	Привязка справочника к редактируемому полю	PICTURE_SETSPR(Тип_справочника, Код_справочника)	<p>Устанавливает связь поля, добавленного предыдущим вызовом PICTURE_ADD или PICTURE_ADDFLD, со справочником.</p> <p>Для такого поля в окне выбора, которое выводится PICTURE_READ или PICTURE_READH, можно вызвать справочник нажатием клавиши F3.</p> <p>Тип и код справочника выбираются на закладке "Справочники" окна выбора элементов программы.</p>
11	Результат выполнения последней операции ввода	CHECK_INPUT()	<p>Функция возвращает результат выполнения последней функции ввода значения INPUT_x. Возвращает 1, если значение было введено, или 0, если пользователь отказался от ввода значения.</p>
Функции для работы с Олап-кубами			
1	Восстановить параметры отчета	uint4 LOADREPPARAM(uint4 pCdUsr,uint4 pCdRep, pchar pPis, uint4 pSize)	<p>Восстановить параметры отчета</p> <ul style="list-style-type: none"> - uint4 pCdUsr - как вариант - код пользователя - uint4 pCdRep - как вариант - код отчета - pchar pPis - указатель на структуру сохранения данных - uint4 pSize - размер структуры <p>Возвращает истинный размер структуры или 0 в случае ошибки</p>

2	Выдача на экран диалога пользователя	QUERYBOX("Сообщение ")	Функция выдает на экран сообщение в диалоговом окне с кнопками "Да" "Нет"
3	Добавить измерение куба	OLAP_AddDimension(Тип измерения, Наименование измерения для отображения,	ID поля измерения, Класс реквизита, Размер, Точность представления) Тип измерения: 1= Horiz, 2= Vert, 3=Free Класс реквизита: 1=строка, 2=число, 3=дата, 4=время
4	Добавить показатель куба	OLAP_AddDataField(Наименование показателя для отображения, ID поля показателя,	Класс реквизита, Размер, Точность представления) Класс реквизита: 1=строка, 2=число, 3=дата, 4=время
5	Добавить строку в куб	OLAP_PutData()	Добавить строку в куб
6	Завершить формирование куба и отобразить результат	OLAP_BuildEnd()	Завершить формирование куба и отобразить результат
7	Начать формирование куба	OLAP_BuildBegin()	Начать формирование куба
8	Отменить формирование куба	OLAP_BuildCancel()	Отменить формирование куба без отображения результата
9	Получить код текущего пользователя	uint4 BPRO_CURRENT_USER(void)	Получить код текущего пользователя
10	Получить полный номер станции	BPRO_STT_FULL()	Возвращает полный номер станции ИС-ПРО Пример: STATION = BPRO_STT_FULL()

11	Получить режим работы модуля	GETMODULEWORK MODE (pCdMd)	Возвращает режим работы модуля ВХОД: pCdMd - Код модуль ВЫХОД: PROT_WORK_MODE_DISABLED - Отключен PROT_WORK_MODE_NORMAL - Рабочий режим PROT_WORK_MODE_DEMO - Демо-режим
12	Сохранить параметры отчета	uint4 SAVEREPPARAM(uint4 pCdUsr, uint4 pCdRep, pchar pPis, uint4 pSize)	Сохранить параметры отчета - uint4 pCdUsr - как вариант - код пользователя - uint4 pCdRep - как вариант - код отчета - pchar pPis - указатель на структуру сохранения данных - uint4 pSize - размер структуры Возвращает размер pPis
13	Установить значение поля измерения или показателя	OLAP_SetValue(ID поля измерения или показателя, Значение)	Установить значение поля измерения или показателя
	Функции для работы с хинтами		
1	Закрыть вложенную ветвь Hint	void HINTCLOSELRVEL(voi d)	Закрыть вложенную ветвь Hint
2	Открыть вложенную ветвь Hint	void HINTOPENLRVEL(voi d)	Открыть вложенную ветвь Hint
3	Получить текст хинта	pchar SYSCDPTOTXT(int4 Rcd)	Получить текст хинта Rcd - Id записи системного протокола
	Функции календаря		

1	Количество воскресений в интервале дат	CLN_SUNDAYS5 (Дата начала интервала, Дата конца интервала)	Функция возвращает количество воскресений по 5-дневной рабочей неделе с учетом праздников и переносов в интервале дат.
2	Количество праздничных дней в интервале дат	CLN_HOLIDAYS5 (Дата начала интервала, Дата конца интервала)	Функция возвращает количество праздничных дней по 5-дневной рабочей неделе с учетом переносов в интервале дат.
3	Количество рабочих дней в интервале дат	CLN_WDAYS5_INT (Дата начала интервала, Дата конца интервала)	Функция возвращает количество рабочих дней по 5-дневной рабочей неделе с учетом праздников и переносов в интервале дат.
4	Количество рабочих дней в месяце	CLN_WDAYS5_MON (Дата)	Функция возвращает количество рабочих дней по 5-дневной рабочей неделе с учетом праздников и переносов.
5	Количество суббот в интервале дат	CLN_SATDAYS5 (Дата начала интервала, Дата конца интервала)	Функция возвращает количество суббот по 5-дневной рабочей неделе с учетом праздников и переносов в интервале дат.
Функции по работе с глобальными переменными			
1	Заблокировать запись таблицы	int LOCKRECORD(int IsMsg, pchar Name, uint4 IdRec)	IsMsg - 1 - не выдавать сообщение если запись уже заблокирована Name - имя таблицы с учетом регистра IdRec - ID -записи (Rcd) возврат 1 - если успешно 0 - нет

2	Заблокировать ресурс	int LOCKRESOURCE(uint4 CdRes, uint4 Ext, int IsMsg)	<p>CdRes - код ресурса 1...999999999</p> <p>Ext - дополнительный код или 0</p> <p>IsMsg - 1 - не выдавать сообщение если ресурс уже заблокирован</p> <p>возврат 1 - если успешно 0 - нет</p>
3	Открыть окно Заметок	void VIEW_NOTE(char WndName)	Функция открывает окно Заметок для окна WndName.
4	Получить значение глобальной переменной	GETVALUEFORNAME (pchar)	<p>Получить значение из переменной с именем pchar, которая была объявлена ранее с помощью функции SETVALUEFORNAME ПРИМЕР: GETVALUEFORNAME ("Name")</p>
5	Разблокировать запись таблицы	int UNLOCKRECORD(pchar Name, uint4 IdRec)	<p>Name - имя таблицы с учетом регистра</p> <p>IdRec - ID -записи (Rcd)</p> <p>возврат 1 - если успешно 0 - нет</p>
6	Разблокировать ресурс	UNLOCKRESOURCE(uint4 CdRes, uint4 Ext)	<p>CdRes - код ресурса 1...999999999</p> <p>Ext - дополнительный код или 0</p>
7	Создать глобальную переменную	SETVALUEFORNAME (pchar, fd)	<p>Создать переменную с именем pchar, в которой будет храниться значение fd, может быть передан любой тип данных ПРИМЕР: SETVALUEFORNAME ("Name", 123)</p>

8	Удалить глобальную переменную	DELETEVALUEFORNAME (pchar)	Удалить переменную с именем pchar из памяти, которая ранее была объявлена функцией SETVALUEFORNAME
Функции получения информации о системе			
1	Отправить событие	void SEND_EVENT (Код события)	Отправить событие
2	Получить вариант установки	BPRO_NET()	Возвращает вариант установки БЭСТ-ПРО: 0 - локальный 1 - сетевой Пример: NET = BPRO_NET()
3	Получить версию системы	BPRO_VER()	Возвращает версию БЭСТ-ПРО в формате "V.MM.RRR" Пример: VER\$ = BPRO_VER()
4	Получить номер станции	BPRO_STT()	Возвращает номер станции БЭСТ-ПРО Пример: STATION = BPRO_STT()
5	Получить путь к каталогу рабочей станции	BPRO_CTL_STT()	Возвращает путь к каталогу рабочей станции системы БЭСТ-ПРО. Для локального варианта, возвращает каталог, в котором установлена система. Пример: STT\$ = BPRO_STT()
6	Получить путь к каталогу серверной части	BPRO_CTL_SRV()	Возвращает путь к каталогу серверной части системы БЭСТ-ПРО. Для локального варианта, возвращает каталог, в котором установлена система. Пример: SRV\$ = BPRO_SRV()

7	Получить режим работы	BPRO_REG()	Возвращает режим работы БЭСТ-ПРО: 0 - демонстрационный 1 - рабочий Пример: REG = BPRO_REG()
Функции преобразования			
1	Вернуть наименование месяца	FMONTH_NAME(Дата, Признак)	Возвращает строковое наименование месяца Признак = 0 со строчной буквы Признак = 1 с прописной буквы
2	Вернуть наименование месяца с указанием падежа	FMONTH_NM_PAD(Дата, Признак, Падеж)	Возвращает строковое наименование месяца с указанием падежа. Признак: 0-со строчной буквы, 1-с прописной буквы Падеж: 0-именительный,1-родительный,2-предложный
3	Вернуть строку времени	FTIME_STR(Время)	Формирует строку времени (ЧЧ:ММ)
4	Вернуть строку даты	FDATE_STR(Дата, Формат)	Формирует строку даты в заданном формате 0 - 7: 0 - ДД месяц ГГ 1 - ДД Месяц ГГ 2 - ДД месяц ГГГГ 3 - ДД Месяц ГГГГ 4 - месяц'ГГ 5 - Месяц'ГГГГ 6 - ДД/ММ/ГГ 7 - ДД/ММ/ГГГГ
5	Восстановить именительный падеж для ФИО	GET_NOMINATIVE_P ADEG (строка ФИО)	возвращает восстановленный именительный падеж для ФИО
6	Округление до заданной точности	BP_ROUND(Число, Точность)	Округляет число до заданной точности.
7	Перевод строки в дату	TXTTODAT ("Дата")	Перевод строки в дату. Формат даты: "DD/ММ/YY" или "DD/ММ/YYYY"
8	Перевод строки во время	TXTTOTIM ("Время")	Перевод строки во время. Формат времени: "ЧЧ:ММ"

9	Перевод строки из формата ANSI в OEM	ANSI_TO_OEM(Строка_ANSI)	Переводит строку Строка_ANSI в формат OEM и возвращает ее
10	Перевод строки из формата OEM в ANSI	OEM_TO_ANSI(Строка_OEM, Строка_ANSI)	Переводит строку Строка_OEM в формат ANSI и возвращает ее
11	Перевод строки из формата OEM в Таблицу 866	OEM_TO_T866(Строка_OEM, Строка_866)	Переводит строку Строка_OEM в формат Таблицы 866 и возвращает ее
12	Перевод строки из формата Таблицы 866 в OEM	T866_TO_OEM(Строка_866, Строка_OEM)	Переводит строку Строка_866 в формат OEM и возвращает ее
13	Перевод числа в текст без валюты	NUMTOTXT(Число, Род, Регистр)	Перевод целого числа в текст без валюты Род: 1 - мужской, 2 - женский, 3 - средний Регистр: 0 - со строчной, 1 - с прописной, 2 - все прописные, 3 - все строчные
14	Перевод числа в текст с наименованием валюты	NUMTOVAL(СУММА, "ВАЛЮТА", КОП, ПОЗ)	Перевод числа в текст с наименованием валюты. КОП: 0 - с разменной единицей, 1 - без ПОЗ: 0 - валюта после суммы, 1 - до Например: NUMTOVAL(15.67, "USD", 1, 1)
15	Показать дробную часть числа	BP_FRACTIONAL(Число, Точность)	показать дробную часть 0 - показать копейки 2 знака 1 - показать копейки 1 знак
16	Преобразование И.О.Ф	GETSHORTIOF(<Строка>)	Преобразование Ф.И.О. из "Фамилия Имя Отчество" в "И. О. Фамилия"
17	Преобразование Ф.И.О.	GETSHORTFIO(<Строка>)	Преобразование Ф.И.О. из "Фамилия Имя Отчество" в "Фамилия И.О."
18	Преобразование числа в строку	NUMTOSTR(Число)	Преобразует число в строку

19	Преобразование числа в строку с параметрами	NUMTOSTRP(Число, "Разделитель_дробной", "Разделитель_триад")	Преобразует число в строку с явным указанием символа разделителя дробной части и разделителем триад. Пример: SUM_CHR = NUMTOSTRP(SUM, ";", "")
20	Склонение ФИО	GET_FIO_PADEG(строка ФИО, пол, склонение)	возвращает склонение ФИО пол может принимать значения: 0 - женский 1 - мужской 2 - автоопределение склонение может принимать значения: 1 - Именительный Номинатив (Nominativ) Есть Кто? Что? 2 - Родительный Генитив (Genitiv) Нет Кого? Чего? 3 - Дательный Датив (Dativ) Давать Кому? Чему? 4 - Винительный Аккузатив (Accusativ) Винить Кого? Что? 5 - Творительный Аблатив (объединяет инструментатив [Instrumental], локатив и аблатив) Доволен/Сотворён Кем? Чем? 6 - Предложны
21	Усечение до заданной точности	TRUNC(Число, Точность)	Усекает число до заданной точности.
Функции работы с аппаратурой			
1	Электронные весы - вес от внешнего драйвера	SCALE_OUT(Номер_драйвера)	возвращает вес в граммах.
2	Электронные весы - Получить вес	SCALE_WEIGHT(Тип_весов, Номер_порта)	Возвращает вес в граммах. Тип весов: 1 - 60-тонные 2- платформенные автомобильные 2 - 2-тонные Номер порта: 1-COM1, 2-COM2, 3-COM3, 4-COM4

3	Электронные весы - Получить код ошибки	SCALE_ERROR()	Возвращает код ошибки функции SCALE_WEIGHT 0-Ошибки нет, 1-Ошибка чтения порта, 2-Ошибка записи в порт, 3-Весы вернули "пустой" вес, 4-Ошибка параметров, 5-Ошибка вызова драйвера, 6-Неверный тип весов
Функции работы с выделенными записями			
1	Очистить выделенные записи(марки) на станции	CLEAR_MARKS(Тип_марки)	Очистить выделенные записи(марки) на станции
2	Поместить номера выделенных записей на сервер	PUT_MRK_SRV(Тип_марки)	Помещает номера выделенных записей заданного типа на сервер, возвращает их кол-во.
3	Удалить все номера, помещенные на сервер	DEL_MRK_SRV(Тип_марки)	Удаляет все номера, помещенные на сервер с указанным типом.
Функции работы с датами			
1	Вернуть год из даты	YEAR (Дата)	Функция возвращает год из даты.
2	Вернуть день из даты	DAY(Дата)	Функция возвращает день из даты.
3	Вернуть месяц из даты	MONTH (Дата)	Функция возвращает номер месяца из даты.
4	Дать дату первого дня года	YEAR_BEG (Дата)	Функция преобразует дату в дату первого дня года.
5	Дать дату первого дня квартала	QUARTER_BEG(Дата)	Функция преобразует дату в дату первого дня квартала.
6	Дать дату первого дня месяца	MONTH_BEG(Дата)	Функция преобразует дату в дату первого дня месяца.

7	Дать дату последнего дня года	YEAR_END (Дата)	Функция преобразует дату в дату последнего дня года.
8	Дать дату последнего дня квартала	QUARTER_END(Дата)	Функция преобразует дату в дату последнего дня квартала.
9	Дать дату последнего дня месяца	MONTH_END(Дата)	Функция преобразует дату в дату последнего дня месяца.
10	Дать день недели	DAY_OF_WEEK(Дата)	Возвращает номер дня недели (от 1 до 7), который соответствует переданной в качестве параметра дате.
11	Дать кол-во дней недели	DAYSOFWEEK_CNT(Дата1, Дата2, День недели)	Возвращает кол-во дней недели (третий параметр), которые попали в промежуток между датами "Дата1" и "Дата2".
12	Количество дней в месяце	MONTH_SIZE (Месяц , Год)	Функция возвращает количество календарных дней в месяце.
13	Сместить дату на n месяцев	GO_MONTH(Дата, n)	Функция смещает дату на n месяцев.
14	Формирование даты из ее составляющих	DATE_PACK(День,Месяц,Год)	Возвращает дату
Функции работы с динамическими массивами			
1	Возвращает значение либо ключ текущего элемента	ARS_GET_CUR(ARRAY_DESCR, RESULT_TYPE)	Считывает в переменную значение либо ключ текущего элемента массива с заданным дескриптором. Если RESULT_TYPE = 1, то возвращается ключ, если RESULT_TYPE = 0, то возвращается значение. Пример: NAME = ARS_GET_CUR(ARRAY_DESCR, 0) KEY = ARS_GET_CUR(ARRAY_DESCR, 1)

2	Присваивание значения элементу массива	ARS_SET(Дескриптор_массив а, СтроковыйИндекс, Значение)	Присваивает значение указанному элементу массива с заданным дескриптором. Пример: ARS_SET(ARRAY_DESCR, "KEY_25_1", 2005)
3	Присвоить значение элементу массива	ARRAY_SET(Дескриптор_массив а, Индекс, Значение)	Присваивает значение указанному элементу массива с заданным дескриптором. Пример: ARRAY_SET(ARRAY_DESCR, 25, "Склад 25")
4	Прочитать значение элемента массива	ARRAY_GET(Дескриптор_массив а, Индекс)	Считывает в переменную значение указанного элемента массива с заданным дескриптором. Пример: SKL_NAME = ARRAY_GET(ARRAY_DESCR, INDEX)
5	Создание массива со строковым индексом	ARS_CREATE()	Создает динамический массив. Возвращает дескриптор массива или 0 в случае неудачи.
6	Создать динамический массив	ARRAY_CREATE()	Создает динамический массив. Возвращает дескриптор массива или 0 в случае неудачи.
7	Уничтожение массива со строковым индексом	ARS_DESTROY(Дескриптор_массив а)	Уничтожает динамический массив с указанным дескриптором.
8	Уничтожить динамический массив	ARRAY_DESTROY(Дескриптор_массив а)	Уничтожает динамический массив с указанным дескриптором.
9	Установка итератора массива на первый элемент	ARS_FIRST(ARRAY_DESCR)	Установка итератора массива с заданным дескриптором на первый элемент. Возвращает 1 в случае успеха, 0 иначе.

10	Установка итератора массива на следующий элемент	ARS_NEXT(ARRAY_DESCR)	Установка итератора массива с заданным дескриптором на следующий, относительно текущего, элемент. Возвращает 1 в случае успеха, 0 - иначе.
11	Чтение значения элемента массива	ARS_GET(Дескриптор_массив а, СтроковыйИндекс)	Считывает в переменную значение указанного элемента массива с заданным дескриптором. Пример: NAME = ARS_GET(ARRAY_DESCR, INDEX)
Функции работы с файлами			
1	Загрузка файлов DBF в SQL	void LOAD_DBF_IN_SQL(const char *PATH, int Code, int Mode)	PATH - путь к каталогу с DBF-файлами Code если кодировка не ANSI то выбор(предпочтение) между DOS == 1 или OEM == 0 Mode 1 - создание таблицы 0 - дополнение
2	Закреть файл	FILE_CLOSE(Дескриптор_файла)	Закрывает файл с указанным дескриптором.
3	Запись в файл	FILE_WRITE(Дескриптор_файла, Строка_источник)	Записывает строку в файл с указанным дескриптором. Добавляет к строке символы перевода строки. Возвращает: 1 - успешно, 0 - неуспешно.
4	Запись в файл без перевода строки	FILE_WRITENC(Дескриптор_файла, Строка_источник)	Записывает строку в файл с указанным дескриптором. Не добавляет к строке символы перевода строки. Возвращает: 1 - успешно, 0 - неуспешно.

5	Открыть файл	FILE_OPEN(Имя_файла, Тип_открытия)	Открывает файл с указанным именем. Возвращает дескриптор файла или 0 в случае неудачи. Тип открытия: 0-чтение,1-запись,2-дозапись в конец.
6	Проверка наличия файла	FILE_EXISTS(Имя_файла)	Проверяет наличие файла с указанным именем. Возвращает: 0 - файл не найден 1 - файл найден
7	Установка на позицию в файле	uint4 FILE_SETPOS(uint2 pDescr, uint4 Pos)	Устанавливает на позицию в файле Pos Возвращает: предыдущую позицию если успешно, -1 - неуспешно.
8	Чтение из файла	FILE_READ(Дескриптор_файла, Поле_строки_приемник)	Читает текст из файл с указанным дескриптором. Символы перевода строки не помещаются в строку. Возвращает: 1 - успешно, 0 - конец файла, -1 - неуспешно.
	Функции работы с фильтрами пользователя		
1	Наименование активного фильтра пользователя	ISR_CUR_QWR()	Возвращает наименование активного фильтра пользователя.
2	Проверка активности фильтра пользователя	ISR_RUNNING()	Проверяет наличие активного фильтра пользователя. Возвращает: 0 - Фильтр пользователя не установлен 1 - Фильтр пользователя установлен
	Функции работы со справочниками		
1	ID выбранной из справочника записи	SPR_OPEN_GETID()	Получить ID выбранной записи после вызова SPR_OPEN_DIALOG

2	Дать имя интегрированного справочника по коду	GET_SPR_BASE(Код_справочника)	По коду проблемного интегрированного справочника дать наименование базы справочника.
3	Дать полное наименование из справочника	GET_SPR_FULLNM(Тип_Справочника, Код_Справочника, "Код_в_справочнике")	Функция возвращает полное наименование из указанного справочника.
4	Диалог выбора из справочника	SPR_OPEN_DLG(Тип справочника, Код справочника)	<p>Вызывает диалог выбора из справочника, возвращает 0 - отмена выбора, иначе ID выбранной записи.</p> <p>Получить значения выбранной записи можно функциями SPR_OPEN_GETID - получить ID выбранной записи SPR_OPEN_GETCD - получить код выбранной записи SPR_OPEN_GETNM - получить наименование выбранной записи</p>
5	Записать в справочник значение по коду	PUT_SPR_VALUE(Тип_справочника, Код_справочника, Код_в_справочнике, Значение)	<p>Добавляет или заменяет в справочнике запись с указанным кодом. Код возврата: 1 - успешно 0 - ошибка</p> <p>Тип и код справочника выбираются на закладке "Справочники" окна выбора элементов программы.</p> <p>Функция поддерживает работу с: - пользовательскими аналитическими справочниками (тип 2) - справочниками пользователя (тип 0)</p>

6	Код выбранной из справочника записи	SPR_OPEN_GETCD()	Получить код выбранной записи после вызова SPR_OPEN_DIALOG
7	Наименование выбранной из справочника записи	SPR_OPEN_GETNM()	Получить наименование выбранной записи после вызова SPR_OPEN_DIALOG
8	Получить дату периода действия аналит. справочника	SPR_ANL_GETDATEEND ("КОД СПР.", "КОД в СПР.")	Получить дату окончания периода действия аналитического справочника "КОД СПР." Код справочника "КОД в СПР." Код в справочнике
9	Получить дату периода действия аналит. справочника	SPR_ANL_GETDATEBEGIN ("КОД СПР.", "КОД в СПР.")	Получить дату начала периода действия аналитического справочника "КОД СПР." Код справочника "КОД в СПР." Код в справочнике
10	Получить номер аналитического справочника по коду	SPR_ANL_GETNUM(Код_справочника)	Возвращает номер аналитического справочника по указанному коду аналитического справочника.
11	Поместить значение в иерархический справочник	PUT_TSPR_VALUE(Тип_Спр, Код_Спр, Код_В_Спр, Наим_В_Спр, Код_Род_Записи)	Поместить значение в иерархический справочник.
12	Поместить полное наименование в справочник	PUT_SPR_FULLNM(Тип_Спр, Код_Спр, Код_В_Спр, Длин_Наим)	Поместить длинное наименование в справочник.

13	Прочитать из справочника значение по коду	GET_SPR_VALUE(Тип_справочника, Код_справочника, Код_в_справочнике)	Возвращает значение, прочитанное из справочника по указанному коду в справочнике. Тип и код справочника выбираются на закладке "Справочники" окна выбора элементов программы.
Функции работы со строками			
1	Вставка подстроки в строку	PUTSTR_TO_STR("Исх.стр","Иск.пст","Пст",Реж,Поз)	Вставка подстроки("Пст") в строку ("Исх.стр") Режим(Реж): 0 - перед "Иск.пст" 1 - после 2 - заменить 3 - с Поз Если Реж < 3 Поз = длина "Пст"
2	Проверка допустимости вхождения	INS_CDCHAR(Символ)	Функция проверки допустимости вхождения символа в код. Функция проверяет принадлежность символа к разделителям.
3	Проверка кода	INS_CDN_VAL("Код", Длина)	Функция проверки кода на несоответствие общим правилам задания кода.
4	Проверка строки	IN_STR("Строка","Список")	Функция проверки строки на вхождение в список допустимых значений.
5	Функция получения подстроки	SUBSTR("строка", позиция, размер)	Функция возвращает подстроку строки "строка" с позиции "позиция" длиной "размер".
Период расчёта зарплаты			
1	Вид учета рабочего времени в указанном периоде	ACC_TIME_TYPE(ДатаРП)	Вид учета рабочего времени в указанном расчетном периоде. ДатаРП - Дата расчетного периода.

2	Дата текущего Расчетного периода	CUR_RP()	Возвращает дату текущего Расчетного периода.
3	Дата текущего Учетного периода	CUR_UP()	Возвращает дату текущего Учетного периода.
Подходный налог UA			
1	Возвращает данные о доходах	PU04_GET_DOX(Дата_РП)	Возвращает данные о доходах для текущего работника по указанному расчетному периоду. Заполняет поля PU04_DOX_...
2	Строка кодов видов оплат "Доход и вычеты"	PU04_STR_VO(Дата_РП, Номер_Строки)	Возвращает строку, содержащую перечень кодов видов оплат "Доход и вычеты", на указанный расчетный период. Номер_Строки - порядковый номер строки (нумерация с нуля). Строк может быть несколько, для получения полного списка вызывать функцию пока она не вернёт пустую строку "", каждый раз наращивая параметр Номер_Строки.
Прочие функции расчёта зарплаты			
1	Заполнение таблицы входимости PayTV	PAY_TV_FILL(Раздел, Код)	Заполнение таблицы входимости PayTV Раздел: 1-Виды оплат, 2-фонды Код: строковый код вида оплаты или фонда
2	Кол-во дней 5-дневной недели из прошлых периодов	DAYS_W5_RP_VOP(ДатаРП, "СписокВО" КодСпискаВО)	Количество дней по 5-дневной рабочей неделе, занятое в указанном Расчетном периоде видами оплат из списка, введенными в прошлых периодах. Список содержит коды ВО отпусков и больничных, разделенные пробелами.

3	Количество дней 5-дневной недели по списку ВО	DAYS_W5_RP_VO(ДатаРП, "СписокВО" КодСпискаВО)	Количество дней по 5-дневной рабочей неделе, занятое в указанном Расчетном периоде видами оплат из списка. Список содержит коды ВО отпусков и больничных, разделенные пробелами.
4	Округление суммы "На руки" для работающих	TRUNC(Сумма, VPL_ROUND())	Округление суммы "На руки" для работающих сотрудников.
5	Округление суммы "На руки" для уволенных	TRUNC(Сумма, VPL_ROUND_UVL())	Округление суммы "На руки" для уволенных сотрудников.
6	Порядок округления суммы "На руки" для работающих	VPL_ROUND()	Порядок округления суммы "На руки" для работающих сотрудников. Используется для передачи параметра округления в функции ROUND() и TRUNC().
7	Порядок округления суммы "На руки" для уволенных	VPL_ROUND_UVL()	Порядок округления суммы "На руки" для уволенных сотрудников. Используется для передачи параметра округления в функции ROUND() и TRUNC().
Рабочее время для расчёта зарплаты			
1	Кол-во план.дней в текущ.расч.периоде	DAYS_PL()	Количество плановых рабочих дней в текущем Расчетном периоде.
2	Кол-во план.дней в текущ.расч.периоде без опр неаяв	DAYS_PL_WRK()	Количество плановых рабочих дней в текущем расчетном периоде без учета дней занятых оправданными неявками

3	Кол-во план.дней в текущ.расч.периоде без опр неяв	DAYS_PL_WRI(ДеньНач, ДеньКон)	Количество плановых рабочих дней в текущем расчетном периоде за указанный интервал дней, без учета дней, занятых оправданными неявками. ДеньНач - число месяца начала интервала (1-31) ДеньКон - число месяца конца интервала (1-31)
4	Кол-во план.дней в указ.расч.периоде	DAYS_PL_RP(ДатаРП)	Количество плановых рабочих дней в указанном расчетном периоде.
5	Кол-во план.дней в указ.расч.периоде без опр неяво	DAYS_PL_RP_WRK(ДатаРП)	Количество плановых рабочих дней в указанном расчетном периоде без учета дней занятых оправданными неявками.
6	Кол-во план.дней в указ.расч.периоде без опр неяво	DAYS_PL_RP_WRI(ДатаРП, ДеньНач, ДеньКон)	Количество плановых рабочих дней в указанном расчетном периоде за указанный интервал дней, без учета дней, занятых оправданными неявками. ДатаРП - дата расчётного периода ДеньНач - день начала интервала (1-31) ДеньКон - день конца интервала (1-31)
7	Кол-во план.часов в текущ.расч.периоде	HRS_PL()	Количество плановых рабочих часов в текущем Расчетном периоде.
8	Кол-во план.часов в текущ.расч.периоде без опр нея	HRS_PL_WRK()	Количество плановых рабочих часов в текущем расчетном периоде без учета дней занятых оправданными неявками.

9	Кол-во план.часов в текущ.расч.периоде без опр нея	HRS_PL_WRI(ДеньНач, ДеньКон)	Количество плановых рабочих часов в текущем расчетном периоде за указанный интервал дней, без учета дней, занятых оправданными неявками. ДеньНач - день начала интервала (1-31) ДеньКон - день конца интервала (1-31)
10	Кол-во план.часов в указ.расч.периоде	HRS_PL_RP(ДатаРП)	Количество плановых рабочих часов в указанном Расчетном периоде.
11	Кол-во план.часов в указ.расч.периоде без опр неяв	HRS_PL_RP_WRK(ДатаРП)	Количество плановых рабочих часов в указанном расчетном периоде без учета дней занятых оправданными неявками.
12	Кол-во план.часов в указ.расч.периоде без опр неяв	HRS_PL_RP_WRI(ДатаРП, ДеньНач, ДеньКон)	Количество плановых рабочих часов в указанном расчетном периоде за указанный интервал дней, без учета дней, занятых оправданными неявками. ДатаРП - дата расчётного периода ДеньНач - день начала интервала (1-31) ДеньКон - день конца интервала (1-31)
13	Кол-во факт.дней в текущ.расч.периоде	DAYS_FK()	Количество фактически отработанных дней в текущем Расчетном периоде.
14	Кол-во факт.дней в указ.расч.периоде	DAYS_FK_RP(ДатаРП)	Количество фактически отработанных дней в указанном Расчетном периоде.

15	Кол-во факт.дней в указ.расч.периоде в интервале	DAYS_FK_RP_INT(ДатаРП, ДеньН, ДеньК)	Количество фактически отработанных дней в указанном расчетном периоде в интервале дней.
16	Кол-во факт.дней в указ.расч.периоде по списку ВО	DAYS_FK_RP_VO(ДатаРП,"СписокВО" КодСпискаВО)	Количество фактически отработанных дней в указанном Расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
17	Кол-во факт.часов в текущ.расч.периоде	HRS_FK()	Количество фактически отработанных часов в текущем Расчетном периоде.
18	Кол-во факт.часов в указ.расч.периоде	HRS_FK_RP(ДатаРП)	Количество фактически отработанных часов в указанном Расчетном периоде.
19	Кол-во факт.часов в указ.расч.периоде в интервале	HRS_FK_RP_INT(ДатаРП, ДеньН, ДеньК)	Количество фактически отработанных часов в указанном расчетном периоде в интервале дней.
20	Кол-во факт.часов в указ.расч.периоде по списку ВО	HRS_FK_RP_VO(ДатаРП,"СписокВО" КодСпискаВО)	Количество фактически отработанных часов в указанном Расчетном периоде по списку видов оплат. Список видов оплат содержит коды видов оплат разделенные пробелом.
21	Плановый табель	PTB_FILL(ДатаРП)	Заполнение полей РТВ_... данными планового табеля за период ДатаРП. Если ДатаРП равна 0, плановый табель заполняется для текущего расчетного периода

22	Факт.дней в расч.периоде по списку ВО в интервале	DAYS_FK_RP_VOI(ДатаРП,"СписокВО" КодСпискаВО,ДеньН,ДеньК)	<p>Количество фактически отработанных дней в указанном Расчетном периоде по списку видов оплат в интервале дней месяца.</p> <p>Список видов оплат содержит коды видов оплат разделенные пробелом. ДеньН - число месяца с 1 по 31 или 0 ДеньК - число месяца с 1 по 31 или 0</p>
23	Факт.часов в расч.периоде по списку ВО в интервале	HRS_FK_RP_VOI(ДатаРП,"СписокВО" КодСпискаВО,ДеньН,ДеньК)	<p>Количество фактически отработанных часов в указанном Расчетном периоде по списку видов оплат в интервале дней.</p> <p>Список видов оплат содержит коды видов оплат разделенные пробелом. ДеньН - день начала с 1 по 31 или 0 - не учитывать ДеньК - день конца с 1 по 31 или 0 - не учитывать</p>
24	Фактический табель	FTB_FILL(ДатаУП,ДатаРП)	<p>Заполнение полей FTB_... данными фактического табеля за период ДатаРП с учетом данных табелирования до даты учетного периода ДатаУП. Если ДатаРП равна 0, табель заполняется для текущего расчетного периода Если ДатаУП равна 0, табель заполняется по состоянию на текущий учетный период.</p>

25	Часов по плану на дату	HRS_PL_DAT(Дата, Признак)	Количество плановых рабочих часов на указанную дату Признак: 0 - не учитывать индивидуальное табелирование, 1 - учитывать индивидуальное табелирование
Расчет зарплаты			
1	Загрузка результатов расчета для работника	PAY_CALC_TAKE(Табельный_номер)	Загружает результаты расчета для работника. Расчет должен быть выполнен функцией PAY_CALC_PRED. Результаты доступны с помощью функций RL_RP_FIRST/RL_UP/FIRST/RL_NEXT.
2	Расчет зарплаты за месяц	PAY_CALC_MON()	Рассчитывает заработную плату за текущий учетный период. Результаты расчета доступны с помощью функций RL_FIRST_RP/RL_FIRST_UP и RL_NEXT.
3	Расчет зарплаты на день	PAY_CALC_DAY(День)	Рассчитывает заработную плату на указанный день с начала месяца. Результаты расчета доступны с помощью функций RL_FIRST_RP/RL_FIRST_UP и RL_NEXT.

4	Расчет зарплаты по предприятию	PAY_CALC_PRED(Режим_расчета)	<p>Рассчитывает заработную плату для всех работников. Результаты расчета доступны с помощью функции PAY_CALC_TAKE.</p> <p>Режим расчета: 1-РЛ; 2-РЛ,Проводки; 3-РЛ,Фонды 4-РЛ,Проводки,Фонды 5-РЛ,Проводки,Фонды,Перечисления.</p> <p>Выход: 1 - ОК, 0 - Расчет был прерван</p>
5	Расчет начислений за месяц	NCH_CALC_MON()	<p>Рассчитывает начисления за текущий учетный период. Результаты расчета доступны с помощью функций RL_FIRST_RP/RL_FIRST_UP и RL_NEXT.</p>
6	Расчет начислений на день	NCH_CALC_DAY(День)	<p>Рассчитывает начисления на указанный день с начала месяца. Результаты расчета доступны с помощью функций RL_FIRST_RP/RL_FIRST_UP и RL_NEXT.</p>
Расчет стипендии			
1	Проверка актуальности расчета	CHK_STP_CLC(CA студента)	<p>Проверяется актуальность расчета. Если состояние неактуально, то выполняется расчет</p>
Расчет численности			

1	Округление численности (UA)	PAY_CHISL_RND(Численность, Место_точки)	<p>Округляет численность по правилу "парной цифры" согласно украинского законодательства.</p> <p>Численность - значение, подлежащее округлению. Место_точки - количество знаков после точки, до которых необходимо округлить Численность.</p> <p>Возвращает округленное значение численности.</p>
2	Расчет вхожд-я раб-ка в ср.спис.числ.в эквиваленте	PAY_CHISL_SSE(Дата_периода)	<p>Возвращает количество вхождений работника в среднесписочную численность в эквиваленте полной занятости в периоде Дата_периода (место точки 3).</p> <p>При использовании в отчетах необходим расчет зарплаты по предприятию.</p>
3	Расчет вхождения раб-ка в среднесписочную числ-ть	PAY_CHISL_SS(Дата_периода)	<p>Возвращает количество вхождений работника в среднесписочную численность в периоде Дата_периода (место точки 3).</p> <p>При использовании в отчетах необходим расчет зарплаты по предприятию.</p>
4	Расчет вхождения работника в списочную численность	PAY_CHISL_SP(Дата_периода)	<p>Возвращает количество вхождений работника в списочную численность в периоде Дата_периода (место точки 3).</p>

5	Расчет численности по предприятию	PAY_CHISL_PRED(Период_Начала, Период_Конца, Режим, СПЧ, ССПЧ, ССПЧЭ)	<p>Рассчитывает численность для всех или группы работников. Формирует на сервере таблицу PayCsl, содержащую рассчитанные размеры численности.</p> <p>Параметры функции: Период_Начала - Дата начального периода расчёта Период_Конца - Дата конечного периода расчёта (Расчёт численности производится за все месяцы с "Период_Начала" по "Период_Конца") Режим - Режим расчета: 0 - Для всех работников 1 - Для работников, отмеченных в реестре Лицевых счетов СПЧ</p>
	Реестр юридических обязательств		
1	Дать ID первой аналит.карточки по фин.обяз-ву.	FB_AnI_First(ID_финОбязательств)	Возвращает ID первой аналитической карточки по указанному ID финансового обязательства.
2	Дать ID первой аналит.карточки по юр.обяз-ву	UB_AnI_First(ID_юрОбязательств)	Возвращает ID первой аналитической карточки указанного юридического обязательства.
3	Дать ID следующей аналит.карточки по юр.обяз-ву	UB_AnI_Next()	Возвращает ID следующей аналитической карточки, указанного в UB_AnI_First юридического обязательства.
4	Дать ID следующей анал.карточки по фин. обяз-ву	FB_AnI_Next()	Возвращает ID следующей аналитической карточки финансового обязательства, указанного в FB_AnI_First.

5	Дать сумму по аналит. карточке фин.обяз-ва.	FB_Anl_GetSm(ID_финОбязательст ва, ID_анлКарточки)	Возвращает сумму аналитической карточки финансово обязательства.
6	Дать сумму по аналит.карточке юр.обяз-ва	UB_Anl_GetSm(ID_юрОбязательств а, ID_анлКарточки)	Возвращает сумму аналитической карточки юридического обязательства.
7	Добавить разность КЕКВ	UB_AddKEKV(ИД_ЮрОбяз, "Код"_или_ИдКЕКВ, ФлагОтзыв, СумРазн, СумОпл)	Добавляет разность КЕКВ к указанному юридическому обязательству. В параметре "Код"_или_ИдКЕКВ можно передавать как строковый код КЕКВ, так и Ид необходимого КЕКВ. Функция возвращает 1 в случае удачи и 0 в случае ошибки.
8	Добавить разность КЕКВ	FB_AddKEKV (ИД_ФинОбяз, "Код"_или_ИдКЕКВ, ИдРазнЮр, СумРазн, СумОпл)	Добавляет разность КЕКВ в указанное финансовое обязательство. Параметры: "Код"_или_ИдКЕКВ - строковый код или Ид шифра разности КЕКВ. ИдРазнЮр - Ид разности КЕКВ юридического обязательства. Функция в случае удачной обработки возвращает 1, иначе - 0.
9	Получит дату юридического обязательства	UB_GetUbDate(Ид)	Ид - ид юридического обязательства. Возвращает дату юридического обязательства.
10	Получит номер разности по КЕКВ для фин.об.	FB_GetKEKVNpp(Ид)	Ид - идентификатор КЕКВ. Возвращает номер КЕКВ для фин. обязательства.
11	Получит следующую разность по КЕКВ фин.обяз.	FB_GetNxtKEKV()	Возвращает Ид следующей в группе, созданной FB_GetFstKEKV, разности по шифрам.

12	Получит юридическое обязательство по финансовому	FB_GetUblId(Ид)	Возвращает Ид юридического обязательства по Ид финансового.
13	Получить дату документа-основания	UB_GetDcDate(Ид)	Ид - ид юридического обязательства. Возвращает дату документа-основания.
14	Получить дату фин.обязательства	FB_GETREGDATE(Ид_фин_обязательств а)	Возвращает дату фин.обязательства
15	Получить дату финансового обязательства	FB_GetFbDate(Ид)	Возвращает дату финансового обязательства по его Ид.
16	Получить Документ фин.обязательства	FB_GETREGNUM(Ид_фин_обязательств а)	Возвращает код Документа фин.обязательства
17	Получить другие существенные условия	UB_GetOther(ид)	Ид - идентификатор юридического обязательства. Возвращает другие существенные условия.
18	Получить ид документа-основания	UB_GetDcId(Ид)	Ид - ид юридического обязательства. Возвращает ид документа-основания.
19	Получить ид контрагента	UB_GetKAgId(Ид)	Ид - ид юридического обязательства. Возвращает ид контрагента по юридическому обязательству.
20	Получить Ид контрагента по фин. обязательству	FB_GetKAgId(Ид)	Возвращает Ид контрагента финансового обязательства по Ид финансового обязательства.
21	Получить код разности по КЕКВ	UB_GetKEKVCd(Ид)	Ид - идентификатор КЕКВ. Возвращает код КЕКВ.
22	Получить код разности по КЕКВ	UB_GetKEKVCd(Ид_разности_КЕКВ)	Ид_разности_КЕКВ - Идентификатор разности КЕКВ. Возвращает код КЕКВ.

23	Получить код разности по КЕКВ для фин.об	FB_GetKEKVCd(Ид)	Ид - идентификатор КЕКВ. Возвращает код КЕКВ.
24	Получить код разности по КЕКВ для фин.об	FB_GetKEKVCd (Ид_Разности_КЕКВ)	Ид_Разности_КЕКВ - идентификатор разности КЕКВ. Возвращает код КЕКВ.
25	Получить номер документа-основания	UB_GetDcCd(Ид)	Ид - ид юридического обязательства. Возвращает указатель на строку, в к-ой хранится номер документа-основания.
26	Получить номер разности по КЕКВ	UB_GetKEKVNpp(Ид)	Ид - идентификатор КЕКВ. Возвращает номер КЕКВ.
27	Получить отметку фин.обяз. в окне статистики	FB_IsMarkedIdF(Ид_фин_обязательства)	Возвращает 1, если запись в реестре статистики отмечена. Иначе - 0.
28	Получить первую запись группы	UB_GetFirst(Ид,Дата1,Дата2)	Создать группу юридических обязательств по объекту учета (Ид) созданных с Дата1 по Дата2. Возвращает Ид первого юридического обязательства группы. Необходима для последующих вызовов UB_GetNext().
29	Получить первую запись группы фин.обязательств	FB_GetFirst()	Создать группу финансовых обязательств по текущему юридическому обязательству. Возвращает Ид первого финансового обязательства. Необходима для последующих вызовов FB_GetNext().

30	Получить первую разность по КЕКВ	UB_GetFstKEKV(Ид)	Создать группу разностей по шифрам юридического обязательства (Ид). Возвращает Ид первого шифра в группе. Необходима для последующего вызова ф-ции UB_GetNxtKEKV().
31	Получить первую разность по КЕКВ фин.обязательства	FB_GetFstKEKV(Ид)	Создает группу разности по шифрам финансового обязательства (Ид). Возвращает Ид первого шифра в группе. Необходима для последующих вызовов FB_GetNxtKEKV().
32	Получить первый код шифра КЕКВ	UB_GetFstKEKVs(Ид)	Возвращает строковый код первого шифра КЕКВ указанного обязательства или пустую строку в случае его отсутствия.
33	Получить признак объединения	UB_GetGrpFlg(ИД_юр.обяз.)	Возвращает признак объединения обязательства по указанному ИД. Возвращаемые значения: 1 - обязательство объединяемое 2 - обязательство объединяющее 3 - обязательство не включенное в объединение 0 - ошибка
34	Получить следующий код шифра КЕКВ	UB_GetNxtKEKVs()	Возвращает строковый код следующего шифра КЕКВ обязательства, указанного в UB_GetFstKEKVs.
35	Получить следующую запись группы	UB_GetNext()	Получить Ид следующего в группе, созданной вызовом ф-ции UB_GetFirst(), юридического обязательства.

36	Получить следующую разность по КЕКВ	UB_GetNxtKEKV()	Получить Ид следующей в группе, созданной вызовом функции UB_GetFrsKEKV(), разности по шифрам.
37	Получить следующую запись в группе фин.обязательств	FB_GetNext()	Возвращает Ид следующего финансового обязательства из группы, созданной FB_GetFirt().
38	Получить сумму документа-основания	UB_GetDcSm(Ид)	Ид - ид юридического обязательства. Возвращает сумму документа-основания.
39	Получить сумму оплаты фин.обязательства	FB_GetKEKVSmOp(Ид)	Ид - ид разности по КЕКВ. Возвращает сумму оплаты по разности по КЕКВ.
40	Получить сумму оплаты юридического обязательства	UB_GetDcPaid(Ид)	Ид - ид юридического обязательства. Возвращает сумму оплаты по юридическому обязательству.
41	Получить сумму разности по КЕКВ	UB_GetKEKVSm(Ид)	Ид - ид разности по КЕКВ Возвращает сумму разности по КЕКВ.
42	Получить сумму разности по КЕКВ фин.об.	FB_GetKEKVSm(Ид)	Ид - ид разности по КЕКВ. Возвращает сумму разности по КЕКВ.
43	Получить сумму разности по КЕКВ фин.обяз-ва расш.	FB_GetKEKVSmEx(Ид_финобязательства, "Код"_или_ИдКЕКВ)	Возвращает сумму разности по КЕКВ финансового обязательства. Ид_финобязательства - Ид финансового обязательства "Код"_или_ИдКЕКВ - строковый код или Ид разности КЕКВ.
44	Получить сумму разности по КЕКВ. Расш. реализация.	UB_GetKEKVSmEx(Ид_ЮрОбяз, "Код"_или_ИдКЕКВ)	Возвращает сумму разности по указанному строковому коду или ИД КЕКВ.

45	Получить сумму шифра группы	FB_GRPARTSM(ID_финОбязательства, ID_БанковскогоДокумента, ID_Шифра)	Возвращает сумму шифра оплаты финансового обязательства
46	Получить тип документа-основания	UB_GetDcTpCd(Ид)	Ид - ид юридического обязательства. Возвращает номер типа документа основания.
47	Проверить отметку юр.обяз.(основной вид)	UB_IsMarkedIdR(ID_обязательства)	Возвращает наличие отметки юр.обязательства с указанным ID.
48	Проверить отметку юр.обязательства по ID	UB_IsMarkedID(Ид)	Ид - идентификатор юридического обязательства. Возвращает 1, если отмечено. Иначе - 0.
49	Установить сумму аналитической карточки	UB_AnI_SetSm(ИД_ЮрОбяз, ИД_АнлКарт, Сумма)	Устанавливает сумму аналитической карточки по указанному обязательству. В случае, если карточки не существует, создается новая запись.
50	Установить сумму аналитической карточки	FB_AnI_SetSm(ИД_ФинОбяз, ИД_АнлКарт, Сумма)	Устанавливает сумму аналитической карточки по указанному финансовому обязательству. В случае, если карточки не существует, создается новая запись.
	Списки видов оплат		
1	Заполнить таблицу проводок, станд. способ форм.	PAY_VO_FILL_TMP_PRW()	Для стандартного способа формирования проводок заполняет временные таблицы tPayVoPrw и tPayVoPrwAnI
2	Наименование списка	SVO_NAME(КодСписка)	Возвращает наименование списка видов оплат по переданному коду списка видов оплат

3	Наличие вида оплаты в списке	SVO_PRESENT(КодСписка, "КодВО")	Проверяет наличие вида оплаты в списке видов оплат. 1 - вид оплаты присутствует в списке. 0 - вид оплаты отсутствует в списке.
	Списки датированных величин		
1	Величина из списка по дате	SDV_VALUE(КодСписка, Дата)	Возвращает величину из списка по переданному коду и дате
2	Дать первую запись из списка датированных величин	SDV_FIRST(КодСписка)	Возвращает первую запись из списка датированных величин. Заполняет поля SDV_...
3	Дать следующую запись из списка датированных величин	SDV_NEXT()	Возвращает следующую запись из списка датированных величин. Заполняет поля SDV_...
4	Наименование списка	SDV_NAME(КодСписка)	Возвращает наименование списка датированных величин по переданному коду списка
	Справочник "Виды оплат"		
1	Больничный - минимальный размер пособия (MPOT)	VOP_BLN_MROT(расчётный_период)	Возвращает минимальный размер пособия (MPOT) для расчёта больничного на указанный расчётный период. (RU)
2	Код группы вида оплаты	VOP_GRP("КодВО")	Возвращает код группы видов оплат в которую входит вид оплаты по переданному коду вида оплаты.
3	Код метода расчета вида оплаты	VOP_MET("КодВО")	Возвращает код метода расчета по переданному коду вида оплаты.

4	Максимальный размер исходной суммы	VOP_LIMIT(код_вида_оплаты, расчётный_период)	Возвращает размер максимальной исходной суммы для расчёта. код_вида_оплаты: числовой - внутренний код вида оплаты; строковый - внешний код вида оплаты.
5	Наименование вида оплаты	VOP_NAME("КодВО")	Возвращает наименование вида оплаты по переданному коду.
6	Наименование группы видов оплат	VOP_GRP_NAME(КодГр)	Возвращает наименование группы видов оплат по переданному коду группы.
7	Наименование метода расчета вида оплаты	VOP_MET_NAME(КодМет)	Возвращает наименование метода расчета видов оплат по переданному коду метода расчета.
8	Процент по размеру стажа	VOP_PRC_STJ(КодВО, Дней, Месяцев, Лет, КодТабл)	Возвращает процент от размера стажа по таблице.
	Справочник преамбул для приказов		
1	Наименование преамбулы	KDR_PAM_NM(Код_преамбулы)	Возвращает наименование преамбулы по переданному коду.
2	Текст преамбулы	KDR_PAM_TEXT(Код_преамбулы)	Возвращает текст преамбулы по переданному коду.
	Справочник производственных заказов		
1	Наименование заказа	PRD_ZAK_NAME("Н омерЗаказа")	Наименование производственного заказа по переданному номеру
2	Счет затрат заказа	PRD_ZAK_SCH("Ном ерЗаказа")	Счет затрат производственного заказа по переданному номеру
	Фонды и проводки		

1	Входимость видов оплат в фонд	GET_LST_VO(код фонда, дата_расчётного_периода)	Входимость видов оплат в фонд - строка с перечнем внутренних кодов видов оплат. Формат строки: "1,2,3" код фонда: числовой - внутренний код, строковый - внешний код
2	Загрузка фондов и проводок	FNDPRW_LOAD()	Загрузка фондов и проводок текущего открытого периода для текущего табельного номера
3	Заполнение таблицы ставок фондов	PAY_FILL_TPayFndStvTb()	Заполняет временную таблицу ставок из настройки "Фонды. Ставки. Таблица ставок."
4	Расчет фондов и проводок	FNDPRW_CALC()	Расчет фондов и проводок для текущего табельного номера в текущем учетном периоде
5	Чтение наименований записи проводок по видам оплат	PRWF_NM()	Чтение наименований записи проводок по видам оплат. Заполняются поля PRWF_...
6	Чтение наименований записи проводок по фонду	FNDF_PRW_NM()	Чтение наименований для текущей записи проводок по фонду. Заполняются поля FNDF_PRW_...
7	Чтение наименований записи фондов	FNDF_NM()	Чтение наименований для текущей записи фонда. Заполняются поля FNDF_...
8	Чтение первой записи проводок по видам оплат по РП	PRWF_RP_FIRST(ДатаРП 0)	Чтение первой записи проводок по видам оплат по расчетному периоду. Заполняются поля PRWF_... Если запись прочитана, возвращает 1, иначе 0.

9	Чтение первой записи проводок по видам оплат по УП	PRWF_UP_FIRST(ДатаУП 0)	Чтение первой записи проводок по видам оплат по учетному периоду. Заполняются поля PRWF_... Если запись прочитана, возвращает 1, иначе 0.
10	Чтение первой записи проводок по фонду	FNDF_PRW_FIRST()	Чтение первой записи проводок по текущей записи фонда. Заполняются поля FNDF_PRW_... Если запись прочитана, возвращает 1, иначе 0.
11	Чтение первой записи фондов по РП	FNDF_RP_FIRST(ДатаРП 0)	Чтение первой записи фондов по расчетному периоду. Заполняются поля FNDF_... Если запись прочитана, возвращает 1, иначе 0.
12	Чтение первой записи фондов по УП	FNDF_UP_FIRST(ДатаУП 0)	Чтение первой записи фондов по учетному периоду. Заполняются поля FNDF... Если запись прочитана, возвращает 1, иначе 0.
13	Чтение следующей записи проводок по видам оплат	PRWF_NEXT()	Чтение следующей записи проводок по видам оплат. Заполняются поля PRWF_... Если запись прочитана, возвращает 1, иначе 0.
14	Чтение следующей записи проводок по фонду	FNDF_PRW_NEXT()	Чтение следующей записи проводок по текущей записи фонда. Заполняются поля FNDF_PRW_... Если запись прочитана, возвращает 1, иначе 0.
15	Чтение следующей записи фондов	FNDF_NEXT()	Чтение следующей записи фондов. Заполняются поля FNDF_... Если запись прочитана, возвращает 1, иначе 0.

	Функци доступа к атрибутам карточки ОС		
1	Значение атрибута по коду	OSATR_GETZN(ATR_CD)	Значение атрибута по его коду в карточке ОС
2	Код первого атрибута	OSATR_FIRST()	Код первого атрибута в карточке ОС
3	Код следующего атрибута	OSATR_NEXT()	Код следующего атрибута в карточке ОС
4	Наименование атрибута по коду	OSATR_GETNM(ATR_CD)	Наименование атрибута по его коду
	Функци доступа к драгматериалам		
1	Вес драгматериала в карточке на дату	OSDRG_GETVES_B(DRG_CD, Дата)	Возвращает вес драгматериала по его коду для карточки ОС на дату
2	Вес драгматериала в карточке по коду	OSDRG_GETVES(DRG_CD)	Возвращает вес драгматериала по его коду для карточки ОС
3	ЕИ драгматериала в карточке на дату	OSDRG_GETEI_D(DRG_CD, Дата)	Наименование единицы измерения драгматериала по его коду для карточки ОС на дату
4	ЕИ драгматериала в карточке по коду	OSDRG_GETEI(DRG_CD)	Наименование единицы измерения драгматериала по его коду для карточки ОС
5	Код первого драгматериала	OSDRG_FIRST()	Возвращает код первого драгматериала в карточке ОС или пустую строку, если список драгматериалов в карточке пуст.
6	Код первого драгматериала на дату	OSDRG_FIRST_D (Дата)	Возвращает код первого драгматериала в карточке ОС на дату или пустую строку, если список драгматериалов в карточке пуст.

7	Код следующего драгматериала	OSDRG_NEXT()	Возвращает код следующего драгматериала в карточке ОС или пустую строку, если следующего драгматериала в списке нет.
8	Наименование драгматериала по коду	OSDRG_GETNM(DRG_CD)	Возвращает наименование драгматериала по его коду
Функции доступа к карточкам ОС			
1	Дать ID карточки ОС по номеру карточки и типу	OS_GET_CRDN_ID (Номер_Карточки, Тип)	Возвращает ID объекта ОС по указанному номеру карточки и типу карточки. Допустимые значения типа карточки: 0 - все типы (предполагается уникальность номера карточки в картотеке) 2 - простая 3 - количественная 6 - карточка с пробегом 7 - нематериальный актив 8 - МНМА 255 - Все типы ОС кроме МНМА
2	Дать инв.номер по номеру карточки и типу	OS_GET_CRDN (Номер_Карточки, Тип)	Возвращает инвентарный номер объекта ОС по указанному номеру карточки и типу карточки. Допустимые значения типа карточки: 0 - все типы (предполагается уникальность номера карточки в картотеке) 2 - простая 3 - количественная 6 - карточка с пробегом 7 - нематериальный актив 8 - МНМА 255 - Все типы ОС кроме МНМА

3	Дать первую карточку в ведомости НА по амортигруппе	OS_NA_CRD_FIRST (ID_Ведомости, Код_АмортГруппы)	Возвращает ID первой карточки в ведомости налоговой амортизации для амортигруппы (ОНУ). Если карточка не найдена, возвращает 0.
4	Дать первую карточку в документе ОС	OS_CRDD_FIRST (ID_Документа)	Возвращает Инв.номер первой карточки в документе ОС.
5	Дать первую карточку в картотеке по амортигруппе	OS_AG_FCRD (Код_АмортГруппы)	Возвращает ID первой карточки в картотеке для амортигруппы (ОНУ). Если карточка не найдена, возвращает 0.
6	Дать след. карточку в ведомости НА по амортигруппе	OS_NA_CRD_NEXT (ID_Ведомости, Код_АмортГруппы)	Возвращает ID следующей карточки в ведомости налоговой амортизации для амортигруппы (ОНУ). Если карточка не найдена, возвращает 0.
7	Дать след. карточку в картотеке по амортигруппе	OS_AG_NCRD (Код_АмортГруппы)	Возвращает ID следующей карточки в картотеке для амортигруппы (ОНУ) после карточки, найденной функцией OS_AG_FCRD или OS_AG_NCRD. Если карточка не найдена, возвращает 0.
8	Дать следующую карточку в локументе ОС	OS_CRDD_NEXT (ID_Документа)	Зачитывает карточку в документе ОС непосредственно после карточки, установка на которую производилась функциями OS_CRDD_FIRST() или OS_CRDD_NEXT() Возвращает Инв.номер карточки ОС или пустую строку, если следующей карточки в документе нет.

9	Дополнительные реквизиты грп.карточки	OS_GRP_INFO (ID_Док, Карт_Номер, Код_реквизита)	Возвращает дополнительные реквизиты групповой карточки по коду реквизита, номеру карточки и ID_Документа. При нулевом значении ID_Документа информация зачитывается в разрезе всей картотеки ОС. Возвращаемые реквизиты по кодам: 1 - Количество карточек в группе (быстрый подсчет, точность не гарантируется) 2 - Количество карточек в группе (Точный подсчет, но медленнее)
10	Количество карточек в документе ОС	OS_CRDD_QT (ID_Документа)	Возвращает количество карточек в документе ОС.
11	Подключение мех-ма объединения грп.карточек	OS_GRP_LINK (Режим_Вызова)	Устанавливает/отменяет режим объединения карточек группового учета (С одинаковым номером карточки) Режим вызова: 0-Установка / 1- Отмена При установленном режиме функции перебора карточек (OS_CRD*_NEXT) пропускают все карточки группового учета с одинаковым уже встретившимся номером. Симметричность вызовов обязательна!

12	Реквизиты карточки в ведомости НА	OS_NA_CRD_GETRK (Код_Реквизита)	<p>Возвращает значение реквизита карточки ОС в картотеке ОС или ведомости Налоговой Амортизации по его коду. Реквизиты зачитываются функциями OS_NA_CRD_FIRST / OS_NA_CRD_NEXT или OS_AG_FCRD/OS_AG_NCRD</p> <p>Коды реквизитов:</p> <p>0 - сумма начисленной амортизации по ведомости 1 - Инвентарный Номер 2 - Номер карточки</p>
13	Установка на карточку по инв.номеру	OS_GET_CRD ("Инвентарный номер")	<p>Устанавливает на запись в картеке ОС по указанному инвентарному номеру. Возвращает в случае успешной установки номер карточки, а противном случае - ноль.</p>
14	Установка на первую карточку	OS_CRD_FIRST ()	<p>Делает текущей первую запись в картеке ОС. Возвращает номер карточки ОС или ноль, если картотека пустая.</p>
15	Установка на первую отмеч.карточку	OS_MRK_FIRST ()	<p>Делает текущей первую отмеченную карточку в картеке ОС. Возвращает номер карточки ОС или ноль, если отметок нет.</p>

16	Установка на след.отмеченную карточку	OS_MRK_NEXT ()	<p>Делает текущей следующую отмеченную запись непосредственно после карточки, установка на которую производилась функциями OS_MRK_FIRST() или OS_MRK_NEXT()</p> <p>Возвращает номер карточки ОС или ноль, если текущая отмеченная запись в картотеке последняя.</p>
17	Установка на следующую карточку	OS_CRD_NEXT ()	<p>Делает текущей запись в картотеке ОС непосредственно после карточки, установка на которую производилась функциями OS_CRD_FIRST() или OS_CRD_NEXT()</p> <p>Возвращает номер карточки ОС или ноль, если текущая запись в картотеке последняя.</p>
Функции доступа к реквизитам карточки ОС			
1	Зачитать текущую запись архива движения	OS_IST_GETCUR ()	<p>Зачитывает для текущей записи архива движения карточки ОС реквизиты документа движения.</p> <p>Возвращает код документа или ноль, если текущая запись архива не определена.</p>
2	Знач-е рекв-та истории карточки на дату	OS_IST_DAT_RK (Дата, Код реквизита)	<p>Возвращает для записи архива движения карточки ОС значение реквизита по его коду на дату (Коды из справочника реквизитов - Отчет 110).</p> <p>Значение возвращается стипом, соответствующим данному реквизиту.</p>

3	Значение реквизита в архиве движения по коду	OS_IST_RK_ZN (Код реквизита)	Возвращает для записи архива движения карточки ОС значение реквизита по его коду (Коды из справочника реквизитов - Отчет 110). Значение возвращается с типом, соответствующим данному реквизиту.
4	Значение реквизита в документе движения по коду	OS_DOC_RK_ZN (Код реквизита)	Возвращает для записи архива движения карточки ОС значение реквизита документа движения по его коду (Коды из справочника реквизитов документов - Отчет 110). Значение возвращается с типом, соответствующим данному реквизиту.
5	Значение реквизита в заголовке по коду	OS_HEAD_RK_ZN (Код реквизита)	Возвращает для заголовка карточки ОС значение реквизита по его коду (Коды из справочника реквизитов - Отчет 110). Значение возвращается с типом, соответствующим данному реквизиту.
6	Значение реквизита истории карт. для документа	OS_DOC_CRD_RK (DOC_ID, CRD_ID, Режим, Код_реквизита)	Возвращает значение реквизита истории карточки ОС для записи документа в истории. Параметры: DOC_ID - ID документа CRD_ID - ID карточки Режим - режим зачитки (0 - с учетом изменений документа / 1 - реквизиты до документа) Код_реквизита - код реквизита истории карточки (отчет 110)

7	Износ всего карточки ОС для записи документа	OS_DOC_CRD_IZN (DOC_ID, CRD_ID, Режим_зачитки, код_реквизита)	Возвращает сумму износа всего для карточки ОС относительно записи в истории. Параметры: 1 - ID документа 2 - ID карточки 3 - режим зачитки (0 - с учетом изменений документа / 1 - реквизиты до документа) 4 - код реквизита: 1 - износ всего бухгалтерский 2 - амортизация налоговая всего
8	Инвентарный номер	OS_RK_INVNMR()	Инвентарный номер карточки ОС
9	Код амортизационной группы	OS_RK_CDONU(Дата)	Код группы налоговой амортизации на дату
10	Код метода амортизации бухгалтерский	OS_RK_CDALG(Дата)	Код метода бухгалтерской амортизации на дату
11	Код метода амортизации налоговый	OS_RK_CDALGN(Дата)	Код метода налоговой амортизации на дату
12	Код ОКОФ (RU)	OS_RK_OKOF()	Код ОКОФ (RU)
13	Код подразделения из лицевого счета МОЛ	OS_MOL_LSPDR (Код_МОЛ)	Возвращает код подразделения МОЛ, указанный в его Лицевом Счете. В качестве параметра передаётся код МОЛ из справочника МОЛ ОС (табельный номер).
14	Номер карточки	OS_RK_CRDNMR()	Номер карточки ОС
15	Первая строка комментария карточки	OSCMT_FIRST(Строка)	Читает 1-ю строку комментария Карточки ОС. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0

16	Последний свободный инв.номер	OS_INMR_LFREE()	Возвращает последний свободный в картотеке инвентарный номер карточки ОС.
17	Последний свободный инв.номер с заданным префиксом	OS_INMR_LPFREE("префикс")	Возвращает последний свободный в картотеке инвентарный номер карточки ОС по заданному префиксу.
18	Последний свободный номер карточки	OS_CNMR_LFREE (Тип_карточки)	Возвращает последний свободный в картотеке номер карточки ОС. Параметр "Тип_карточки" (NUM) равен нулю. В случае ненулевого значения параметра функция вернёт последний свободный номер для карточки типа ПНМА (UA).
19	Признак наличия реквизита в архиве движения	OS_IST_RK_PRZ (Код реквизита)	Возвращает единицу, если реквизит с указанным кодом содержится в архиве движения карточки ОС и ноль в противном случае Код реквизита из системного справочника реквизитов (Отчет 110)
20	Признак элемента групповой карточки	OS_RK_GRPGRD()	Возвращает единицу в случае, если карточка является элементом групповой и ноль в противном случае.
21	Проверка дублирования инвентарного номера	OS_INMR_CHECK("инв.номер")	Если инвентарный номер, указанный в качестве параметра, уже имеется в картотеке ОС, функция возвращает единицу. В противном случае, ноль. Проверка не учитывает ведущие пробелы.

22	Следующая строка комментария карточки	OSCMT_NEXT(Строка)	Читает следующую строку комментария Карточки ОС. Заполняет поле Строка Если строка прочитана, возвращает 1, иначе 0
23	Следующий свободный инв.номер	OS_INMR_NFREE("И нв.номер")	Возвращает следующий свободный в картотеке инвентарный номер карточки ОС после указанного.
24	Следующий свободный номер карточки	OS_CNMR_NFREE (Номер_карточки, Тип_карточки)	Возвращает следующий свободный в картотеке номер карточки ОС. после указанного. Параметр "Тип_карточки" (NUM) равен нулю. В случае ненулевого значения параметра функция вернёт следующий свободный номер для карточки типа ПНМА (UA).
25	Установка на первую запись архива движения	OS_IST_FIRST ()	Делает текущей первую запись архива движения карточки ОС. Возвращает дату записи архива.
26	Установка на последнюю запись архива движения	OS_IST_LAST ()	Делает текущей последнюю запись архива движения карточки ОС. Возвращает дату записи архива.
27	Установка на предыдущ.запись архива движения	OS_IST_PREV ()	Делает текущей предыдущую запись архива движения карточки ОС. Возвращает дату записи архива или ноль, если предыдущей записи не существует. Без предварительного вызова OS_IST_FIST или OS_IST_LAST возвращает ноль.

28	Установка на следующую запись архива движения	OS_IST_NEXT ()	Делает текущей следующую запись архива движения карточки ОС. Возвращает дату записи архива или ноль, если следующей записи не существует. Без предварительного вызова OS_IST_FIST или OS_IST_LAST возвращает ноль.
Функции взаиморасчетов с контрагентами			
1	Функция получения сальдо взаим. по контрагенту	GETPTN_SMBAL("Код контрагента",Дата, Валюта,Котировка)	Функция возвращает сальдо взаиморасчетов по коду контрагента на дату, в указанной валюте и котировке.
2	Функция получения сальдо по контрагенту на дату	GETPTN_SMBAL_D("Код контрагента", Дата)	Функция возвращает сальдо взаиморасчетов по коду контрагента на указанную дату
Функции генерации проводок			
1	Аналитическая карточка по счёту	ANLCRD("Счёт","К1","К2","К3","К4","К5","К6","К7","К8","К9","К10")	Функция создаёт аналитическую карточку на указанном счёте по указанным кодам на уровнях аналитики.

2	Аналитическая карточка по счету (SQL)	ANLCRDSQL("Счёт", "K1", "K2", "K3", "K4", "K5", "K6", "K7", "K8", "K9", "K10")	<p>Функция создаёт аналитическую карточку на указанном счёте по указанным кодам на уровнях аналитики.</p> <p>Причины, по которым нужно отдавать предпочтение ANLCRDSQL :</p> <p>1) очень частая установка определенного количества карточек с одинаковыми набором кодов на уровнях аналитики;</p> <p>2) карточки предполагаемого набора должны быть заведомо созданы;</p> <p>3) карточки предполагаемого набора должны быть актуальны.</p>
3	Аналитическая карточка по счету дебета	ANLCRDDT("Счёт", "K1", "K2", "K3", "K4", "K5", "K6", "K7", "K8", "K9", "K10")	<p>Функция создаёт аналитическую карточку на указанном счёте дебета по указанным кодам на уровнях аналитики.</p> <p>Полезна в случае частого использования вызовов вида :</p> <p>call SETANLCDT("Счет", ANLCRD_CREATE(.....))</p>
4	Аналитическая карточка по счету кредита	ANLCRDKT("Счёт", "K1", "K2", "K3", "K4", "K5", "K6", "K7", "K8", "K9", "K10")	<p>Функция создаёт аналитическую карточку на указанном счёте кредита по указанным кодам на уровнях аналитики.</p> <p>Полезна в случае частого использования вызовов вида :</p> <p>call SETANLCKT("Счет", ANLCRD_CREATE(.....))</p>
5	Проводка в валюте документа	ENTRY("СчДт", "СчКт", "Сумма", "Комментарий")	<p>Функция формирует проводку в валюте документа по указанной корреспонденции счетов.</p>

6	Проводка в валюте документа на дату	ENTRY_DAT("СчДт", "СчКт", Сумма, "Комментарий", Дата)	Функция формирует проводку в валюте документа по указанной корреспонденции счетов на дату.
7	Проводка в указанной валюте	POST("СчДт", "СчКт", Дата, "Валюта", Курс, Сумма, "Комментарий")	Функция формирует проводку по корреспонденции счетов.
8	Проводка в учетной валюте	ENTRY_DCUR("СчДт", "СчКт", Сумма, "Комментарий")	Функция формирует проводку в учетной валюте по указанной корреспонденции счетов.
9	Проводка в учетной валюте на дату	ENTRY_DCUR_DAT("СчДт", "СчКт", Сумма, "Комментарий", Дата)	Функция формирует проводку в учетной валюте по указанной корреспонденции счетов на дату.
10	Установка аналитическая карточки для счёта	SETANLC("Счет", ID_Карточки)	Функция устанавливает аналитическую карточку на указанном счете по коду карточки.
11	Установка аналитической карточки для счета дебета	SETANLCDT("Счет", ID_Карточки)	Функция устанавливает аналитическую карточку на указанном счете дебета по коду карточки.
12	Установка аналитической карточки для счета кредита	SETANLCKT("Счет", ID_Карточки)	Функция устанавливает аналитическую карточку на указанном счете кредита по коду карточки.
13	Установка признака переоценочной проводки	SETDIFRATEMODE() (Устаревшая реализация)	Функция устанавливает признак переоценочной для текущей формируемой проводки. Вызывается перед функциями генерации проводок или в программе вычисления суммы проводки.

14	Установка признака фильтрации проводки	SET_DFR_HIST()	<p>Функция устанавливает для текущей формируемой проводки признак того, что при переоценке сумма проводки будет пересчитываться по собственному (историческому) курсу.</p> <p>Указание этого признака приводит к тому, что курсовая разница по этой проводке не возникает.</p> <p>Функция работает только для украинского законодательства системы.</p>
15	Установка признаков переоценочной проводки	SETDIFRATE("Вал")	<p>Функция устанавливает признак переоценочной для текущей формируемой проводки. Также необходимо указать валюту переоценки в параметре "Вал".</p> <p>Вызывается перед функциями генерации проводок или в программе вычисления суммы проводки.</p>
	Функции для FR-отчетов (зарплата)		
1	Аналитические карточки по выбранному договору	PAY_GET_ANC_DOG()	
2	Аналитические карточки по выбранному подразделению	PAY_GET_ANC_PDR()	
3	Данные расчетного листа по подразделениям приписки	PAY_FR_KPURLPDRS PS (Начальный период, Конечный период, Режим)	<p>Начальный период - начальный отчетный период (первое число месяца)</p> <p>Конечный период - конечный отчетный период (первое число месяца)</p> <p>Режим - 0 - по всем работникам 1 - по списку работников (PayLstV1)</p>

4	Заполнение данных для отчётов по НДФЛ (RU)	PAY_FILL_PDX(pDtN, pDtK, pPdr, pIncSubPdr, pSteCd)	<p>Заполнение данных для отчётов по НДФЛ (RU)</p> <p>pDtN - дата начала отчётного периода</p> <p>pDtK - дата конца отчётного периода</p> <p>pPdr - признак "По указанным подразделениям": 1 - да, 0 - нет</p> <p>pIncSubPdr - признак "Включать подчинённые подразделения": 1 - да, 0 - нет</p> <p>pSteCd - код структурной единицы, если пусто - по всем структурным единицам</p> <p>pLstWrk - признак "формировать по списку работников":</p> <p style="padding-left: 40px;">Список работников должен находиться в таблице PayLstV1</p> <p style="padding-left: 40px;">Значения:</p> <p style="padding-left: 80px;">1 - в пол</p>
5	Заполнение данных для Приложения 23 в ПФУ (UA)	PAY_FR_DODPFU(Период, CE)	<p>Формирователь данных для Приложения 23 в ПФУ (UA)</p> <p>Период - учетный период</p> <p>CE - список кодов структурных единиц, разделенных запятыми.</p>
6	Заполнение данных для расчета больничного по ИФ	PAY_FR_BLNSF(pModule, pMode)	<p>Заполнение данных для расчета среднего заработка для оплаты больничного по источникам финансирования.</p> <p>pModule - код модуля</p> <p>pMode - (0 - терминация, 1 - заполнение данными)</p>
7	Заполнение данных для формы 4-ФСС (RU)	PAY_FR_4FSS(Период, CE)	<p>Формирователь данных для формы 4-ФСС (RU)</p> <p>Период - учетный период</p> <p>CE - список кодов структурных единиц, разделенных запятыми.</p>

8	Заполнение данных для формы 4-ФСС (RU) 2010	PAY_FR_4FSS_10(Период, CE)	Формирователь данных для формы 4-ФСС (RU) 2010 Период - учетный период CE - код структурной единицы
9	Заполнение данных для формы 4-ФСС (RU) 2011	PAY_FR_4FSS_11(Период, CE)	Формирователь данных для формы 4-ФСС (RU) 2011 Период - учетный период CE - код структурной единицы
10	Заполнение данных для формы П4 (RU)	PAY_FR_P4(pDate)	Заполнение данных для формы П-4 pDate - период, за который формируется отчет
11	Заполнение данных для формы Приложение 6 (UA)	PAY_FR_D6(Период, CE)	Формирователь данных для формы Приложение 6 (UA) Период - учетный период CE - список кодов структурных единиц, разделенных запятыми.
12	Заполнение данных для формы Ф4-ФСС (UA)	PAY_FR_F4(Период, CE, Признак)	Формирователь данных для формы Ф4-ФСС (UA) Период - учетный период CE - список кодов структурных единиц, разделенных запятыми. Признак - 1 - нарастающим итогом с начала года, 0 - за указанный период
13	Заполнение данных для экспорта в ПФР (RU)	PAY_FILL_PFR(pMode)	Заполнение таблиц для отчёта в ПФР (RU) pMode - режим работы (резерв, не используется)
14	Заполнение справочника "Группы категорий персонал"	PAY_FILL_SPRGKAT()	Заполнение справочника "Группы категорий персонала" (Списки категорий персонала, которые принадлежат определенной группе категорий).
15	Заполнение таблиц Kdr1Pv	PAY_PVN_FILL_KDR1PV(pFrom, pType)	

16	Заполнение таблиц баланса переработки	GET_KPU_STM(ДатаНачала, ДатаКонца, Признак)	ДатаНачала - Начало отчётного периода ДатаКонца - Конец отчётного периода Признак - Использовать список работников из таблицы PayLstV1, где PayLv_Val = Kpu_Rcd, PayLV_Part = номер станции
17	Заполнение таблицы KpuNch2	PAY_FR_KPUNCH2(Kpu_Tn, Date)	Заполнение таблицы "Постоянные начисления работника" KpuNch2 Kpu_Tn - табельный номер работника Date - дата, на которую рассчитывается стаж и суммы постоянных начислений
18	Заполнение формы 1Т	PAY_FR_P4QRT(нача ло, конец, признаки)	
19	Исходящее сальдо на дату	PAY_GET_SLDOOUT(pDate, pPrz, pSpsRb)	Получить исходящее сальдо на дату pDate - дата pPrz - 0 - За предприятием, 1 - За работниками pSpsRb - 1 - по списку работников (KpuRcdOtpTot должна быть заполнена Kpu_Rcd работников)
20	Мемориальный ордер № 5 - параметры	PAY_FR_MO5_CFG()	Диалоговое окно параметров отчёта "Мемориальный ордер № 5" Параметры сохраняются в таблицах PayMo5*
21	Определение значения MPOT (RU)	PAY_FR_MMBInRU(Date)	Определение значения минимального размера оплаты труда (MPOT) RU Date - дата, на которую получаем MPOT

22	Определение интервала стажа для больничного	PAY_FR_GETSTJINT(pCdVo, pQtDay, pQtMon, pQtYer, pStjBeg, pStjEnd)	<p>ВХОД:</p> <p>pCdVo - код вида оплаты больничного (Vo_Cd)</p> <p>pQtDay - кол-во дней стажа</p> <p>pQtMon - кол-во месяцев стажа</p> <p>pQtYer - кол-во лет стажа</p> <p>ВЫХОД:</p> <p>pStjBeg - интервал ОТ</p> <p>pStjEnd - интервал ДО</p>
23	Определение страхового тарифа фонда	PAY_FR_GETFNDPRC (pCdFnd, pDatRp, pDatUp)	<p>pCdFnd - Код фонда</p> <p>pDatRp - Расчетный период</p> <p>pDatUp - Учетный период</p>
24	Получение кодов ВО, форм. вид дохода НДФЛ	PAY_GET_VO_VDX(uint2 pVdxCd)	<p>Функция получения кодов вида оплат, формирующих в вид дохода НДФЛ</p> <p>ВХОД:</p> <p>pVdxCd - Вид дохода</p> <p>ВЫХОД:</p> <p>Строковый перечень ВО, разделенных запятой</p>
25	Прилож. 2 Фонда страх. от несч. случаев (UA)	PAY_FR_Fsns(Период, CE, Признак)	<p>Заполнение данных для отчёта "Приложение № 2 Фонда страхования от несчастных случаев"</p> <p>Период - дата отчётного периода</p> <p>CE - список кодов структурных единиц, разделенных запятыми.</p> <p>Признак :</p> <p>0 - за учётный период</p> <p>1 - за расчётный период</p>
26	Экспорт поднаборов в "БЕСТ-Зарплату"	PAY_FILL_BASE_ExpBZ()	<p>Запись поднаборов необходимых для экспорта в "БЕСТ-Зарплату" в сетевые таблицы.</p> <p>Для последующей работы с ними из Фаст Репорта.</p>
	Функции для FR-отчетов (кадры)		

1	Заполнение данных для отчета 142	KDR_FR_REPPM(pDate)	<p>Заполнение данных для отчета 142</p> <p>"Отчет про кадровый состав, принятых и уволенных работников"</p> <p>pDate - период, за который формируется отчет</p>
2	Заполнение данных для формы 1-К	KDR_FR_1K(pDate, pSave)	<p>Заполняет таблицу Kdr1K данными для печати формы 1-К</p> <p>pDate - дата</p> <p>pSave - true - заполнять серверную БД данными для отчета</p>
3	Заполнение данных для формы 1-ПВ (квартальная)	KDR_FR_1PVQ(УП, П, ПП, СЕ, П)	<p>Заполняет таблицу Kdr1PvQ данными для отчета.</p> <p>дата УП- период отчета. (учетный период)</p> <p>П - 1 по подразделениям, 0 - по предприятию</p> <p>ПП - 1 - включать подчиненные подразделения</p> <p>СЕ - список кодов структурных единиц, разделенных запятыми (SysSte_Cd)</p> <p>П - признак - один отчет на выбранные структурные единицы</p> <p>Отчет формируется за период с начала года</p>

4	Заполнение данных для формы 1-ПВ (месячная)	KDR_FR_1PVM(дата УП, П, ПП, СЕ, П)	<p>Заполняет таблицу Kdr1PvM данными для отчета.</p> <p>дата УП- период отчета. (учетный период)</p> <p>П - 1 по подразделениям, 0 - по предприятию</p> <p>ПП - 1 - включать подчиненные подразделения</p> <p>СЕ - список кодов структурных единиц, разделенных запятыми (SysSte_Cd)</p> <p>П - признак - один отчет на выбранные структурные единицы</p> <p>Отчет формируется за период с начала года</p>
5	Заполнение данных для формы 2-К	KDR_FR_2K(pDate, pSave)	<p>Заполняет таблицу Kdr2K данными для печати формы 2-К</p> <p>pDate - дата</p> <p>pSave - заполнять серверную БД данными для отчета</p>
6	Заполнение данных для формы 3-ПВ (полугодовая)	KDR_FR_3PV(УП, П, ПП, СЕ, П)	<p>Заполняет таблицу Kdr3Pv данными для отчета 3 - ПВ (полугодовая).</p> <p>дата УП- период отчета. (учетный период)</p> <p>П - 1 по подразделениям, 0 - по предприятию</p> <p>ПП - 1 - включать подчиненные подразделения</p> <p>СЕ - список структурных единиц, разделенных запятыми (SysSte_Cd)</p> <p>П - признак - один отчет на выбранные структурные единицы</p> <p>Отчет формируется за период с начала года</p>
7	Заполнение данных для формы 6-ПВ	KDR_FR_6PV(Дата, признак(0,1))	<p>Дата - дата формирования отчета</p> <p>Признак - 1 - по отмеченным записям реестра, 0 - по всем</p> <p>Заполняет таблицу данными для печати формы 6-ПВ</p>

8	Заполнение данных для формы 9-ДС	KDR_FR_9DS('Дата начала периода','Дата окончания периода', Признак формирования	<p>Заполняет таблицу Kdr9Ds данными для печати формы 9-ДС</p> <p>Признак = 1 - формировать по структурным единицам, 0 - по предприятию</p> <p>// При формировании по структурным единицам требует заполнения PayRepC :</p> <p>// PayRepC_CdTbl = 1812</p> <p>// PayRepC_IdRep = 15</p> <p>// PayRepC_CdUsr = BPRO.USER_RCD.Value</p> <p>// PayRepC_CdPrm = 1</p> <p>// PayRepC_Cd = 1</p> <p>// в PayRepC_CdPar писать CA структурной единицы</p> <p>// в PayRepC_Nm - код структурной единицы</p>
9	Заполнение фактических дней/часов по работникам	KDR_FR_GETFDAYS(pDtUP, pPdr, pIncSubPd, pByKpuRcd, pNotPayCalc)	<p>Заполняет фактические дни/часы по работникам по учетному периоду (таблица KdrTbl_FDays)</p> <p>pDtUP - Учетный период</p> <p>pPdr - Признак: 1 - по подразделениям, 0 - по предприятию</p> <p>pIncSubPd - Признак: 1 - включать подчиненные подразделения, 0 - не включать подчиненные подразделения</p> <p>pByKpuRcd - Признак: 1 - по списку работников, 0 - нет</p> <p>pNotPayCalc - Признак: 1 - не выполнять расчёт зарплаты для текущего УП - табель не содержит</p> <p>автоматически рассчитанных табелируемых видов опла</p>

10	Заполнение фактического табеля	KDR_FR_GETFTBL(pDtUP, pPdr, pIncSubPd, pByKpuRcd, pNotPayCalc)	<p>Заполняет фактический табель по учетному периоду (таблица KdrGetFTbl)</p> <p>pDtUP - Учетный период pPdr - Признак: 1 - по подразделениям, 0 - по предприятию pIncSubPd - Признак: 1 - включать подчиненные подразделения, 0 - не включать подчиненные подразделения pByKpuRcd - Признак: 1 - по списку работников, 0 - нет pNotPayCalc - Признак: 1 - не выполнять расчёт зарплаты для текущего УП - табель не содержит</p> <p>автоматически рассчитанных табелируемых видов оплат текущего</p>
11	Заполнить таблицу интервалов Лицевого счета	KDR_FILL_ILS(Дата начала, Дата конца, Код реквизита)	Заполняет таблицу KdrILS интервалами изменения лицевого счета по предприятию.
12	Надбавки, используемые в штатном расписании	KDR_FR_GETVOST()	Возвращает строку с кодами видов оплат, через запятую
13	Экспорт личных карточек по категориям должностей	KDR_FR_EXPLK(pMrk, pPath, pPrz, pMrkSE)	<p>Экспортирует данные личных карточек работников по категориям должностей по указанному пути</p> <p>pMrk - Строка кодов отмеченных категорий должностей pPath - Путь к файлам экспорта pPrz - Признак: 0 - экспорт по предприятию, 1 - экспорт по структурным единицам pMrkSE - Строка кодов отмеченных структурных единиц</p>
	Функции для валютных счетов		

	Устаревшая реализация		
1	Переоценка валютных счетов	RECALC_CURBAL()	Функция переоценки всех валютных счетов на дату документа с формированием проводок.
2	Переоценка по ежедневной смене курса	RECALC_CURBL_D()	Функция переоценки всех валютных счетов на дату документа с формированием проводок. Курсовая разница вычисляется как разность курсов на указанную и предыдущую даты, умноженную на остаток на указанную дату.
3	Расчёт баланс. стоимости валюты прихода	CALC_BVAL_IN("Счёт",ID_карточки,Дата,"КодВалюты")	Функция расчёта балансовой стоимости валюты прихода за указанную дату
4	Расчёт баланс. стоимости валюты расхода	CALC_BVAL_OUT("Счёт",ID_карточки,Дата,"КодВалюты")	Функция расчёта балансовой стоимости валюты расхода за указанную дату
5	Расчёт баланс. стоимости валюты сальдо	CALC_BVAL_SLD("Счёт",ID_карточки,Дата,"КодВалюты")	Функция расчёта балансовой стоимости валюты сальдо на начало дня указанной даты
6	Расчёт курсовой разницы по ежедневной смене курса	DIF_RATE_D("Счёт",ID_карточки,Дата)	Функция расчёта суммы курсовой разницы по указанному валютному счёту и аналитике на указанную дату. Курсовая разница вычисляется как разность курсов на указанную и предыдущую даты, умноженную на остаток на указанную дату.
7	Расчёт курсовой разницы по счёту	DIF_RATE("Счёт",ID_карточки,Дата)	Функция расчёта суммы курсовой разницы по указанному валютному счёту и аналитике на указанную дату.

8	Расчёт средневзвешенного курса	CALC_BVAL_CURS("Счёт", ID_карточки, Дата, "КодВалюты")	Функция расчёта средневзвешенного курса, рассчитанного по балансовой стоимости валюты на заданную дату.
9	Расчёт среднего курса	CALC_AVE_CURS(Дата, "КодВалюты", "КодКотировки")	Функция расчёта значения среднего курса валюты по указанной котировке за период с даты начала первого открытого периода по указанную дату включительно.
Функции для валютный счетов			
1	Переоценка валютных счетов	OSB_CURBAL()	Функция переоценки всех валютных счетов на дату документа с формированием проводок.
2	Переоценка валютных счетов	OSB_CURBAL2(РЕЖИМ)	Функция переоценки всех валютных счетов на дату документа с формированием проводок. Режим расчета 0 - по курсу валюты на дату проводки 1 - по курсу валюты следующего дня за датой проводки
3	Расчёт курсовой разницы по счёту	OSB_DIF_RATE2("Счёт", ID_карточки, Дата, "Вал", РЕЖИМ)	Функция расчёта суммы курсовой разницы по указанному валютному счёту, аналитике и валюте на указанную дату. Режим расчета 0 - по курсу валюты на дату проводки 1 - по курсу валюты следующего дня за датой проводки

4	Расчёт курсовой разницы по счёту	OSB_DIF_RATE("Счёт", ID_карточки, Дата, "Вал")	Функция расчёта суммы курсовой разницы по указанному валютному счёту, аналитике и валюте на указанную дату.
Функции для вычисления расхождений в уч. запасов			
1	Функция вычисл. расхожд. по пост. запасов за пер.	SKL_CALCDIFZK(ДатаНач, ДатаКон, Вид расхожнения)	Функция вычисляет расхождения по поступлению запасов за период Вид равен 1 расх.вычисляется по стоимости закупки Вид равен 0 расх.вычисляется по учетной стоимости
2	Функция вычисл. расхожд. по спис. запасов за пер.	SKL_CALCDIFSEB(Шифр склада, ДатаНач, ДатаКон)	Функция вычисляет расхождения по списанию запасов за период
Функции для полей и справочников пользователя			
1	Дать значение из справочника по ключу	UF_GET_SPR_VAL(Код,Поле)	Возвращает значение из справочника по заданному ключу. "Код" - номер справочника, "Поле" - ключ. Пример: UF_GET_SPR_VAL(1,U_1)
2	Дать значение из справочника по ключу (стр.)	UF_GET_SPR_VAS(Код,Поле)	Возвращает строковое значение из справочника по заданному ключу. "Код" - номер справочника, "Поле" - ключ. Пример: UF_GET_SPR_VAS(1,U_1)
3	Дать значение из справочника по полю	UF_GET_FLD_VAL(Поле)	Возвращает значение по заданному полю пользователя, к которому привязан справочник. Пример: UF_GET_FLD_VAL(U_1)
4	Дать значение из справочника по полю (стр.)	UF_GET_FLD_VAS(Поле)	Возвращает строковое значение по заданному полю пользователя, к которому привязан справочник. Пример: UF_GET_FLD_VAS(U_1)

5	Дать первое значение из справочника	UF_GET_SPR_FST(Код,Ключ,Значение)	Возвращает реквизиты первой записи справочника. "Код" - номер справочника,"Ключ" - поле ключа, "Значение" - поле значения. Пример: UF_GET_SPR_FST(1,KEY_1,VAL_1)
6	Дать первую запись реестра по коду реестра	UF_RSTFIRST(<Код реестра>)	Устанавливает реестр пользователя с указанным кодом на первую запись Возвращает 0 - ошибка, иначе - успешно
7	Дать последнее значение из справочника	UF_GET_SPR_LST(Код,Ключ,Значение)	Возвращает реквизиты последней записи справочника. "Код" - номер справочника,"Ключ" - поле ключа, "Значение" - поле значения. Пример: UF_GET_SPR_LST(1,KEY_1,VAL_1)
8	Дать последнюю запись реестра по коду реестра	UF_RSTLAST(<Код реестра>)	Устанавливает реестр пользователя с указанным кодом на последнюю запись Возвращает 0 - ошибка, иначе - успешно
9	Дать предыдущую запись реестра по коду реестра	UF_RSTPREV(<Код реестра>)	Устанавливает реестр пользователя с указанным кодом на предыдущую запись Возвращает 0 - ошибка, иначе - успешно
10	Дать предыдущее значение из справочника	UF_GET_SPR_PRV(Ключ,Значение)	Возвращает реквизиты предыдущей записи справочника. "Ключ" - поле ключа,"Значение" - поле значения. Пример: UF_GET_SPR_PRV(KEY_1,VAL_1)

11	Дать следующее значение из справочника	UF_GET_SPR_NXT(Ключ,Значение)	Возвращает реквизиты следующей записи справочника. "Ключ" - поле ключа,"Значение" - поле значения. Пример: UF_GET_SPR_NXT(KEY_1,VAL_1)
12	Дать следующую запись реестра по коду реестра	UF_RSTNEXT(<Код реестра>)	Устанавливает реестр пользователя с указанным кодом на следующую запись Возвращает 0 - ошибка, иначе - успешно
13	Добавить новую запись реестра по во временую табл	UF_TMP_RSTPUT(<Код реестра>)	Добавить новую запись реестра по коду реестра только во временную таблицу Возвращает 0 - ошибка, иначе - успешно
14	Добавить новую запись реестра по коду реестра	UF_RSTPUT(<Код реестра>)	Добавить новую запись реестра по коду реестра Возвращает 0 - ошибка, иначе - успешно
15	Добавить новую запись реестра по коду реестра и ко	UF_RSTPUTRCD(<Код реестра>, <Идентификатор записи картотеки>)	Добавить новую запись реестра по коду реестра и коду записи картотеки Возвращает 0 - ошибка, иначе - успешно
16	Записать значение в справочник по ключу	UF_PUT_SPR_VAL(Код,Поле,Значение)	Записывает в справочник значение по заданному ключу. "Код" - номер справочника, "Поле" - ключ. Пример: UF_PUT_SPR_VAL(1,U_1,"ABC")
17	Записать значение в справочник по полю	UF_PUT_FLD_VAL(Поле,Значение)	Записывает в справочник значение по заданному полю пользователя, к которому привязан справочник. Пример: UF_PUT_FLD_VAL(U_1,"ABC")

18	Записать значение поля	UF_WRITEFIELD(Имя таблицы, Имя поля, Код (CA) записи таблицы, Значение поля)	<p>Сохраняет значение поля пользователя для записи картотеки</p> <p>Имя таблицы - наименование таблицы картотеки</p> <p>Имя поля - наименование поля пользователя</p> <p>Код (CA) записи таблицы - идентификатор записи картотеки</p> <p>Значение поля - сохраняемое значение поля пользователя</p> <p>Возвращает 0 - ошибка, иначе запись сохранена успешно</p> <p>Пример: UF_WRITEFIELD("SprDol", "U_Val", 1, 1.10)</p>
19	Зачитать значение поля	UF_READFIELD(Имя таблицы, Имя поля, Код (CA) записи таблицы)	<p>Возвращает значение поля пользователя для записи картотеки</p> <p>Имя таблицы - наименование таблицы картотеки</p> <p>Имя поля - наименование поля пользователя</p> <p>Код (CA) записи таблицы - идентификатор записи картотеки</p> <p>Пример: z = UF_READFIELD("SprDol", "U_Val", 1)</p>
20	Зачитать значения полей	UF_READ()	<p>Зачитывает значения всех полей пользователя картотек, соответствующих подключенным ресурсам.</p>
21	Изменить текущую запись реестра по коду реестра	UF_RSTCHANGE(<Код реестра>)	<p>Изменить текущую запись реестра по коду реестра</p> <p>Возвращает 0 - ошибка, иначе - успешно</p>
22	Изменить текущую запись реестра по коду реестра во	UF_TMP_RSTCHANGE(<Код реестра>)	<p>Изменить текущую запись реестра по коду реестра только во временной таблице</p> <p>Возвращает 0 - ошибка, иначе - успешно</p>

23	Найти запись реестра по ключу	UF_RSTSEEK(<Код реестра>, <Вид ключа>)	Выполняет поиск в реестре пользователя с указанным кодом по ключу, где <Вид ключа> 0 - искать по уникальному ключу реестра, 1 - искать по ключу сортировки реестра Возвращает 0 - ошибка, иначе - успешно
24	Очистить справочник	UF_DSTR_SPR(Код)	Очищает справочник с номером "Код".
25	Очистить справочник по полю	UF_DSTR_SPRFLD(Поле)	Очищает справочник по заданному полю пользователя, к которому привязан справочник. Пример: UF_DSTR_SPRFLD(U_1)
26	Сортировать реестр по ключу	UF_RSTSORT(<Код реестра>, <Вид ключа>)	Сортирует реестр пользователя с указанным кодом по ключу, где <Вид ключа> 0 - сортировать по уникальному ключу реестра, 1 - сортировать по ключу сортировки реестра Возвращает 0 - ошибка, иначе - успешно
27	удалить запись реестра во временном наборе	UF_TMP_RSTDELETE (<Код реестра>)	Удалить запись реестра во временном наборе только из временной таблицы Возвращает 0 - ошибка, иначе - успешно
28	Удалить значение из справочника по ключу	UF_DEL_SPR_VAL(Код, Поле)	Удаляет из справочника значение по заданному ключу. "Код" - номер справочника, "Поле" - ключ. Пример: UF_DEL_SPR_VAL(1,U_1)

29	Удалить значение из справочника по полю	UF_DEL_FLD_VAL(Поле)	Удаляет из справочника значение по заданному полю пользователя, к которому привязан справочник. Пример: UF_GET_FLD_VAL(U_1)
30	Удалить значение поля	int UF_DELETEFIELD (Имя таблицы, Имя поля, Код (CA) записи таблицы, Значение пол)	Удаляет значение поля. Имя таблицы - наименование таблицы картотеки Имя поля - наименование поля пользователя Код (CA) записи таблицы - идентификатор записи картотеки Значение поля - сохраняемое значение поля пользователя
31	Удалить текущую запись реестра по коду реестра	UF_RSTDELETE(<Код реестра>)	Удалить текущую запись реестра по коду реестра Возвращает 0 - ошибка, иначе - успешно
Функции для работы с автотранспортом			
1	Вывод строки прот. программы опр. стоимости услуг	ATPSRV_HINT("СТРОКА ПРОТОКОЛА")	Выводит строку протокола программы подсчета стоимости услуг в модуле "Учет транспортных услуг" ВОЗВРАТ: 1-удачно/0-нет
2	Записать дополнительный статус автомобиля на дату	ATPWRK_SET_DSTT (ID_Автомобиля, Дата, "Код_Статуса")	Записывает строковый код дополнительного статуса автомобиля в модуле "Учет работы автотранспорта" на указанную дату. При успешном выполнении возвращает единицу. Если код указанного статуса отсутствует в справочнике, возвращает ноль.

3	Записать статус автомобиля на дату	ATPWKR_SET_STT (ID_Автомобиля, Дата, "Код_Статуса")	Записывает строковый код статуса автомобиля в модуле "Учет работы автотранспорта" на указанную дату. При успешном выполнении возвращает единицу. Если код указанного статуса отсутствует в справочнике, возвращает ноль.
4	Зачитать дополнительный статус автомобиля на дату	ATPWKR_GET_DSTT (ID_Автомобиля, Дата)	Возвращает строковый код дополнительного статуса автомобиля из модуля "Учет работы автотранспорта" на указанную дату.
5	Зачитать реквизиты настроек	ATPNST_GET_RK (Код_Реквизита)	Возвращает значения полей из настроек по коду реквизита. Коды реквизитов: 0 - Среднемесячное количество рабочих часов (Управление персоналом - Учет кадров - Настройка)
6	Зачитать справочник тарифных коэффициентов	ATP_STK_TARIF ("Код", Дата, Квалиф_категория, Режим_Вызова)	Зачитывает из справочника тарифных коэффициентов (Подсистема "Управление персоналом, Кадры") значение реквизита согласно Режиму_Вызова: 1 - Мин. Коэффициент 2 - Макс. Коэффициент 3 - Мин. Оклад 4 - Макс.Оклад
7	Зачитать статус автомобиля на дату	ATPWKR_GET_STT (ID_Автомобиля, Дата)	Возвращает строковый код статуса автомобиля из модуля "Учет работы автотранспорта" на указанную дату.

8	Получить данные талона заказчика по путевому листу	ATP_GETTLN("Номер ПЛ", ДатПЛ)	Функция заполняет данные талона заказчика по путевому листу. Возвращает: 1-талон заказчика найден, 0-не найден
9	Получить кол. выполненных ТКм автомобилем за период	ATP_GETTKM_ALL("Гос.номер", ДатНач, ДатОконч)	Функция возвращает количество ТКм, выполненных автомобилем за период ДатНач..ДатОконч.
10	Получить кол. дней по статусу автомобиля	ATP_GETDAY_STT("Гос.номер", ДатНач, ДатОконч, Статус)	Функция возвращает количество дней автомобиля, имеющего Статус (0 - Эксплуатация, 1 - Ремонт, 2 - ТО-1, 3 - ТО-2, 4 - Аренда) за период ДатНач..ДатОконч.
11	Получить конечный остаток топлива на дату	ATP_FUE_END("Гос.номер", Дата, Тип_топлива)	Функция возвращает конечный остаток топлива автомобиля с номером "Гос.номер" на указанную дату. Тип_топлива - номер типа топлива (1 или 2)
12	Получить начальный остаток топлива на дату	ATP_FUE_BEG("Гос.номер", Дата, Тип_топлива)	Функция возвращает начальный остаток топлива автомобиля с номером "Гос.номер" на указанную дату. Тип_топлива - номер типа топлива (1 или 2)
13	Получить общий пробег автомобиля за период	ATP_GETPRB_ALL("Гос.номер", ДатНач, ДатОконч)	Функция возвращает общий пробег автомобиля за период ДатНач..ДатОконч.
14	Получить оплату работы водителей по путевому листу	ATP_GETSMPAY("Номер ПЛ", ДатПЛ)	Функция возвращает сумму оплаты работы водителей по путевому листу.
15	Получить отработанное время автомобиля за период	ATP_GETTIM_ALL("Гос.номер", ДатНач, ДатОконч)	Функция возвращает отработанное время автомобиля за период ДатНач..ДатОконч.

16	Получить пробег автомобиля с грузом за период	ATP_GETPRB_WGT("Гос.номер", ДатНач, ДатОконч)	Функция возвращает пробег автомобиля с грузом за период ДатНач..ДатОконч.
17	Получить сумму оказанных услуг по путевому листу	ATP_GETSMSRV("Номер ПЛ", ДатПЛ)	Функция возвращает сумму оказанных услуг по путевому листу.
18	Получить тариф по таблице тарифов на услуги	ATP_TARIF("Шифр", ГОР, ВЕРТ)	Функция возвращает тариф по указанным шифру таблицы услуг и значениям горизонтальной, вертикальной шкал
	Функции для работы с аналитикой		
	Функции доступа к данным картотек		
1	Получить ID картотеки счета	ANLCRD_GETSANL("Счет")	Функция возвращает ID аналитической картотеки счета.
2	Получить количество уровней картотеки	ANLCRD_LCNT(ИД_Картотеки)	Функция возвращает количество уровней указанной картотеки. В случае ошибки возвращается 0.
3	Получить количество уровней картотеки счета	ANLCRD_LCNTSCH("Номер_Счета")	Функция возвращает количество уровней картотеки указанного счета. В случае ошибки возвращается 0.
	Функции доступа к данным карточки		
1	Обновить карточку в аналитической картотеке	ANLCRD_UPDATE_CDNM ("КОД Справочника", "КОД в справочнике")	Обновить карточку в аналитической картотеке Функция работает только с данными аналитического справочника.
2	Очистить кеш аналитической картотеки	ANLCRD_CLEAR_CACHE()	Очистить кеш аналитической картотеки

3	Получить дополнит. код на уровне аналит. карточки	ANLCRD_GET_DCD(ID_карточки, НомерУровня)	Функция возвращает дополнительный код на указанном уровне аналитики указанной карточки. Значение дополнительного кода аналитики определяется типом реквизита на уровне.
4	Получить код на уровне аналит. карточки	ANLCRD_GET_CD(ID_карточки, НомерУровня)	Функция возвращает код на указанном уровне аналитики указанной карточки. Номер уровня аналитики принимает значение от 1 до 5 включительно.
5	Получить наименование на уровне аналит. карточки	ANLCRD_GET_NM(ID_карточки, НомерУровня)	Функция возвращает наименование на указанном уровне указанной аналитической карточки.
6	Получить строку аналитики	ANLCRD_GET_STR(ID_карточки)	Функция возвращает для заданной аналитической карточки строку в формате: "Код1.Код2.Код3.Код4.Код5...КодN Наименование N"
7	Получить тип реквизита на уровне аналит. карточки	ANLCRD_GET_TP(ID_карточки, НомерУровня)	Функция возвращает код типа реквизита на указанном уровне указанной аналитической карточки.
8	Получить уровень аналитической карточки	ANLCRD_GET_LEV(ID_карточки)	Функция возвращает номер уровня заданной аналитической карточки.
9	Получить уровень аналитической карточки картотеки	ANLCRD_GETLVCR(ИД_Картотеки, ИД_Карточки)	Функция возвращает уровень указанной карточки картотеки. В случае ошибки будет возвращен 0.
Функции поиска и навигации			

1	Выбрать карточку (по картотеке)	ANLCRD_C_SEL(ID_картотеки)	Функция выводит окно выбора аналитических карточек из указанной картотеки. Функция возвращает ID выбранной аналитической карточки или 0.
2	Выбрать карточку (по счету)	ANLCRD_SELECT("Счёт")	Функция выводит окно выбора аналитических карточек из картотеки, привязанной к указанному счету. Функция возвращает ID выбранной аналитической карточки или 0.
3	Найти карточку или создать её (по картотеке)	ANLCRD_C_CRT(ID_картотеки,"K1","K2","K3","K4","K5","K6","K7","K8","K9","K10")	Функция возвращает ID аналитической карточки, состоящей из комбинации указанных кодов. В случае отсутствия такой карточки будет произведена попытка её создания.
4	Найти карточку или создать её (по счёту)	ANLCRD_CREATE("Счёт","K1","K2","K3","K4","K5","K6","K7","K8","K9","K10")	Функция возвращает ID аналитической карточки, состоящей из комбинации указанных кодов. В случае отсутствия такой карточки будет произведена попытка её создания.
5	Получить ID карточки (по картотеке)	ANLCRD_C_ID(ID_картотеки,"K1","K2","K3","K4","K5","K6","K7","K8","K9","K10")	Функция возвращает ID аналитической карточки, состоящей из комбинации указанных кодов.
6	Получить ID карточки (по счёту)	ANLCRD_ID("Счёт","K1","K2","K3","K4","K5","K6","K7","K8","K9","K10")	Функция возвращает ID аналитической карточки, состоящей из комбинации указанных кодов.
7	Получить ID первой карточки в картотеке	ANLCRD_C_FRST(ID_картотеки,НомерЦикла или 0)	Функция возвращает ID первой аналитической карточки в картотеке.

8	Получить ID первой карточки в картотеке	ANLCRD_C_LFST(ID_картотеки,НомерЦикла или 0)	Функция возвращает ID первой аналитической карточки в картотеке. При переборе карточек будут учитываться карточки всех уровней.
9	Получить ID первой карточки в картотеке	ANLCRD_C_CDFRS(ID_Картотеки,Номер_цикла_или_0,"K1","K2","K3","K4","K5","K6","K7	Функция возвращает ID первой аналитической карточки в картотеке, соответствующей переданным кодам аналитики. Код аналитики на каждом уровне может быть заполнен или не заполнен ("").
10	Получить ID первой карточки на счете	ANLCRD_CDFIRST("Счет",Ном_цикла_или_0,"K1","K2","K3","K4","K5","K6","K7","K8",	Функция возвращает номер первой аналитической карточки на счете, соответствующей переданным кодам аналитики. Код аналитики на каждом уровне может быть заполнен или не заполнен ("").
11	Получить ID первой карточки на счёте	ANLCRD_FIRST("Счёт",НомерЦикла или 0)	Функция возвращает ID первой аналитической карточки на счёте.
12	Получить ID первой карточки на счёте	ANLCRD_LFIRST("Счёт",НомерЦикла или 0)	Функция возвращает ID первой аналитической карточки на счёте. При переборе карточек будут учитываться карточки всех уровней.
13	Получить ID следующей карточки	ANLCRD_CDNEXT(Номер_цикла_или_0)	Функция возвращает ИД следующей аналитической карточки или 0. Эта функция работает в паре с функциями ANLCRD_CDFIRST или ANLCRD_C_CDFRS.

14	Получить ID следующей карточки на счёте	ANLCRD_NEXT(НомерЦикла или 0)	Функция возвращает ID следующей аналитической карточки на счёте. Эта функция работает в паре с функциями ANLCRD_FIRST или ANLCRD_LFIRST.
15	Получить номер уровня карточки	ANLCRD_CDLVL(Номер_цикла_или_0)	Функция возвращает номер уровня аналитической карточки. Эта функция работает в паре с функциями ANLCRD_CDFIRST, ANLCRD_C_CDFRS или ANLCRD_CDNEXT.
	Функции для работы с ведомостью себестоимости		
1	Рассчитать ведомость себестоимости	CALC_SEB(<Дата>,<Статус>)	Дата - должна попадать в период рассчитываемой ведомости Статус - 0- не закрывать ведомость, 1-закрывать ведомость
	Функции для работы с договорами		
1	Дать первую операцию Договора	SETOPR_FIRST(ID Договора)	Функция позиционирует на первую операцию Договора. Возвращает 1 в случае успеха 0 - в противном случае.
2	Дать следующую операцию Договора	SETOPR_NEXT()	Функция позиционирует на следующую операцию Договора. Возвращает 1 в случае успеха, 0 - в противном случае.
3	Заполнить поля операции договора по калькуляции	SETOPRIDCALC_VAL (ИД Операции)	Функция заполняет поля операции договора по заданному ИД оперции.
4	Зачитать поля операции Договора	SETOPRID_VAL (ИД Операции)	Функция зачитывает значение группы полей операции по заданному ИД оперции.

5	Зачитать поля операции Договора	SETOPRF_VALUES()	Функция зачитывает значения группы полей DOGOPR_...
6	Получит вид деятельности по договору	GETDOG_VID (ИД договора)	Возвращает вид деятельности по договору.
7	Получить ID аналитической картотеки	GETDOG_ANLKRT(ИД договора)	Функция возвращает ИД аналитической картотеки.
8	Получить ID аналитической карточки	GETDOG_IDANL (ИД договора)	Функция возвращает ИД аналитической карточки.
9	Получить внешний номер Договора	GETDOG_EXTNMR(ИД Договора)	Функция возвращает внешний номер Договора.
10	Получить дату договора	GETDOG_DAT(ИД договора)	Функция возвращает дату договора.
11	Получить дату начала Договора	GETDOG_DATBEG(ИД Договора)	Функция возвращает дату начала Договора.
12	Получить дату окончания Договора	GETDOG_DATEND(ИД Договора)	Функция возвращает дату окончания Договора.
13	Получить запись Договора	SETDOG_POS(ИД Договора)	Функция позиционирует на запись Договора.
14	Получить значение реквизита предмета договора	GETDOG_PRD(ID Предмета договора, ID Реквизита)	Функция возвращает значение реквизита Предмета договора, где ID Реквизита 1 - площадь; 2 - срок; 3 - адрес.
15	Получить ИД Договора	GETDOG_ID("Вид деят-ти", "Код журнала", "Номер,Дата")	Функция возвращает ID Договора.
16	Получить код валюты Договора	GETDOG_VALCD(ИД Договора)	Функция возвращает код валюты Договора.
17	Получить код или наименование журнала по Договору	GETDOG_JRN (ИД договора, Признак)	Получить код или наименование журнала по Договору Признак = 0 код журнала Признак = 1 наименование журнала

18	Получить код контрагента по Договору	GETDOG_PTNC(ИД Договора)	Функция возвращает код контрагента по Договору.
19	Получить код центра ответственности Договора	GETDOG_COCD(ИД Договора)	Функция возвращает код центра ответственности Договора.
20	Получить комментарий по Договору	GETDOG_CMT(ID Договора)	Функция возвращает комментарий по Договору.
21	Получить наименование центра ответственности	GETDOG_CO(ИД Договора)	Функция возвращает наименование центра ответственности.
22	Получить номер договора	GETDOG_NMR(ИД дговора)	Функция возвращает номер договора.
23	Получить объект сделки Договора	GETDOG_OBJ(ИД Договора)	Функция возвращает объект сделки Договора.
24	Получить первый договор по контрагенту	DOG_KAG_FIRST(Код контрагента)	Функция возвращает ИД первого договора по контрагенту.
25	Получить сальдо по договору	GETDOG_SALDO(ИД договора,КОД валюты,КОД котировки,ДАТА начала,ДАТА окончания)	Заполняет поля DOGSUM_... Возвращает 1 в случае успеха, 0 - в противном случае.
26	Получить сальдо по услуге договора (ЖЕК)	GETDOG_SALDOUSL(ИД договора,ДАТА начала,ДАТА окончания,ИД услуги)	Заполняет поля DOGSUM_... Возвращает 1 в случае успеха, 0 - в противном случае.
27	Получить следующий договор по контрагенту	DOG_KAG_NEXT()	Функция возвращает ИД следующего договора по контрагенту.
28	Получить статус договора	GETDOG_ST (ИД договора, Дата)	Получить статус по Договору на заданную дату. Если Дата равна 0 - функция возвращает текущий статус.
29	Получить сумму Договора	GETDOG_SUM(ID Договора)	Функция возвращает сумму Договора.

30	Получить тип Договора	GETDOG_TYPE(ИД Договора)	Функция возвращает тип Договора.
	Функции для работы с должностями		
1	Получения данных справочника должностей по коду	READ_SPRDOL(Код_должности)	Функция заполняет набор полей должности по ее коду. В случае неудачи возвращает 0.
	Функции для работы с историей карточки		
1	Входной износ на дату	IN_IZN_SM(Дата)	Входной износ ОС на дату
2	Входной налог. износ на дату	IN_IZN_SMN(Дата)	Входной налог. износ ОС на дату
3	Первонач. налог. стоимость на дату	PSTM_SMN(ДАТА)	Первоначальная налог. стоимость ОС на дату
4	Первоначальная стоимость на дату	PSTM_SM(ДАТА)	Первоначальная стоимость ОС на дату
5	СПИ общий на дату	SPI_ALL(ДАТА)	Срок полезного использования общий на дату
6	СПИ общий налоговый на дату	SPIN_ALL(ДАТА)	Срок полезного использования общий налоговый на дату
7	СПИ прошедший на дату	SPI_FAKT(ДАТА)	Срок полезного использования прошедший на дату
8	СПИ прошедший налоговый на дату	SPIN_FAKT(ДАТА)	Срок полезного использования налоговый прошедший на дату
	Функции для работы с историей переоценки		
1	Дооценка/уценка износа на дату	IZN_P_SM(ДАТА)	Дооценка/уценка износа на дату
2	Дооценка/уценка перв.стоим. на дату	PSTM_P_SM(ДАТА)	Дооценка/уценка первоначальной стоимости на дату

	Функции для работы с ком. услугами		
1	Выбор дома	GET_SPRDOMJEK(код_дома, признак_выбора_нескольких_домов)	Выбор дома
2	Дата открытого периода	JO_GETDTPR(код структурной единицы)	Дата открытого периода по коду структурной единицы
	Функции для работы с мемориальными ордерами		
	Управление объектом МО		
1	Очистить объект МО	FLASH_MO_OBJ(ИД_Книги, Дата_Начала, Дата_Конца)	Функция производит сброс кеша согласно переданным параметрам.
2	Получить итоговую сумму мемориального ордера	GET_MO_ISM(ИД_Книги, Порядок_МО, Дата_Начала, Дата_Конца, Вид_Ордера)	<p>Функция возвращает итог мемориального ордера согласно переданным параметрам.</p> <p>Параметры:</p> <p>ИД_Книги - ИД бухг. книги; Порядок_МО - очередность мемориального ордера; Дата_Начала - дата начала расчетов; Дата_Конца - дата окончания расчетов; Вид_Ордера - итоговый вид представления ордера, 0 - МО, 1 - Журнал-Главная</p>
3	Проверка принадлежности проводки	TEST_MO_PRW(ИД_Книги, Дата_Начала, Дата_Конца, ИД_Проводки)	<p>Функция возвращает строковый код ордера, в итоги которого попала указанная проводка.</p> <p>Требует предварительного вызова BUILD_MO_PACK.</p>

4	Создать объект МО	BUILD_MO_PACK(ИД_Книги, Порядок_МО, Дата_Начала, Дата_Конца, Вид_Ордера)	<p>Функция создает объект МО согласно переданным параметрам и возвращает идентификатор доступа.</p> <p>Параметры:</p> <p>ИД_Книги - ИД бухг. книги; Порядок_МО - очередность мемориального ордера; Дата_Начала - дата начала расчетов; Дата_Конца - дата окончания расчетов; Вид_Ордера - итоговый вид представления ордера, 0 - МО, 1 - Журнал-Главная</p>
Функции для работы с налогами			
1	Получить налогообл. сумму налога	GETTAXNSMD("Шифр налога")	Функция возвращает налогооблагаемую сумму по указанному шифру налога.
2	Получить налогообл. сумму по ставке налога	GETTAXNSMRATED("Шифр налога", "Шифр ставки налога")	Функция возвращает налогооблагаемую сумму по шифру и ставке налога.
3	Получить сумму налога	GETTAXSUMD("Шифр налога")	Функция возвращает сумму налога по указанному шифру налога.
4	Получить сумму по ставке налога	GETTAXSUMRATED("Шифр налога", "Шифр ставки налога")	Функция возвращает сумму налога по указанному шифру и ставке налога.
Функции для работы с налогами (документ)			
1	Получить сумму налога по документу	GETDOC_TAXSUM("Шифр налога")	Функция возвращает сумму указанного налога по товарному документу. Использовать в печати реестров товарных документов.

2	Получить сумму налога с товаров/у по документу	GETTAXSUMDOCP ("Шифр налога",Тип)	Функция возвращает сумму указанного налога по документу. Тип = 0 (ТМЦ), 1 (Услуги), 2 (ТМЦ+Услуги)
	Функции для работы с налогами (позиция)		
1	Получить сумму налога по позиции	GETTAXSUMP ("Шифр налога")	Функция возвращает сумму налога с указанным шифром для позиции товарного документа.
2	Получить сумму по ставке налога для позиции	GETTAXSUMRATEP ("Шифр налога", "Шифр ставки налога")	Функция возвращает сумму по указанной ставке налога для позиции товарного документа.
	Функции для работы с объектами учёта		
	Функции доступа к параметрам объектов		
1	Получить код бизнес-процесса	OU_GETCDBPR()	Функция возвращает код бизнес-процесса.
2	Получить наименование объекта учёта	OU_GET_NM(ID_объекта)	Функция возвращает наименование объекта учёта по заданному ID.
	Функции навигации по объектам учёта		
1	Получить ID объекта учёта по номеру счёта	OU_FIND_ON_SCH("НомерСчёта")	Функция возвращает ID объекта учёта по заданному номеру банковского счёта. В случае отсутствия подходящего объекта учёта функция возвращает 0.
2	Получить ID первого объекта учёта	OU_FIRST()	Функция возвращает ID первого объекта учёта.
3	Получить ID следующего объекта учёта	OU_NEXT()	Функция возвращает ID следующего объекта учёта. Требуется предварительного вызова функции OU_FIRST().
	Функции для работы с ордерами		

1	Получить кол-во отгруженн. товара конт-ту за пер.	GETQTSKLDCKAG (1, 2, 3, 4, 5, 7)	Получить кол-во отгруженн. товара конт-ту за пер. 1 - ID Товара(Num), 2 - Группа ТМЦ(Str) 3 - Код склада(Str), 4 - Код контрагента(Str) 5 - Шифр ЕИ(Str), 6 - Дата с., 7 - Дата по.
2	Получить сумму себестоимости по документу основани	GETSUMSKLDOCOS(I D док.основания,Код док.основания)	Функция возвращает сумму по себестоимости по документу основания Коды документов оснований 1 - Накладная, Счет-накладной 6 - Отчет консигнатора
3	Создание товарной позиции ордера	INS_SKL_DOC(ИД ордера, артикул, количество, сумма, комментарий)	Создание товарной позиции ордера
Функции для работы с отчетами Склад			
1	Заполнить набор полей КСУ	FILL_PARAM_KRT(ID карточки)	Функция по ID находит карточку складского учета, в случае нахождения заполняет набор полей, иначе возвращает 0.
2	Получить Rcd текущего ордера	GETCURRENTSKLDCR CD	Функция возвращает Rcd открытого ордера
Функции для работы с периодами ОС			
1	Дата конца периода для указанной даты	DAT_PER_END(Дата)	Дата последнего дня периода, в который входит указанная дата.
2	Дата начала периода для указанной даты	DAT_PER_BEG(Дата)	Дата первого дня периода, в который входит указанная дата.
3	Дата первого периода в подсистеме ОС	DAT_FIRST()	Дата первого периода в подсистеме ОС
4	Дата текущего периода в подсистеме ОС	DAT_CUR()	Дата текущего периода подсистемы ОС

5	Законодательство подсистемы ОС	OSN_CD_LAW()	Возвращает код законодательства: 1 - RU, 2 - UA
6	Интервал в месяцах между датами	DAT_INTERVAL (Дата_1, Дата_2)	Возвращает интервал в месяцах между указанными датами
7	Код структурной единицы (мультифирм.)	OSN_CD_STE()	Возвращает код текущей структурной единицы (для мультифирменного предприятия).
8	Тип периода для указанной даты	TYPE_PER(Дата)	Тип периода (количество месяцев), в который входит указанная дата.
9	Тип предприятия для указанной даты	TYPE_FIRM(Дата)	Тип предприятия для периода, в который входит указанная дата. 0 - Промышленное, 1 - Бюджетное, 2 - Смешанное.
	Функции для работы с Планом счетов		
	Расчетные функции для работы с данными счетов		

1	Добавить параметр расчета	OSB_DATA("Сч", Сч_АНЛ, "СчКОРР", СчКОРР_АНЛ)	<p>Функция добавляет в набор параметров расчета указанные данные.</p> <p>Параметры :</p> <p>"Сч" - счет</p> <p>Сч_АНЛ - аналитика счета</p> <p>"СчКОРР" - коррсчет</p> <p>СчКОРР_АНЛ - аналитика коррсчета</p> <p>Примечание:</p> <p>В параметре "СчКОРР" указывается номер детального корр.счета, для дальнейшего получения оборотов субсчетов основного счета ("Сч") в корреспонденции с конкретным субсчетом.</p> <p>Или «*» , для перебора корр.счетов и получения оборотов основного счета в разрезе всех корр.счетов.</p> <p>В параметр СчКОРР_АНЛ</p>
2	Очистить параметры расчета	OSB_DATA_CLEAR()	<p>Функция очищает буфер параметров расчета.</p>

3	Подготовка объекта расчета	OSB_CALC(Дата_Начала, Дата_Конца, "Валюта")	<p>Функция подготавливает объект расчета в соответствии с установленными параметрами, периодом и валютой для фильтрации.</p> <p>В случае удачной подготовки возвращается номер объекта. На основе этого номера формируют свои результаты другие функции в этом разделе.</p> <p>Параметры :</p> <p>Дата_Начала - начало периода расчета Дата_Конца - конец периода расчета "Валюта" - валюта для фильтрации</p> <p>Примечание : установка параметров производится функцией OSB_DATA.</p>
---	----------------------------	--	---

4	Подготовка объекта расчета расширенная	OSB_CALC2(Дата_Начала, Дата_Конца, "Валюта", Парам1, Парам2, Парам3, 0)	<p>Функция подготавливает объект расчета в соответствии с установленными параметрами, периодом и валютой для фильтрации.</p> <p>В случае удачной подготовки возвращается номер объекта. На основе этого номера формируют свои результаты другие функции в этом разделе.</p> <p>Параметры :</p> <p>Дата_Начала - начало периода расчета</p> <p>Дата_Конца - конец периода расчета</p> <p>"Валюта" - валюта для фильтрации</p> <p>Парам1 - получение оборотов и сальдо по верхнему уровню аналитики</p> <p>Парам2 - 0 - фактический учет / 1 - план</p>
---	--	---	---

5	Получить вх. остатки по дебету в базовой валюте	OSB_DT_BSLDBV(Объект, "Сч", Сч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение входящего остатка дебета в базовой валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации Курс - курс отношения национальной валюты к базовой</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p> <p>Параметр Курс дает возможность управления пере</p>
---	---	--	--

6	Получить вх. остатки по кредиту в базовой валюте	OSB_KT_BSLDBV(Объект, "Сч", Сч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение входящего остатка дебета в базовой валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации Курс - курс отношения национальной валюты к базовой</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p> <p>Параметр Курс дает возможность управления пере</p>
7	Получить входящие остатки по дебету	OSB_DT_BSLD(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает значение входящего остатка дебета по заданному счету и аналитике.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации</p> <p>Параметры Сч_АНЛ и "Вал" могут не указываться.</p>

8	Получить входящие остатки по дебету в валюте учета	OSB_DT_BSLDV(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает значение входящего остатка дебета в национальной валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p>
9	Получить входящие остатки по кредиту	OSB_KT_BSLDV(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает значение входящего остатка кредита по заданному счету и аналитике.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации</p> <p>Параметры Сч_АНЛ и "Вал" могут не указываться.</p>

10	Получить входящие остатки по кредиту в вал. учета	OSB_KT_BSLDV(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает значение входящего остатка кредита в национальной валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p>
----	---	---	---

11	Получить исх. остатки по дебету в базовой валюте	OSB_DT_ESLDBV(Объект, "Сч", Сч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение исходящего остатка дебета в базовой валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации Курс - курс отношения национальной валюты к базовой</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p> <p>Параметр Курс дает возможность управления пер</p>
----	--	--	--

12	Получить исх. остатки по кредиту в базовой валюте	OSB_KT_ESLDBV(Объект, "Сч", Сч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение входящего остатка кредита в базовой валюте по заданному счету, аналитике и виду валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации Курс - курс отношения национальной валюты к базовой</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p> <p>Параметр Курс дает возможность управления пер</p>
13	Получить исходящие остатки по дебету	OSB_DT_ESLD(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает исходящие остатки по дебету указанного счета и аналитике.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации</p> <p>Параметры Сч_АНЛ и "Вал" могут не указываться.</p>

14	Получить исходящие остатки по дебету в вал. учета	OSB_DT_ESLDV(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает исходящие остатки по дебету в национальной валюте указанного счета, аналитики и вида валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта</p> <p>Параметр Сч_АНЛ может не указываться.</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p>
15	Получить исходящие остатки по кредиту	OSB_KT_ESLD(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает исходящие остатки по кредиту указанного счета и аналитике.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта фильтрации</p> <p>Параметры Сч_АНЛ и "Вал" могут не указываться.</p>

16	Получить исходящие остатки по кредиту в вал. учета	OSB_KT_ESLDV(Объект, "Сч", Сч_АНЛ, "Вал")	<p>Функция возвращает исходящие остатки по кредиту в национальной валюте указанного счета, аналитики и вида валют.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "Вал" - валюта</p> <p>Параметры Сч_АНЛ, "Вал" могут не указываться.</p> <p>Примечание. Функция используется для детализации валютных остатков по видам, которые имеются на определенном счете и/или аналитической карточке.</p>
17	Получить обороты по дебету	OSB_DT_TURN(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал")	<p>Функция возвращает значение оборота по дебету в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p>

18	Получить обороты по дебету в базовой валюте	OSB_DT_TURNBV(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение оборота в базовой валюте по дебету в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта Курс - курс отношения национальной валюты к базовой</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p> <p>Параметр Курс дает возможность управления переоценкой остатков валют, которые отличаются</p>
19	Получить обороты по дебету в нац. валюте	OSB_DT_TURNV(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал")	<p>Функция возвращает значение оборота в национальной валюте по дебету в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p>

20	Получить обороты по кредиту	OSB_KT_TURN(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал")	<p>Функция возвращает значение оборота по кредиту в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p>
21	Получить обороты по кредиту в базовой валюте	OSB_KT_TURNBV(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал", Курс)	<p>Функция возвращает значение оборота в базовой валюте по кредиту в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта Курс - курс отношения национальной валюты к базовой</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p> <p>Параметр Курс дает возможность управления переоценкой остатков валют, которые отличаютс</p>

22	Получить обороты по кредиту в нац. валюте	OSB_KT_TURNV(Объект, "Сч", Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал")	<p>Функция возвращает значение оборота в национальной валюте по кредиту в заданном отношении.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета "КорСч" - коррсчет КорСч_АНЛ - аналитика коррсчета "Вал" - валюта</p> <p>Параметры Сч_АНЛ, "КорСч", КорСч_АНЛ, "Вал" могут не указываться или указываться частично.</p>
23	Получить первую аналит. карточку по кодам	OSB_ANL_CDFST(Объект, Номер_Цикла, "Сч", "", "", "", "", "", "", "", "")	<p>Функция возвращает ID первой аналитической карточки на счете, соответствующей переданным кодам.</p> <p>Код аналитики на каждом уровне может быть заполнен или не заполнен ("").</p> <p>Следующее значение можно получить с помощью функции OSB_ANL_LNEXT.</p>
24	Получить первую аналитику с указанием уровня	OSB_ANL_LVLFST(Объект, Номер_Цикла, "Сч", НомУровня)	<p>Функция возвращает идентификатор 1-ой аналитической карточки на указанном уровне (с вложенным циклом), по которой есть обороты и сальдо.</p> <p>Следующее значение можно получить с помощью функции OSB_ANL_LNEXT.</p>

25	Получить первую аналитическую карточку	OSB_ANL_FIRST(Объект, "Сч")	<p>Функция возвращает первую аналитическую карточку счета, по которой есть обороты или сальдо.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет</p>
26	Получить первую анл. карточку с уч. цикла	OSB_ANL_LFIRST(Объект, Номер_Цикла, "Сч")	<p>Функция возвращает первую аналитическую карточку счета, по которой есть обороты или сальдо с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет</p> <p>Примечание: Первый номер цикла сканирования равен 1.</p>
27	Получить первую валюту счета	OSB_VAL_FIRST(Объект, "Сч")	<p>Функция возвращает первую валюту счета по которой есть сальдо или обороты.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет</p>

28	Получить первую валюту счета с уч. цикла	OSB_VAL_LFIRST(Объект, Номер_Цикла, "Сч")	<p>Функция возвращает первую валюту счета по которой есть сальдо или обороты с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет</p> <p>Примечания: Первый номер цикла сканирования равен 1.</p>
29	Получить первый корр. счет	OSB_COR_FIRST_ALL (Объект, "Счет", Сч_АНЛ)	<p>Функция возвращает первый корр. счет по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета</p> <p>Параметр Сч_АНЛ может не указываться.</p>
30	Получить первый корр. счет(Детальный)	OSB_COR_FIRST(Объект, "Счет", Сч_АНЛ)	<p>Функция возвращает первый корр. счет по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - детальный бухгалтерский счет Сч_АНЛ - аналитика счета</p> <p>Параметр Сч_АНЛ может не указываться.</p>

31	Получить первый корр.счет с уч. цикла	OSB_COR_LFIRST_AL L(Объект, Номер_Цикла, "Сч", Сч_АНЛ)	<p>Функция возвращает первый корр. счет по указанным параметрам с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет Сч_АНЛ - аналитика счета</p> <p>Примечания: Параметр Сч_АНЛ может не указываться. Первый номер цикла сканирования равен 1.</p>
32	Получить первый корр.счет с уч. цикла(Детальный)	OSB_COR_LFIRST(Объект, Номер_Цикла, "Сч", Сч_АНЛ)	<p>Функция возвращает первый корр. счет по указанным параметрам с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - детальный бухгалтерский счет Сч_АНЛ - аналитика счета</p> <p>Примечания: Параметр Сч_АНЛ может не указываться. Первый номер цикла сканирования равен 1.</p>

33	Получить следующий корр. счет	OSB_COR_NEXT_ALL (Объект, "Счет", Сч_АНЛ)	<p>Функция возвращает следующий корр. счет по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет Сч_АНЛ - аналитика счета</p> <p>Параметр Сч_АНЛ может не указываться.</p>
34	Получить следующий корр. счет(Детальный)	OSB_COR_NEXT(Объект, "Счет", Сч_АНЛ)	<p>Функция возвращает следующий корр. счет по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - детальный бухгалтерский счет Сч_АНЛ - аналитика счета</p> <p>Параметр Сч_АНЛ может не указываться.</p>
35	Получить следующий корр.счет с уч.цикла	OSB_COR_LNEXT_AL L(Объект, Номер_Цикла, "Сч", Сч_АНЛ)	<p>Функция возвращает следующий корр. счет по указанным параметрам с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет Сч_АНЛ - аналитика счета</p> <p>Примечания: Параметр Сч_АНЛ может не указываться. Первый номер цикла сканирования равен 1.</p>

36	Получить следующий корр.счет с уч.цикла(Детальный)	OSB_COR_LNEXT(Объект, Номер_Цикла, "Сч", Сч_АНЛ)	<p>Функция возвращает следующий корр. счет по указанным параметрам с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - детальный бухгалтерский счет Сч_АНЛ - аналитика счета</p> <p>Примечания: Параметр Сч_АНЛ может не указываться. Первый номер цикла сканирования равен 1.</p>
37	Получить следующую аналитическую карточку	OSB_ANL_NEXT(Объект, "Сч")	<p>Функция возвращает следующую аналитическую карточку счета, по которой есть обороты или сальдо.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет</p>
38	Получить следующую анл. карточку с уч. цикла	OSB_ANL_LNEXT(Объект, Номер_цикла, "Сч")	<p>Функция возвращает следующую аналитическую карточку счета, по которой есть обороты или сальдо с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет</p> <p>Примечание: Первый номер цикла сканирования равен 1.</p>

39	Получить следующую валюту счета	OSB_VAL_NEXT(Объект, "Сч")	<p>Функция возвращает следующую валюту счета по которой есть сальдо или обороты.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC "Сч" - счет</p>
40	Получить следующую валюту счета с уч. цикла	OSB_VAL_LNEXT(Объект, Номер_Цикла, "Сч")	<p>Функция возвращает следующую валюту счета по которой есть сальдо или обороты с учетом цикла сканирования.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла сканирования "Сч" - счет</p> <p>Примечания: Первый номер цикла сканирования равен 1.</p>
41	Установить режим сбора данных о проводках	OSB_ENTRY_ON(Объект)	<p>Функция устанавливает режим сбора данных о проводках.</p> <p>Параметры : Объект - объект, подготовленный OSB_CALC</p> <p>Примчание : Режим необходим для работы функций OSB_GET_FENTRY и OSB_GET_NENTRY.</p>
Функции доступа к параметрам счёта			
1	Получить валютность счёта	ACC_GET_VAL("Счёт")	<p>Функция возвращает для указанного счёта значение: 1-мультивалютный счёт, 2-немультивалютный счёт, 0-ошибка.</p>

2	Получить вид счета	ACC_GET_VID("Счет")	Функция возвращает для указанного счета значение : не 0 - вид счета, 0 - ошибка.
3	Получить ИД бухг.книги по строковому коду	ACC_GET_BOOKID("Код_Книги")	Функция возвращает ИД бухгалтерской книги по строковому коду.
4	Получить код Журнала операций	ACC_GET_JOCD("Счет")	Функция возвращает для указанного счета код соответствующего Журнала операций.
5	Получить код книги счёта	ACC_GET_BOOK("Счёт")	Функция возвращает код книги указанного синтетического счёта.
6	Получить наименование Журнала операций	ACC_GET_JONM("Код_Журнала_Операций")	Функция возвращает наименование журнала операций.
7	Получить наименование счёта	ACC_GET_NM("Счёт")	Функция возвращает наименование указанного синтетического счёта.
8	Получить номер родителя указанного счета	ACC_OWNER("Счет")	Функция возвращает строковый номер указанного счета.
9	Получить признак открытости счёта	ACC_GET_OPN("Счёт")	Функция возвращает для указанного счёта значения: 1-консолидирующий, 2-детальный, 0-ошибка.
10	Получить признак переоценки	ACC_GET_PEROC("Счёт")	Функция возвращает признак переоценки: 0 - Нет; 1 - По дебету; 2 - По кредиту.
11	Получить приоритет Журнала операций	ACC_GET_JOPRT("Код_Журнала_Операций")	Функция возвращает приоритет Журнала операций.
12	Получить статью баланса	ACC_GET_STBAL("Счёт")	Функция возвращает статью баланса: 0 - Монетарный; 1 - Не монетарный.

13	Получить тип аналитики счёта	ACC_GET_ANL("Счёт")	Функция возвращает для указанного счёта значение: 1-счёт с аналитикой, 2-счёт без аналитики, 0-ошибка.
14	Получить тип сальдо счёта	ACC_GET_STYPE("Счёт")	Функция возвращает для указанного счёта значение: 1-свёрнутое сальдо, 2-развёрнутое сальдо, 0-ошибка.
15	Получить тип счёта	ACC_GET_TYPE("Счёт")	Функция возвращает тип указанного счёта: 0 - Активный, 1 - Пассивный, 2 - Активно-пассивный, 4 - Забалансовый, 255 - ошибка.
16	Проверить подчинённость субсчёта	ACC_GET_SUB("СубСчёт", "Счёт")	Для указанных параметров функция возвращает значение: 1-"СубСчёт" является субсчётом "Счёт", 2-"СубСчёт" не является субсчётом "Счёт", 0-ошибка.
Функции навигации по Плану счетов			
1		ACC_FIRST(НомерЦикла или 0, КодКниги, "Счёт")	Функция возвращает номер первого счёта из Плана счетов по указанной книге или первого субсчёта по указанному счёту.
2	Получить номер следующего счёта	ACC_NEXT(НомерЦикла или 0)	Функция возвращает номер следующего счёта из Плана счетов. Требуется первоначальный вызов функции ACC_FIRST.
Функции Плана счетов для отчетов			
1	Возвращает ключ записи счёта	ACC_SRCID(Счет)	Возвращает ключ записи указанного счёта. Ключ уникальный.

2	Возвращает количество уровней ветки	ACC_SLEVELS(Счет)	Возвращает количество уровней ветки указанного счета.
3	Возвращает уровень счета	ACC_SLEV(Счет)	Возвращает уровень указанного счета.
4	Дать первый детальный счет ветки	ACC_FIRSTDET(Номер_Цикла_Или_0, "Книга", Счет)	Дать первый детальный счет ветки указанного счета (последнего уровня).
5	Дать первый синт. счет ветки	ACC_FIRSTALL(Номер_Цикла_Или_0, "Книга", Счет)	Дать первый синтетический счет ветки указанного Счета. (Уровня 1).
6	Дать следующий детальный счет ветки	ACC_NEXTDET(Номер_Цикла_Или_0, "Книга", Счет)	Дать следующий детальный счет ветки указанного в ACC_FIRSTDET счета.
7	Дать следующий счет ветки	ACC_NEXTALL(Номер_Цикла_Или_0, "Книга", Счет)	Дать следующий счет в дереве счетов ветки указанного в ACC_FIRSTALL счета.
Функции для работы с Приказами			
1	Добавить надбавку в набор надбавок	KDPRK_NDB_ADD(<код надбавки>, <Процент>, <Сумма>)	Функция добавляет надбавку с указанным кодом Параметры: <код надбавки> - код надбавки в наборе <Процент> - % надбавки <Сумма> - сумма надбавки Проверяется существование надбавки, если найдена, то ничего не делает, иначе добавленная надбавка становится текущей Если значение <Процент> не нулевое то <Сумма> игнорируется
2	Получение основания операции приказа	KDPRK_OPR_OSN()	Получение текста основания операции приказа.
3	Получение основания приказа	KDPRK_GET_OSN()	Получение текста основания приказа.

4	Получение параметров операции	KDPRK_OPR_DATA()	Получение параметров операции. Заполняет поля KDP_OPR_...
5	Получение параметров приказа	KDPRK_PRK_DATA()	Получение параметров приказа. Заполняет поля KDP_PRK_...
6	Получение преамбулы приказа	KDPRK_GET_PRA()	Получение текста преамбулы приказа.
7	Получить код родительского подразделения	KDPRK_PDR_PAR(Код_подразделения)	<p>Функция возвращает код родительского подразделения по коду.</p> <p>Если текущий приказ - на ввод в действие Штатного расписания, то используется справочник подразделений Штатного расписания.</p> <p>Иначе используется системный справочник подразделений. Дата справочника подразделения выбирается из текущей операции приказа.</p>

8	Получить наименование подразделения	KDPRK_PDR_NM(Код_подразделения)	<p>Функция возвращает наименование подразделения по коду.</p> <p>Если текущий приказ - на ввод в действие Штатного расписания, то используется справочник подразделений Штатного расписания.</p> <p>Иначе используется системный справочник подразделений. Дата справочника подразделения выбирается из текущей операции приказа.</p>
9	Получить полное наименование подразделения	KDPRK_PDR_FULL(Код_подразделения, Падеж)	<p>Функция возвращает полное наименование подразделения по коду. Падеж может принимать значения: 0 - Именительный 1 - Родительный 2 - Дательный</p> <p>Если текущий приказ - на ввод в действие Штатного расписания, то используется справочник подразделений Штатного расписания.</p> <p>Иначе используется системный справочник подразделений. Дата справочника подразделения выбирается из текущей операции приказа.</p>

10	Получить тип подразделения по коду	KDPRK_PDR_TYPE(Код_подразделения)	<p>Функция возвращает тип подразделения по коду.</p> <p>Если текущий приказ - на ввод в действие Штатного расписания, то используется справочник подразделений Штатного расписания.</p> <p>Иначе используется системный справочник подразделений. Дата справочника подразделения выбирается из текущей операции приказа.</p>
11	Проверка существования надбавки в наборе надбавок	KDPRK_NDB_EXISTS(<код надбавки>)	<p>Функция проверяет существование надбавки с указанным кодом в наборе надбавок</p> <p>Параметры: <код надбавки> - код надбавки в наборе</p> <p>Проверяется существование надбавки, если найдена, надбавка становится текущей</p>

12	Сортировка операций в приказе	KDPRK_OPR_SORT("Список")	<p>Сортировка операций в приказе.</p> <p>"Список" - это набор из наименований полей сортировки, разделенных пробелом:</p> <p>- Информация по подразделению PDREPORT - № п/п для отчетов PDCD - код PDNM - наименование</p> <p>- Информация по должности DOLREPORT - № п/п для отчетов DOLCD - код DOLNM - наименование</p> <p>- Информация по профессии PRFCD - код PRGNM - наименование</p> <p>- Информация по работнику RABFIO - Фамилия И.О. RABLK - номер ЛК RABTN - табельный номер</p>
13	Установить % надбавки по коду надбавки	KDPRK_NDB_SET_PROS(<код надбавки>, <Процент>)	<p>Функция устанавливает % надбавки по коду надбавки</p> <p>Параметры:</p> <p><код надбавки> - код надбавки в наборе <Процент> - процент надбавки</p> <p>Проверяется существование надбавки, если не найдена, то ничего не делает, иначе измененная надбавка становится текущей</p>
14	Установить запись операции по ID	KDPRK_OPR_ID(ID_операции)	Установить текущую запись операции по ID.
15	Установить первую запись набора надбавок	KDPRK_NDB_FIRS()	Установить первую запись набора надбавок.

16	Установить первую запись набора отпусков	KDPRK_OTP_FIRS()	Установить первую запись набора отпусков.
17	Установить первую запись набора стажей	KDPRK_STG_FIRS()	Установить первую запись набора стажей.
18	Установить первую запись операции	KDPRK_OPR_FIRS()	Установить первую запись операции.
19	Установить предыдущую запись набора надбавок	KDPRK_NDB_PREV()	Установить предыдущую запись набора надбавок.
20	Установить предыдущую запись набора отпусков	KDPRK_OTP_PREV()	Установить предыдущую запись набора отпусков.
21	Установить предыдущую запись набора стажей	KDPRK_STG_PREV()	Установить предыдущую запись набора стажей.
22	Установить предыдущую запись операции	KDPRK_OPR_PREV()	Установить предыдущую запись операции.
23	Установить следующую запись набора надбавок	KDPRK_NDB_NEXT()	Установить следующую запись набора надбавок.
24	Установить следующую запись набора отпусков	KDPRK_OTP_NEXT()	Установить следующую запись набора отпусков.
25	Установить следующую запись набора стажей	KDPRK_STG_NEXT()	Установить следующую запись набора стажей.
26	Установить следующую запись операции	KDPRK_OPR_NEXT()	Установить следующую запись операции.
27	Установить сумму надбавки по коду надбавки	KDPRK_NDB_SET_SUM(<код надбавки>, <Сумма>)	Функция устанавливает сумму надбавки по коду надбавки Параметры: <код надбавки> - код надбавки в наборе <Сумма> - сумма надбавки Проверяется существование надбавки, если не найдена, то ничего не делает, иначе измененная надбавка становится текущей

28	Установить текущую запись операции	KDPRK_OPR_CURR()	Установить текущую запись операции.
29	Формирование основания приказа	KDPRK_PUT_OSN()	Формирование текста основания приказа.
30	Формирование преамбулы приказа	KDPRK_PUT_PRA()	Формирование текста преамбулы приказа.
31	Формирование строки текста приказа	KDPRK_PUT("Текст")	Формирование строки текста приказа.
32	Формирование строки текста приказа+перевод строки	KDPRK_PUTLN("Текст")	Формирование строки текста приказа с добавлением признака перевода строки.
33	Формирование текста приказа	KDPRK_TEXT("Номер", "Дата")	Формирует текст приказа с указанным номером и датой.
34	Формирование текста приказа по ID приказа	KDPRK_TEXT_ID("Id приказа")	Формирует текст приказа с указанным Id.
35	Формирование текста произвольной операции	KDPRK_PUT_ARB()	Включение в текст приказа текста текущей произвольной операции.
36	Функция возвращает Дату приказа назначения работни	KDPRK_TO_GET_NAZNACH_DATE (<Дата>, <Реквизит>)	<p>Функция возвращает Дату приказа назначения работника</p> <p>Параметры:</p> <p><Дата> - дата с которой нужно начинать поиск 0=искать начиная с текущей даты</p> <p><Реквизит> - имя реквизита(имя поля LK_... из "Ощие сведения Личной карточки")</p> <p>Если не найден приказ, то возвращается дата ПРИКАЗА НА ПОСТУПЛЕНИЕ</p> <p>Пример :</p> <p>KDPRK_TO_GET_NAZNACH_DATE (0, "LK_KAT") - возвратит последний приказ в котором была присвоена Категория</p>

37	Функция возвращает Номер приказа назначения работ.	KDPRK_TO_GET_NAZNACH_PRKNUM (<Дата>, <Реквизит>)	<p>Функция возвращает Номер приказа назначения работника</p> <p>Параметры:</p> <p><Дата> - дата с которой нужно начинать поиск 0=искать начиная с текущей даты</p> <p><Реквизит> - имя реквизита(имя поля LK_... из "Ощие сведения Личной карточки")</p> <p>Если не найден приказ, то возвращается номер ПРИКАЗА НА ПОСТУПЛЕНИЕ</p> <p>Пример :</p> <p>KDPRK_TO_GET_NAZNACH_PRKNUM(0, "LK_KAT") - возвратит последний приказ</p> <p>в котором была присвоена Категория</p>
38	Функция возвращает свободный табельный номер начин	KDPRK_GET_FREETN (<код начального номера>)	<p>Функция возвращает свободный табельный номер начиная с указанного начального номера</p> <p>Параметры:</p> <p><код начального номера> - код табельного номера с которого начинается поиск</p> <p>Возврат:</p> <p>Возвращает свободный табельный номер</p>

39	Функция возвращает Текст с указанным стилем шрифта	KDPRK_TEXT_SET_STYLE(<Текст>, <Стиль>)	<p>Функция возвращает Текст с указанным в параметре Стиль стилем шрифта</p> <p>Параметры:</p> <p><Текст> - текст для обработки</p> <p><Стиль> - стиль шрифта может принимать значения</p> <p>0 - обычный</p> <p>1 - жирный</p> <p>2 - наклонный</p> <p>3 - подчеркнутый</p> <p>или вариации</p> <p>12 - жирный + наклонный</p> <p>13 - жирный + подчеркнутый</p> <p>23 - наклонный + подчеркнутый</p> <p>123 - жирный + наклонный</p>
40	Функция возвращает ФИО для текущей ТО в вариациях	KDPRK_TO_GET_FIO(<Текст ФИО>, <Стиль написания>)	<p>Функция возвращает <Фамилию имя отчество> для текущей ТО в вариациях написания.</p> <p>Параметры:</p> <p><Текст ФИО></p> <p>Текст <Фамилия имя отчество></p> <p><Стиль написания></p> <p>0 - Фамилия Имя Отчество</p> <p>1 - ФАМИЛИЯ Имя Отчество</p> <p>2 - ФАМИЛИЯ ИМЯ ОТЧЕСТВО</p>

41	Функция нормализует Должность из Должность+Подразд.	KDPRK_NORMALIZE_DOL(<Должность>, <Подразделение>)	Функция возвращает <Должность>+<Подразделение > без повторов слов. Параметры: <Должность> - наименование должности <Подразделение> - наименование подразделения Пример : «Начальник отдела» + «Отдела информатизации» = «Начальник Отдела информатизации».
	Функции для работы с производственными заказами		
1	Получить статус Производственного заказа	GETPZKSTATUS ("Номер", "Код журнала")	Функция по номеру заказа и по коду журнала возвращает статус производственного заказа: черновик-0, выполняется-1, закрыт-2.
2	Читать реквизиты производственного заказа по ИД	READ_PZK(ИД произв. заказа)	Функция читает реквизиты производственного заказа по ИД. Возвращает True, если чтение реквизитов прошло успешно, иначе – False.
	Функции для работы с профессиями		
1	Получение наименования профессии	PRF_GET_NAME("Код профессии", "Падеж")	Функция возвращает наименование профессии по коду с учетом падежа(0-имен, 1-род, 2-дат, 3-твор, 4-имен.полный).
	Функции для работы с путевыми листами		

1	Записать статус работы водителя для путевого листа	ATPTBL_SET_STT (ID_Путевого_листа, "Код_Статуса")	Записывает строковый код статуса водителя в "Табеле водителей" для Путевого Листа. При успешном выполнении возвращает единицу. Если код указанного статуса отсутствует в справочнике, возвращает ноль.
2	Зачитать ведомость выдачи топлива по ID	ATP_GETVVF_ID (ID_Ведомости)	Зачитывает реквизиты ведомости выдачи топлива по ID ведомости, переданной в качестве параметра. В случае успешной зачитки возвращает 1, иначе 0.
3	Зачитать первую карточку по а/м в вед.расхода топл	ATPURT_FAVTKRT (ID_Ведомости, ID_Автомобиля)	Зачитывает первую карточку по автомобилю в ведомости расхода топлива и делает её текущей. Возвращает количество расхода по карточке.
4	Зачитать реквизиты режима работы на дату	ATPGRF_GET_DRK (Код_Графика, Дата, Код_Реквизита)	Для указанного графика работы водителя на дату возвращает значение реквизита режима работы по его коду. Коды реквизитов: 0 - Продолжительность работы, NUM,2 1 - Продолжительность оплачиваемого времени, NUM,2 2 - Продолжительность перерыва, NUM,2

5	Зачитать реквизиты водителя на дату	ATPVOD_GET_RK (Код_Водителя, Дата, Доп_Инф, Код_Реквизита)	<p>Возвращает реквизиты по путевым листам на дату для водителя (первому или второму водителю в ПЛ).</p> <p>Коды реквизитов:</p> <p>0 - количество путевых листов на дату с указанным графиком работы. Код графика помещается в параметр <Доп_Инф>. Для Доп_Инф = 0 возвращает общее количество ПЛ без учета графика работы.</p> <p>1 -То же, что и 0-й код, но без учета ПЛ с нулевым временем работы.</p> <p>2 - ID первого Путевого Листа на указанную дату или 0, если ПЛ нет.</p> <p>3 - ID следующего Путевого Листа на указан</p>
---	-------------------------------------	--	---

6	Зачитать реквизиты графика работы	ATPGRF_GET_RK (Код_Графика, Код_Реквизита)	<p>Для указанного графика работы водителя возвращает значение реквизита графика по его коду. Коды реквизитов:</p> <p>0 - Начало дневного времени, TIME 1 - Начало ночного времени, TIME 2 - Начало вечернего времени, TIME 3 - Время начала 1-й смены, TIME 4 - Продолжительность 1-й смены NUM,2 5 - Время начала 2-й смены, TIME 6 - Продолжительность 2-й смены NUM,2 7 - Время начала 3-й смены, TIME 8 - Продолжительность 3-й смены NUM,2) 9 - Время начала обеда 1-й смены, TIME 10 - Продолжительнос</p>
7	Зачитать реквизиты Путевого Листа на дату	PLST_WRK_TIM (ID_ПутевогоЛиста, Дата, Код_реквизита)	<p>Для указанного Путевого Листа на указанную дату возвращает значение реквизита согласно указанному коду.</p> <p>Значения кодов:</p> <p>0 - общее время ЧЧ.ДС 1 - время работы ЧЧ.ДС 2 - моточасы расчетн. ЧЧ.ДС 3 - пробег</p>
8	Зачитать след. карточку по а/м в вед.расхода топл.	ATPURT_NAVTKRT (ID_Ведомости, ID_Автомобиля)	<p>Зачитывает следующую карточку по автомобилю в ведомости расхода топлива и делает её текущей. Возвращает количество расхода по карточке.</p>

9	Значение реквизитов ведомости выдачи топлива	ATPVVF_GET_RK (RK_CD)	<p>Возвращает значение реквизита ведомости выдачи топлива по его коду.</p> <p>Требует зачитки ведомости функциями ATP_FVVF_ID()/ATP_NVVF_ID()</p> <p>Коды реквизитов: 0 - количество 1 - цена по ведомости 2 - ID складской карточки 3 - ID номенклатуры 4 - Код топлива 5 - ID ведомости выдачи 6 - дата ведомости 7 - номер ведомости 8 - остаток при выезде 9 - фактический расход 10 - возвращено 11 - цена по строке выдачи 12 - удельная плотность топлива на дату ведомости 13 - комментарий заголовка ведомости 14 -</p>
10	Значения реквизитов карточки вед.расхода топл.	ATPURT_KRT_RK (Код_реквизита)	<p>Возвращает значения реквизитов карточки вед.расхода топлива после установки на них функциями ATPURT_FAVTKRT и ATPURT_NAVTKRT.</p> <p>Коды реквизитов: 0 - Цена 1 - Количество 2 - Сумма</p>
11	Получить ID перв. ПЛ по а/м в вед.расхода топлива	ATPURT_AVТ_FPL (ID_Ведомости, ID_Автомобия)	<p>Возвращает ID первого путевого листа для автомобиля в ведомости расхода топлива по ID ведомости и ID автомобиля</p>

12	Получить ID первого автомобиля в вед.расхода топл.	ATP_URT_FAVT (ID_Ведомости)	Возвращает ID первого автомобиля в ведомости расхода топлива по ID ведомости.
13	Получить ID первого ПЛ в документе	FIRST_PLST_ID ()	Функция возвращает ID первого путевого листа в документе. В случае отсутствия путевых листов возвращает 0.
14	Получить ID первого ПЛ для а/м за период	FIRST_PLST_PER (ID_Автомобиля, Дата_1, Дата_2)	Возвращает ID первого путевого листа для автомобиля в интервале между датами Дата_1 и Дата_2 или ноль, если в указанном интервале ПЛ отсутствуют.
15	Получить ID первой ведомости выдачи топлива	ATP_FVVF_ID (PLST_ID, Режим)	Возвращает ID первой ведомости выдачи топлива для путевого листа, по укзанному ID в качестве параметра или ноль, если ведомостей для путевого листа нет. Режим: 0 - основное топливо 1 - дополнительное топливо
16	Получить ID первой ТТН закладки Груз	ATP_FTTN_ID(ID_Пу тевого_Листа)	Функция возвращает ID первой ТТН на закладке Груз или 0, если список накладных пуст.
17	Получить ID последнего ПЛ автомобиля на дату	LAST_PLST_DAT (ID_Автомобиля, Дата)	Функция возвращает ID последнего закрытого путевого листа для автомобиля перед указанной датой. В случае отсутствия путевого листа возвращает 0.

18	Получить ID предыдущего Путевого листа	PLST_PREV_ID (ID_Путевого_Листа)	Возвращает ID предыдущего по дате выезда Путевого Листа для автомобиля относительно текущего ПЛ, ID которого передаётся в качестве параметра. Если путевой лист для автомобиля первый, а возвращает ноль.
19	Получить ID следующей ведомости выдачи топлива	ATP_NVVF_ID (PLST_ID, Режим)	Возвращает ID следующей ведомости выдачи топлива для путевого листа, по указанному ID в качестве параметра или ноль, если следующей ведомости для путевого листа нет. Требуется предварительного вызова функции ATP_FVVF_ID. Режим: 0 - основное топливо 1 - дополнительное топливо
20	Получить ID след. ТТН закладки Груз	ATP_NTTN_ID(ID_Путевого_Листа)	Функция возвращает ID следующей ТТН на закладке Груз, или 0, если следующей накладной нет.
21	Получить ID след.а/м в вед.расхода топлива	ATPURT_AVT_NPL (ID_Ведомости, ID_Автомобиль)	Возвращает ID следующего путевого листа для автомобиля в ведомости расхода топлива по ID ведомости и ID автомобиля или ноль, если следующий ПЛ не найден.
22	Получить ID след.автомобиля в вед.расхода топл.	ATP_URT_NAVT (ID_Ведомости)	Возвращает ID следующего автомобиля в ведомости расхода топлива по ID ведомости, если следующий не найден, возвращает ноль.

23	Получить ID следующего ПЛ в документе	NEXT_PLST_ID ()	Функция возвращает ID следующего путевого листа в документе. В случае отсутствия следующего путевого листа возвращает 0.
24	Получить ID следующего ПЛ для а/м за период	NEXT_PLST_PER (ID_Автомобиля)	Возвращает ID первого путевого листа для автомобиля после успешного вызова FIRST_PLST_PER (ID_Автомобиля, Дата_1, Дата_2) в интервале между датами Дата_1 и Дата_2 или ноль, если в указанном интервале следующий ПЛ отсутствует.
25	Получить параметры первого груза по ТТН	GET_FIRST_WGT(ID_ТТН)	Функция заполняет поля параметров первой строки груза по ТТН. Возвращает: 1-найдена первая строка груза по ТТН, 0-груз по ТТН отсутствует.
26	Получить параметры первой точки маршрута ПЛ	GET_FIRSTPOINT(ID_ПутевогоЛиста)	Функция заполняет поля параметров первой точки маршрута путевого листа. Возвращает: 1-найдена первая точка маршрута, 0-отсутствует маршрут.
27	Получить параметры первой ТТН	GET_FIRST_TTN(ID_ТочкиМаршрута)	Функция заполняет поля параметров первой ТТН точки маршрута путевого листа. Возвращает: 1-найдена первая ТТН, 0-ТТН отсутствуют.
28	Получить параметры путевого листа	GET_PLST_FLD(ID_ПутевогоЛиста)	Функция заполняет поля параметров путевого листа.

29	Получить параметры след. груза по ТТН	GET_NEXT_WGT(ID_ТТН)	Функция заполняет поля параметров следующей строки груза по ТТН. Возвращает: 1-найдена следующая строка груза по ТТН, 0-поиск закончен.
30	Получить параметры след. точки маршрута ПЛ	GET_NEXTPOINT(ID_ПутевогоЛиста)	Функция заполняет поля параметров следующей точки маршрута путевого листа. Возвращает: 1-найдена следующая точка маршрута, 0-маршрут закончен.
31	Получить параметры след. ТТН	GET_NEXT_TTN(ID_ТочкиМаршрута)	Функция заполняет поля параметров следующей ТТН точки маршрута путевого листа. Возвращает: 1-найдена следующая ТТН, 0-поиск закончен.
32	Получить статус путевого листа	PLST_GET_STT (ID_Путевого_Листа)	Возвращает код статуса для Путевого Листа, ID которого передан в качестве параметра. Возвращаемые значения: 0 - Путевой Лист не найден. 1 - статус "Открыт" 2 - статус "Закрыт"
33	Проверка наличия закрытого ПЛ с указанной датой	PLST_GET_DCLS (ID_Автомобиля, Дата)	Проверяет наличие закрытого Путевого Листа с указанной датой по автомобилю. Возвращает максимальное ID ПЛ, при отсутствии закрытого ПЛ возвращает ноль.
34	Температура воздуха на дату	ATP_GET_TMPR (Дата, Смена)	Возвращает температуру воздуха на указанную дату для указанной смены (1 или 2). Для нулевой смены функция вернёт наличие записи в базе температур (1-есть/0 - температура на эту дату не указана).

35	Установка признака зачитки полей польз. ПЛ	PLST_UF_PRZ(Параметр)	Функция устанавливает признак зачитки полей пользователя путевого листа для функции GET_PLST_FLD () Значения Параметра: 0 - не зачитывать 1 - зачитывать
36	Установка признака зачитки польз.полей маршрута	ATPZVS_UF_PRZ(Параметр)	Функция устанавливает признак зачитки полей пользователя строки задания водителю путевого листа для функций GET_FIRSTPOINT / GET_NEXTPOINT Значения Параметра: 0 - не зачитывать 1 - зачитывать
	Функции для работы с реестром проводок		
	Функции доступа к параметрам проводки		
1	Получить ID аналитики дебета проводки	ENT_DT_ANL(ID_проводки)	Функция возвращает ID аналитической карточки по счёту дебета проводки с заданным ID.
2	Получить ID аналитики кредита проводки	ENT_KT_ANL(ID_проводки)	Функция возвращает ID аналитической карточки по счёту кредита проводки с заданным ID.
3	Получить ID документа операции	ENT_OPR_DOCID(ID_проводки)	Функция возвращает ID документа хоз. операции, к которой привязана проводка с заданным ID.
4	Получить ID типовой операции	ENT_OPR_TO(ID_проводки)	Функция возвращает ID типовой операции (шаблона проводок), на основании которой создана хозяйственная операция проводки.

5	Получить ID хозяйственной операции	ENT_OPR_ID(ID_проводки)	Функция возвращает ID хозяйственной операции, к которой привязана проводка с заданным ID.
6	Получить валюту проводки	ENT_VAL(ID_проводки)	Функция возвращает валюты проводки по заданному ID.
7	Получить внешний номер документа	ENT_OPR_DOCNMV(ID_проводки)	Функция возвращает внешний номер документа хозяйственной операции, к которой привязана проводка с заданным ID.
8	Получить дату документа	ENT_OPR_DOCDAT(ID_проводки)	Функция возвращает дату документа хозяйственной операции, к которой привязана проводка с заданным ID.
9	Получить дату проводки	ENT_DATE(ID_проводки)	Функция возвращает дату проводки по заданному ID.
10	Получить код документа операции	ENT_OPR_DOC(ID_проводки)	Функция возвращает код документа хозяйственной операции к которой привязана проводка с заданным ID.
11	Получить комментарий к проводке	ENT_NOTE(ID_проводки)	Функция возвращает строку комментария к проводке по заданному ID.
12	Получить курс валюты проводки	ENT_VALCUR(ID_проводки)	Функция возвращает курс валюты проводки по заданному ID.
13	Получить наименование хозяйственной операции	ENT_OPR_NM(ID_проводки)	Функция возвращает наименование хозяйственной операции, к которой привязана проводка с заданным ID.

14	Получить номер документа	ENT_OPR_DOCNMR(ID_проводки)	Функция возвращает номер документа хозяйственной операции, к которой привязана проводка с заданным ID.
15	Получить номер счёта дебета проводки	ENT_DT(ID_проводки)	Функция возвращает номер счёта дебета проводки по заданному ID.
16	Получить номер счёта кредита проводки	ENT_KT(ID_проводки)	Функция возвращает номер счёта кредита проводки по заданному ID.
17	Получить пользователя создавшего проводку	ENT_USER(ID_проводки)	Функция возвращает код пользователя создавшего проводку по заданному ID.
18	Получить пользовательский идентификатор проводки	ENT_POST_CD(ID_проводки)	Функция возвращает пользовательский идентификатор проводки с заданным ID.
19	Получить признак ручной проводки	ENT_PRZ_HND(ID_проводки)	Функция возвращает признак ручной проводки по заданному ID.
20	Получить сумму проводки	ENT_SM(ID_проводки)	Функция возвращает сумму проводки по заданному ID.
Функции навигации по реестру проводок			
1	Освободить ресурсы объекта OSB	OSB_RELEASE(Объект)	Функция освобождает ресурсы, которые были заняты объектом OSB.
2	Получить ID первой проводки	ENT_GET_FIRST(НомерЦиклаИли0,ДатаНач,ДатаКон,"Счёт",ID_карточки,ТипОборота)	Функция возвращает ID первой проводки в пределах заданного цикла сканирования, удовлетворяющей заданным параметрам. Тип оборота: 0-дебет и кредит,1-дебет,2-кредит.

3	Получить ID первой проводки по расчётам	ENT_GET_FCALC(НомерЦиклаИли0, "Счёт", ID_карточки, "Кор.счёт", ТипОборота)	Функция возвращает ID первой проводки, отобранной последней функцией ACC_CALC, удовлетворяющей заданным параметрам. Тип оборота: 0-дебет и кредит,1-дебет,2-кредит.
4	Получить ID первой проводки расчёта OSB	OSB_GET_FENTRY(Объект, Номер_Цикла)	<p>Функция возвращает ID первой проводки расчёта по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла итерации</p> <p>Примечание : функция работает только в том случае, когда для объекта расчёта OSB был установлен режим сбора информации о проводках функцией OSB_ENTRY_ON.</p>

5	Получить ID первой проводки расчёта с параметрами	OSB_GET_FPENTRY(Объект, Номер_Цикла, "Счет", Сч_Анл, "Корр_Счет", Тип_Оборота)	<p>Функция возвращает ID первой проводки расчёта по указанным параметрам фильтрации.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла итерации "Счет" - счет фильтрации (обязательный параметр) Сч_Анл - аналитика счета фильтрации "Корр_Счет" - корреспондирующий счет фильтрации Тип_Оборота - тип оборота фильтрации (1 - по дебету, 2 - по кредиту, 0 - оба)</p> <p>Примечание : функция работает только в том случае, когда для объекта расчёта OSB был установлен режим сбора инфо</p>
6	Получить ID следующей проводки	ENT_GET_NEXT(НомерЦиклаИли0)	Функция возвращает ID следующий проводки в пределах заданного цикла сканирования, начатого функцией ENT_GET_FIRST.

7	Получить ID следующей проводки расчёта OSB	OSB_GET_NENTRY(ИД_Объекта, Номер_Цикла)	<p>Функция возвращает ID следующей проводки расчёта по указанным параметрам.</p> <p>Параметры: Объект - объект, подготовленный OSB_CALC Номер_Цикла - номер цикла итерации</p> <p>Примечание : функция работает только в том случае, когда для объекта расчёта OSB был установлен режим сбора информации о проводках функцией OSB_ENTRY_ON.</p>
Функции для работы с исполнителями задач			
1	Создание исполнителя в задаче	CREATE_ISP(<Тип исполнителя>,<Код исполнителя>)	<p>Тип исполнителя - 1-сотрудник, 2-центр ответственности Код исполнителя - строка не более 50 символов</p>
2	Удаление исполнителя в задаче	DELETE_ISP(<Тип исполнителя>,<Код исполнителя>)	<p>Тип исполнителя - 1-сотрудник, 2-центр ответственности Код исполнителя - строка не более 50 символов</p>
Функции для работы со справочником ЦО			
1	Получит код ЦО	SPR_CO_GETCD(Код_ЦО, Уровень_ЦО)	<p>Возвращает код центра ответственности по заданному коду центра ответственности и уровню.</p>
2	Получить ИД из справочника	SPR_CO_GETID(Код_ЦО)	<p>Получить ИД из справочника.</p>
3	Получить наименование ЦО	SPR_CO_GETNM(Код_ЦО, Уровень_ЦО)	<p>Возвращает наименование центра ответственности по заданному коду центра ответственности и уровню.</p>

4	Получить тип справочника	SPR_CO_GETTYP(Код_ЦО)	Получить тип справочника: 1 - Сотрудник 2 - Подразделение 3 - Внешний 4 - Контрагент или 0 - Ошибка
5	Получить уровень ЦО	SPR_CO_GETLEV(Код_ЦО)	Возвращает уровень центра ответственности по заданному коду.
Функции для работы со стажами			
1	Получение количества дней доп. стажа	GET_ADDSTG_LEN(" КодСтажа", "Дата")	Получение количества дней дополнительного стажа на указанную дату для текущего сотрудника КодСтажа - Код дополнительного стажа Дата - Дата, на которую необходимо получить стаж Заполняет переменные: WRK_STG_YEARS - Количество лет стажа WRK_STG_MONTHS - Количество месяцев стажа WRK_STG_DAYS - Количество дней стажа Возвращает 1 - если стаж есть, 0 - если нет

2	Расчет стажей на дату	KDR_FILL_STG_DATE (Дата,Режим)	<p>Расчитывает размер стажей на указанную дату</p> <p>Параметры:</p> <p>Дата - дата расчета</p> <p>Режим - режим вывода запроса на расчет. true - выводить запрос на расчет; false - не выводить запрос на расчет</p> <p>При любом значении параметра Режим проверяется актуальность расчета.</p> <p>При значении false, в случае обнаружения неактуального расчета, расчет выполняется без запроса пользователя</p>
Функции доступа к свойствам автомобиля			
1	Заполнить реквизиты автомобиля/прицепа	AVT_GET_PRM (ID_Автомобиля/прицепа)	<p>Функция зачитывает реквизиты автомобиля/прицепа по его ID.</p> <p>Запляет поля AVT_* (Реквизиты автомобиля)</p> <p>Запляет поля AVTM_* (Реквизиты марки автомобиля)</p> <p>Если автомобиль и марка найдены, возвращает 1, иначе возвращает ноль.</p>

2	Зачитать значение реквизита ремонта по коду	AVTRPR_GET_RK (Код_Реквизита)	<p>Функция возвращает значение реквизита по его коду из записи ремонта, ранее зачитанной функциями AVTRPR_FIRST()/ AVTRPR_NEXT()</p> <p>Коды реквизитов:</p> <p>1 - Тип ремонта (0-тек./1-кап./2-ТО-1/3-ТО-2/4-ТО-3) 2 - Дата начала плановая 3 - Время начала плановое 4 - Дата Окончания плановая 5 - Время окончания плановое 6 - Дата начала факт. 7 - Время начала факт. 8 - Дата окончания факт. 9 - Время окончания факт. 10 - Пробег 11 - Код водителя 12 - Таб.№ водителя 13 - Код заказчика (контрагент) 14 -</p>
3	Первая запись в картотеке автотранспорта	AVT_GET_FIRST ()	<p>Возвращает ID первого автомобиля/прицепа в картотеке автотранспорта или ноль, если в картотеке нет записей.</p>
4	Получить ID автомобиля по его гаражному номеру	AVTG_GET_ID ("Гар.номер")	<p>Возвращает ID автомобиля по его гаражному номеру или ноль, если автомобиля с указанным гаражным номером в картотеке автотранспорта нет.</p>

5	Получить ID автомобиля по его Гос.номеру	AVT_GET_ID ("Гос.номер")	Возвращает ID автомобиля по его Гос.номеру или ноль, если автомобиля с указанным гос.номером в картотеке автотранспорта нет.
6	Получить время работы по ПЛ на дату	АТРАVT_WRK_TIM(ID_Автомобили , Дата , Режим_Вызова)	Для указанного автомобиля на указанную дату возвращает время работы (ЧЧ.ММ). (Для многодневных ПЛ данные берутся с закладки "График") Режимы вызова: 0 - Учитывать только закрытые ПЛ 1 - Учитывать все ПЛ по автомобилю за день.
7	Получить дату последнего ремонта по типу ремонта	AVT_LRPR_DAT (ID_Автомобили, Тип_Ремонта)	Возвращает дату последнего проведения ремонта для автомобиля с указанным ID по типу ремонта: 0 - текущий 1 - капитальный 2 - ТО-1 3 - ТО-2 4 - ТО-3 Если запись указанного типа ремонта отсутствует, возвращает ноль.
8	Получить код типа автомобиля	АТР_TYPE_CD("Гос.номер")	Функция возвращает код типа автомобиля по его гос.номеру

9	Получить моточасы расчетн. по ПЛ на дату	АТРАVT_MTH_TIM(ID_Автомобиля , Дата , Режим_Вызова)	<p>Для указанного автомобиля на указанную дату возвращает значение поля "Моточасы расчетн." (ЧЧ.ДС). (Для многодневных ПЛ данные берутся с закладки "График")</p> <p>Режимы вызова: 0 - Учитывать только закрытые ПЛ 1 - Учитывать все ПЛ по автомобилю за день.</p>
10	Получить наименование типа автомобиля	АТР_TYPE_NM("Гос.номер")	Функция возвращает наименование типа автомобиля по его гос.номеру
11	Получить первую запись ремонта в интервале дат	AVTRPR_FIRST (ID_Автомобиля, Дата1, Дата2)	Устанавливает на первую запись ремонта в интервале дат.
12	Получить первую запись ремонта по Таб.№	AVTR_TN_FIRST (ТН_Водителя, Дата1, Дата2)	Устанавливает на первую запись ремонта в интервале для водителя по ТН. Возвращает ID ремонта или ноль, если ремонт не найден.
13	Получить свойства марки автомобиля	AVT_GET_MRK("Гос.номер")	<p>Функция читает данные марки автомобиля по его гос.номеру. Заполняет поля AVTM_... Если автомобиль и марка найдены, возвращает 1, иначе возвращает 0.</p>
14	Получить след. запись ремонта	AVTRPR_NEXT (ID_Автомобиля)	Устанавливает на следующую запись ремонта. Даты периода устанавливаются при вызове AVTRPR_FIRST()

15	Получить следующую запись ремонта по Таб.№	AVTR_TN_NEXT (TH_Водителя)	Устанавливает на следующую запись ремонта. Даты периода устанавливаются при вызове AVTR_TN_FIRST() Возвращает ID ремонта или ноль, если ремонт не найден.
16	Следующая запись в картотеке автотранспорта	AVT_GET_NEXT ()	Возвращает ID следующего автомобиля/прицепа в картотеке автотранспорта или ноль, если следующая запись отсутствует. При работе использует результат, возвращенный предыдущим вызовом AVT_GET_FIRST() или AVT_GET_NEXT().
17	Создать запись ремонта	AVTRPR_CREATE (ID_Автомобиля)	Создаёт запись ремонта в карточке автомобиля на основании реквизитов, предварительно установленных функцией AVTRPR_SET_RK ()

18	Установить значение реквизита ремонта по коду	AVTRPR_SET_RK (ID_Автомобиля, Значение, Код_Реквизита)	<p>Функция Устанавливает значение реквизита по его коду для создания записи ремонта функцией AVTRPR_CREATE(). Перед созданием записи ремонта функция может быть вызвана несколько раз с различными кодами реквизитов.</p> <p>Коды реквизитов:</p> <p>1 - Тип ремонта, NUM.0 (0-тек./1-кап./2-ТО-1/3-ТО-2/4-ТО-3)</p> <p>2 - Дата начала плановая, DATE</p> <p>3 - Время начала плановое, TIME</p> <p>4 - Дата Окончания плановая, DATE</p> <p>5 - Время окончания плановое, TIME</p> <p>6 - Дата начала факт., DATE</p> <p>7 - Время начала факт., TIME</p> <p>8 - Дата оконч</p>
Функции доступа к справочнику маршрутов			
1	Дать первую точку маршрута	TRDRT_FPOINT (Код_маршрута)	Зачитывает поля первой точки маршрута по его коду (строка). В случае успешного поиска возвращает единицу, в противном случае - ноль.
2	Дать следующую точку маршрута	TRDRT_NPOINT (Код_маршрута)	Зачитывает поля следующей точки маршрута по его коду (строка). В случае успешного поиска возвращает единицу, в противном случае - ноль.
3	Зачитать поля заголовка маршрута	TRDRT_GET_FLD (Код_маршрута)	Зачитывает поля маршрута по его коду (строка). В случае успешного поиска возвращает единицу, в противном случае - ноль.

Функции карточки затрат			
1	Вернуть кол-во по карточке затрат	PCOST_PQT (Дата периода, ID продукции)	Функция возвращает кол-во продукции по карточке затрат за указанный период.
2	Вернуть сумму затрат на материалы	PCOST_MTR_SUM (Дата периода, ID продукции)	Функция возвращает сумму материальных затрат за заданный период.
3	Вернуть сумму затрат на материалы (списание)	PCOST_MTR_EXP(Дата периода, ID продукции)	Функция возвращает сумму списанных материальных затрат на ГП за заданный период
4	Вернуть сумму затрат по наклад. расходам	PCOST_OVR_SUM (Дата периода, ID продукции)	Функция возвращает сумму затрат по наклад. расходам за заданный период.
5	Вернуть сумму затрат по статье калькуляции	PCOST_ITM_SUM (Дата периода, ID продукции, "Код статьи")	Функция возвращает сумму по статье калькуляции за указанный период.
6	Вернуть сумму общих затрат	PCOST_SUM (Дата периода, ID продукции)	Функция возвращает сумму общих затрат за указанный период.
7	Вернуть сумму трудозатрат	PCOST_SAL_SUM (Дата периода, ID продукции)	Функция возвращает сумму трудозатрат за заданный период.
8	Вернуть сумму трудозатрат (списание)	PCOST_SAL_EXP(Дата периода, ID продукции)	Функция возвращает сумму списанных трудозатрат на ГП за заданный период
Функции карточки предприятия			
1	Получить информацию по счёту предприятия	GET_CRTPRED_SCH("номер счёта")	Заполняет поля счёта карточки предприятия, по номеру счёта. Вход : номер счёта предприятия Выход : будут перезаписаны поля PRED_RS_*(если возврат 1) Возврат: 1-успешно/0-нет

2	Получить признак Программная классификация расх.МБ	SYSKBK_GET_PRZ ("Код_КБК")	Возвращает признак "Программная классификация расходов местного бюджета" по коду записи в справочнике "Коды бюджетной классификации".
3	Получить счет предприятия, по реквизитам КПКР и Фо	GET_CRTSCHBYRCV ("КОД КПКР", "КОД_ФОНД А")	Заполняет поля счёта карточки предприятия, по реквизитам КПКР и Фонд в разрезе казначейского типа(искл. логически удаленные). Выход : будут перезачитаны поля PRED_RS_*(если возврат 1) Возврат: 1-удачно/0-нет
Функции модуля "График отпусков"			
1	Итоги отпусков по группе раб. на дату и вида балан	GO_GET_LST_OTP_GRP_DATE_VID (ДатаОтп, ВидБлн)	Функция формирует набор данных КпуОтпTot, содержащий итоговые данные отпусков по группе работников, значения Кпу_Rcd которых присутствуют в таблице КпуRcdОтпTot Параметры: ДатаОтп - дата, на которую выполняется расчёт отпусков, ВидБлн - Вид итогов баланса отпусков: 0 - Предоставление отпуска или компенсации (план на дату, факт-полностью) 1 - Резерв отпусков (план на дату, факт на дату, отпуска, переходящие через расч. дату - полностью)

2	Итоги отпусков по группе раб. по списку дат	GO_GET_LST_OTP_LST_DATE(Неиспольз, ИспДниНаДату)	<p>Функция формирует набор данных КпуОтпTot, содержащий итоговые данные отпусков по группе работников (значения Кпу_Rcd которых присутствуют в таблице КпуRcdОтпTot) и по списку дат (значения которых присутствуют в таблице LstDatОтпTot)</p> <p>Параметры: Неиспольз - Включать в набор: 0 - Все отпуска 1 - Только неиспользованные отпуска ИспДниНаДату: 0 - Используемые дни отпуска полностью 1 - Расчет использованных дней отпуска на дату</p>
3	Получить период, за который предоставляется отпуск	GO_GET_OTP_PER("N_ЛК",ДатаОтп,ДатаНач,ДатаКон)	<p>Функция возвращает даты начала и конца периода, за который предоставляется отпуск (отпускного периода). Возвращает нулевое значение в случае ошибки.</p> <p>Параметры: N_ЛК - № личной карточки, ДатаОтп - дата начала отпуска, ДатаНач - дата начала отпускного периода (возвращаемый параметр), ДатаКон - дата конца отпускного периода (возвращаемый параметр).</p>

4	Построить итоги отпусков по группе работн на дату	GO_GET_LST_OTP_GRP_DATE(ДатаОтп, Неиспольз, ИспДниНаДату)	<p>Функция формирует набор данных КпуОтпTot, содержащий итоговые данные отпусков по группе работников, значения Кпу_Rcd которых присутствуют в таблице КпуRcdОтпTot</p> <p>Параметры: ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0 - Все отпуска 1 - Только неиспользованные отпуска ИспДниНаДату: 0 - Исползованные дни отпуска полностью 1 - Расчет исползованных дней отпуска на дату</p>
5	Построить итоги отпусков по работнику(ID)на дату	GO_GET_LST_OTP_RCD_DATE("ID работника",ДатаОтп,Неиспольз,ИспДниНаДату)	<p>Функция формирует набор данных КпуОтпTot, содержащий итоговые данные отпусков по работнику.</p> <p>Параметры: ID работника - значение поля Кпу_Rcd записи работника, ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0-Все отпуска 1-Только неиспользованные отпуска ИспДниНаДату: 0 - Исползованные дни отпуска полностью 1 - Расчет исползованных дней отпуска на дату</p>

6	Построить итоги отпусков по работнику(№ ЛК)на дату	GO_GET_LST_OTP_D ATE("N_ЛК",ДатаОтп,Неиспольз,ИспНаДату)	<p>Функция формирует набор данных КрпОтпTot, содержащий итоговые данные отпусков по работнику.</p> <p>Параметры: N_ЛК - № личной карточки, ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0-Все отпуска 1-Только неиспользованные отпуска ИспНаДату - 1 - Использованные дни на дату 0 - Использованные дни за весь отпуск</p>
7	Построить набор итогов отпусков по группе работник	GO_GET_LST_OTP_GRP(ДатаОтп, Неиспольз)	<p>Функция формирует набор данных КрпОтпTot, содержащий итоговые данные отпусков по группе работников, значения Крп_Rcd которых присутствуют в таблице КрпRcdОтпTot</p> <p>Параметры: ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0 - Все отпуска 1 - Только неиспользованные отпуска</p>

8	Построить набор итогов отпусков по работнику(ID)	GO_GET_LST_OTP_RCD("ID работника",ДатаОтп,Неиспольз)	<p>Функция формирует набор данных КрпОтпTot, содержащий итоговые данные отпусков по работнику.</p> <p>Параметры: ID работника - значение поля Крп_Rcd записи работника, ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0-Все отпуска 1-Только неиспользованные отпуска</p>
9	Построить набор итогов отпусков по работнику(№ ЛК)	GO_GET_LST_OTP("N_ЛК",ДатаОтп,Неиспольз)	<p>Функция формирует набор данных КрпОтпTot, содержащий итоговые данные отпусков по работнику.</p> <p>Параметры: N_ЛК - № личной карточки, ДатаОтп - дата, на которую выполняется расчёт неиспользованных отпусков, Неиспольз - Включать в набор: 0-Все отпуска 1-Только неиспользованные отпуска</p>
Функции модуля "Оповещения о наступающих событиях"			
1	Получить значения второй помеченной записи	KDR_GET_NXTNFE('Id предыдущей помеченной записи','CA сотрудника','Дата события',	Возвращает Id следующей помеченной записи/0-Нет помеченных событий
2	Получить значения первой помеченной записи	KDR_GET_FSTNFE('CA сотрудника','Дата события','Тип события')	Возвращает Id первой помеченной записи/0-Нет помеченных событий

	Функции модуля "Штатное расписание"		
1	Получить код родительского подразделения	SR_GET_PARPDR("Номер штатного расписания", "Код подразделения")	Возвращает код родительского подразделения штатного расписания. При отсутствии родительского подразделения или неверных значениях переданных параметров возвращает пустую строку.
	Функции навигации по журналу взаиморасчетов		
1	Дать дату записи	ACN_GETDAT()	Функция возвращает дату текущей записи в журнале взаиморасчетов.
2	Дать ИД накладной на основании которой был возврат	ACN_GETRTOSNID()	Функция возвращает ИД накладной на основании которой был возврат, по текущей записи в журнале взаиморасчетов по возврату.
3	Дать ИД связного документа	ACN_GETFLWID()	Функция возвращает ИД связного документа по текущей записи в журнале взаиморасчетов.
4	Дать курс записи	ACN_GETKURS()	Функция возвращает курс текущей записи в журнале взаиморасчетов.
5	Дать первую запись по возврату на осн. накладной	ACN_FIRSTRETNK(ID накладной)	Функция ищет первую запись в журнале взаиморасчетов по возврату на основании накладной. При нахождении возвращает 1, иначе - 0.
6	Дать первую запись по договору	ACN_FIRSTDOG(ID Договора)	Функция ищет первую запись в журнале взаиморасчетов по договору. При нахождении возвращает 1, иначе - 0.

7	Дать первую запись по накладной	ACN_FIRSTNAK(ID накладной)	Функция ищет первую запись в журнале взаиморасчетов по накладной. При нахождении возвращает 1, иначе - 0.
8	Дать первую запись по операции договора	ACN_FIRSTDOGOP(ID договора, ID операции)	Функция ищет первую запись в журнале взаиморасчетов по операции договора. При нахождении возвращает 1, иначе - 0.
9	Дать первую запись по отчету консигнатора	ACN_FIRSTCON(ID отч. конс.)	Функция ищет первую запись в журнале взаиморасчетов по отчету консигнатора. При нахождении возвращает 1, иначе - 0.
10	Дать первую запись по счету	ACN_FIRSTACC(ID счета)	Функция ищет первую запись в журнале взаиморасчетов по счету. При нахождении возвращает 1, иначе - 0.
11	Дать следующую запись	ACN_NEXT()	Функция ищет следующую запись в журнале взаиморасчетов. При нахождении возвращает 1, иначе - 0.
12	Дать сумму записи	ACN_GETSM()	Функция возвращает сумму текущей записи в журнале взаиморасчетов.
13	Дать сумму счета	ACN_GETACCSM(ID счета)	Функция возвращает сумму счета.
14	Дать тип записи	ACN_GETTYPE()	Функция возвращает тип текущей записи в журнале взаиморасчетов.
	Функции навигации по реестру финансовых таблиц		
	Функции навигации		

1	Дать первую таблицу	FTBL_FIRST(НомерЦиклаИли0, ШаблонТаблицы)	Функция возвращает ID первой расчетной таблицы по заданному через код или ID шаблону.
2	Дать первую таблицу с учетом номера п/п	FTBL_FIRSTN(НомерЦиклаИли0, ШаблонТаблицы)	Функция возвращает ID первой расчетной таблицы по заданному через код или ID шаблону с учетом номера по порядку.
3	Дать следующую таблицу по шаблону	FTBL_NEXT(НомерЦиклаИли0, ШаблонТаблицы)	Функция возвращает ID следующей расчетной таблицы по заданному через код или ID шаблону. Требует первоначального вызова FTBL_FIRST.
4	Дать следующую таблицу по шаблону с учетом номера	FTBL_NEXTN(НомерЦиклаИли0, ШаблонТаблицы)	Функция возвращает ID следующей таблицы по заданному через код или ID шаблону с учетом номера по порядку. Требует первоначального вызова FTBL_FIRSTN.
5	Копировать таблицу	FTBL_COPY(ID Таблицы, ДатаПериода, Номер, КодПодр, Квартал, Год)	Функция копирует таблицу с указанным ID и создает новую таблицу по ключевым реквизитам: ДатаПериода, Номер, КодПодр, Квартал, Год Функция возвращает ID скопированной таблицы.
6	Найти таблицу	FTBL_FIND(ШаблонТаблицы, ДатаПериода, Номер, КодПодр, Квартал, Год)	Найти таблицу по шаблону и ключевым реквизитам. Функция возвращает ID расчетной таблицы по заданному через код или ID шаблону и ключевым реквизитам.
7	Создать таблицу	FTBL_CREATE(ШаблонТаблицы, ДатаПериода, Номер, КодПодр, Квартал, Год)	Создать таблицу по шаблону и ключевым реквизитам.

8	Удалить таблицу	FTBL_DELETE(ШаблонТаблицы, ДатаПериода, Номер, КодПодр, Квартал, Год)	Удалить таблицу по шаблону и ключевым реквизитам.
9	Установить текущий журнал	FTBL_SEL_JRN("КодЖурнала")	Функция устанавливает текущий журнал шаблонов расчетных таблиц для корректной работы функций навигации по таблицам шаблона. Функция работает с учетом настроек прав доступа к журналам.
10	Установить текущий журнал расчетных таблиц	FTBL_SEL_JRN_RT("КодЖурнала")	Функция устанавливает текущий журнал расчетных таблиц для корректной работы функций навигации по таблицам. Функция работает с учетом настроек прав доступа к журналам.
	Функции навигации по спецификации документа		
1	Возвращает ИД счета по внешнему номеру и контрагент	ACC_GETID(Внеш. номер документа, Код контрагента)	Функция ищет ИД счета по внешнему номеру и коду контрагента При нахождении возвращает ИД документа, иначе - 0.

2	Выполнить прог-му калькуляции для строк документа	TRDS_RUNPRG(изменившийся реквизит)	Функция производит запуск программы калькуляции спецификации для всех строк документа. Параметр изменившийся реквизит может принимать такие значения: 20 - атрибут 37 - характеристика 54 - количество тары 55 - залоговая цена тары 57 - количество 59 - количество в осн. ЕИ 73 - скидка с цены 74 - цена 75 - сумма 77 - сумма к оплате 79 - цена в баз. валюте 80 - сумма в баз. валюте 81 - залоговая цена 82 - залоговая стоимость 83 - ставка налога 100 - сумма оприходования 103 - налоговая цена
3	Дать артикул по строке документа	TRDS_GETART()	Функция возвращает артикул по текущей строке документа.
4	Дать ИД аналитической карточки по строке документа	TRDS_GETRCDANK()	Функция возвращает ИД аналитической карточки по текущей строке документа.
5	Дать ИД товара по строке документа	TRDS_GETOPA()	Функция возвращает ИД товара по текущей строке документа.
6	Дать кол-во по ном. группе спецификации документа	TRDS_GETGRPQT(тип ном., "код группы", "ЕИ")	Функция возвращает количество товара по номенклатурной группе в документа в указанной ЕИ.
7	Дать кол-во по строке документа в осн. ЕИ	TRDS_GETQTOSN()	Функция возвращает количество в основной ЕИ по текущей строке документа.

8	Дать комментарий строки документа	TRDS_GETCMT()	Функция возвращает комментарий строки документа.
9	Дать метод расчета текущего налога	TRDSTAX_ISIZVL()	Функция возвращает метод расчета текущего налога по строке документа. 1 - извлекается, 0 - начисляется.
10	Дать ном. группу по строке документа	TRDS_GETGRP()	Функция возвращает ном. группу по текущей строке документа.
11	Дать первую ном. группу спецификации документа	TRDS_FIRSTGRP()	Функция ищет первую номенклатурную группу в документе. При нахождении возвращает код группы, иначе - пустую строку.
12	Дать первую строку спецификации документа	TRDS_FIRST()	Функция ищет первую строку спецификации документа. При нахождении возвращает 1, иначе - 0.
13	Дать первую строку спецификации указанного раздела	TRDS_FIRSTDOCP(Тип документа, ИД документа, ИД раздела)	Функция ищет первую строку спецификации указанного раздела указанного документа. При нахождении возвращает 1, иначе - 0. Параметр "Тип документа" может принимать значения: 1 - товарный документ; 2 - счет; 4 - заказ.
14	Дать первую строку указанного документа	TRDS_FIRSTDOC(Тип документа, ИД документа)	Функция ищет первую строку спецификации указанного документа. При нахождении возвращает 1, иначе - 0. Параметр "Тип документа" может принимать значения: 1 - товарный документ; 2 - счет; 17 - заказ.

15	Дать первый налог по строке документа	TRDSTAX_FIRST()	Функция ищет первый налог по строке документа. При нахождении возвращает - 1, иначе - 0.
16	Дать следующий налог по строке документа	TRDSTAX_NEXT()	Функция ищет следующий налог по строке документа. При нахождении возвращает - 1, иначе - 0.
17	Дать следующую ном. группу спецификации документа	TRDS_NEXTGRP()	Функция ищет следующую номенклатурную группу в документе. При нахождении возвращает код группы, иначе - пустую строку.
18	Дать следующую строку спецификации документа	TRDS_NEXT()	Функция ищет следующую строку спецификации документа. При нахождении возвращает 1, иначе - 0.
19	Дать сумму к оплате по строке документа	TRDS_GETSMOPL()	Функция возвращает сумму к оплате по текущей строке документа.
20	Дать сумму по строке документа	TRDS_GETSM()	Функция возвращает сумму по текущей строке документа.
21	Дать сумму текущего налога	TRDSTAX_GETSM()	Функция возвращает сумму текущего налога по строке документа.
22	Дать шифр ставки текущего налога	TRDSTAX_GETRT()	Функция возвращает шифр ставки текущего налога по строке документа.
23	Дать шифр текущего налога	TRDSTAX_GETTX()	Функция возвращает шифр текущего налога по строке документа.
24	Обновить реестр спецификации документа	TRDS_SPECREF()	Функция обновляет реестр спецификации документа
25	Создание товарной позиции счета	INS_TRDS_ACC(ИД счета, артикул, количество, сумма, комментарий)	Создание товарной позиции счета
Функции отчета РСВ-1			

1	Заполнение стажей для РСВ-1	PAY_FILL_STGRSV1(ДатаНач, ДатаКон, ПоСЕ, "КодСЕ", ВклПодчСЕ)	Заполнить раздел 6.8 отчета РСВ-1 Параметры: ДатаНач - дата начала периода расчета ДатаКон - дата конца периода расчета ПоСЕ - 1 - по выбранной структурное единице, 0 - по всем КодСЕ - код выбранной структурное единицы или пустая строка ВклПодчСЕ - включать подчиненные структурные единицы
2	Инициализация РСВ-1	PAY_RSV1_INIT	Инициализация РСВ-1 Возврат: 1 - успешно 0 - ошибка
3	Получить первую запись выборки раздела 6.8	PAY_RSV1_R68GETFIRST	Получить первую запись выборки раздела 6.8 Возврат: 1 - успешно 0 - выборка пуста В случае успеха заполняет переменные: RSV68_TN RSV68_BEG RSV68_END RSV68_TUS RSV68_OSU RSV68_STS RSV68_DNP

4	Получить следующую запись выборки раздела 6.8	PAY_RSV1_R68GETN EXT	Получить следующую запись выборки раздела 6.8 Возврат: 1 - успешно 0 - Конец выборки В случае успеха заполняет переменные: RSV68_TN RSV68_BEG RSV68_END RSV68_TUS RSV68_OSU RSV68_STS RSV68_DNP
5	Терминация PCB-1	PAY_RSV1_TERM	Терминация PCB-1 Возврат: 1 - успешно 0 - ошибка
6	Установить выборку раздела 6.8 по Таб.№	PAY_RSV1_R68PART (ТабНом)	Установить выборку раздела 6.8 по Таб.№ Параметры: ТабНом - табельный номер на котором необходимо спозиционироваться или 0 - вся выборка
Функции передачи данных для экспорта			
1	Функции создания детальной строки в группе	EXT_G_ADD_DTL("Код формы",НомерГруппы,ДатаПериода,ТипПериода)	Функция создаёт детальную строку для указанной группы реквизитов формы и возвращает её числовой номер.
2	Функции удаления строк по группе реквизитов	EXT_G_CLEAR("Код формы",НомерГруппы,ДатаПериода,ТипПериода)	Функция удаляет все детальные строки по указанной группе реквизитов. Тип периода - как в функции EXT_SET_RCV.
3	Функции установки значения реквизита группы	EXT_G_SET_RCV("Код формы",НомерГруппы,НомерСтр,"Код реков",ДатаПер,ТипПер,Значени	Функция устанавливает значение реквизита группы формы по указанным параметрам. Тип периода: 0-месяц, 10-квартал, 20-полугодие, 25-9 месяцев, 30-год

4	Функции установки значения реквизита формы	EXT_SET_RCV("Код формы", "Код реквизита", ДатаПериода, ТипПериода, Значение)	Функция устанавливает значение реквизита формы по указанным параметрам. Тип периода: 0-месяц, 10-квартал, 20-полугодие, 25-9 месяцев, 30-год
5	Функция проверки признака активности	EXT_IS_ACTIVE()	Функция проверяет активность механизма передачи в модуль экспорта в данных электронном виде.
Функции пересчета количеств по номенклатуре			
1	Дать точность единицы измерения	NOM_GETACC ("Артикул", "Шифр Ед.Изм.")	Функция возвращает точность единицы измерения для указанной номенклатуры.
2	Пересчет количества по номенклатуре	RECALC_UNITS("Артикул", "К-во вход.", "Шифр ЕИ вход.", "Шифр ЕИ результат.")	Пересчет указанного количества по номенклатуре в новую единицу измерения. Функция возвращает пересчитанное количество.
3	Расчитать вес брутто в указанной ед.изм.	CALCQT_BRUTTO("Артикул", "Количество", "Ед.Изм.", "Ед.Изм. брутто")	Расчет веса брутто в указанной единице измерения исходя из заданного количества
4	Расчитать вес нетто в указанной ед.изм.	CALCQT_NETTO("Артикул", "Количество", "Ед.Изм.", "Ед.Изм. нетто")	Расчет веса нетто в указанной единице измерения исходя из заданного количества
Функции по "Денежным средствам"			
1	Получить Ид шифра по коду шифра	FIN_SPRSPPID(Код_Шифра)	Возвращает Ид шифра по коду шифра

2	Получить курсовую разницу	GETKRS(СчетДС,"Вал", "ВалВыб",Дата)	Функция возвращает сумму курсовой разницы по валюте выбора в заданной валюте по счету "Денежных средств" и дате операционного дня.
3	Получить наименование контрагента	GETFLOWPTNNM (Ид документа)	Возвращает наименование контрагента из денежной операции
4	Разноска платежа по договорам	FIN_SETDOGSUM(Ид договора, Ид операции договора, Сумма)	Записывает сумму платежа на указанный договор
5	Разноска платежа по счетам	FIN_SETACCSUM(Ид счета, Сумма)	Записывает сумму платежа на указанный счет
6	Реквизиты выписки банка	READ_FLOW(Ид выписки банка)	Читает реквизиты выписки банка по Ид.
7	Реквизиты кассового документа	READ_CASHDOC(Ид кассового документа)	Читает реквизиты кассового документа по Ид.
8	Реквизиты платежного документа	READ_BNKDOC(Ид платежного документа)	Читает реквизиты платежного документа по Ид.
Функции получения информации по ном. группе			
1	Получить ЕИ группы номенклатуры	SKL_GETGRPEI("Код группы")	Функция по коду номенклатурной группы возвращает настроенную для нее ЕИ.
2	Получить код родительской группы для группы ном.	SKL_GETGRPPAR("Код группы")	Функция по коду группы номенклатуры возвращает код родительской группы.
3	Получить комментарий группы номенклатуры	SKL_GETGRPCMT("Код группы")	Функция по коду группы номенклатуры возвращает ее комментарий.
4	Получить номер уровня группы номенклатуры	SKL_GETGRPLEV("Код группы")	Функция по коду номенклатурной группы возвращает номер уровня группы.
Функции получения остатка и движения			

1	Функция получения кол-ва прихода номенкл.	GETSTOR_MOVPN("Код подр.", "Артикул", Дата нач., Дата оконч.)	Функция возвращает кол-во прихода номенкл. за заданный период по месту хранения.
2	Функция получения кол-ва прихода прод.	GETSTOR_MOVP("Код подр.", ID прод., Дата нач., Дата оконч.)	Функция возвращает кол-во прихода продукции по месту хранения за заданный период.
3	Функция получения кол-ва расхода номенкл.	GETSTOR_MOVRN("Код подр.", "Артикул", Дата нач., Дата оконч.)	Функция возвращает кол-во расхода номенкл. за заданный период по месту хранения.
4	Функция получения кол-ва расхода прод.	GETSTOR_MOVR("Код подр.", ID прод., Дата нач., Дата оконч.)	Функция возвращает кол-во расхода продукции по месту хранения за заданный период.
5	Функция получения остатка ном. по месту хранения	GETSTOR_OSTN("Код подр.", "Артикул", Дата)	Функция возвращает количество остатка номенкл. по месту хранения на конец дня.
6	Функция получения остатка прод. по месту хранения	GETSTOR_OST("Код подр.", ID прод., Дата)	Функция возвращает количество остатка продукции по месту хранения на конец дня.
7	Функция получения суммы прихода номенклатуры	GETSTOR_MOVPNS("Код подр.", "Артикул", Дата нач., Дата оконч.)	Функция возвращает сумму прихода номенкл. за заданный период по месту хранения
8	Функция получения суммы прихода прод.	GETSTOR_MOVPS("Код подр.", ID прод., Дата нач., Дата оконч.)	Функция возвращает сумму прихода продукции по месту хранения за заданный период
9	Функция получения суммы расхода прод.	GETSTOR_MOVRS("Код подр.", ID подр., Дата нач., Дата оконч.)	Функция возвращает сумму расхода продукции по месту хранения за заданный период.
	Функции получения параметров номенклатуры		

1	Дать артикул по коду аналитики	NOM_GETART ("Код аналитики")	Функция возвращает артикул для номенклатуры с указанным кодом аналитики.
2	Дать запас для торгового зала	GETNRMTRD("артикул", "код склада")	Функция возвращает нормативный запас указанного артикула для торгового зала для указанного склада
3	Дать максисальный нормативный запас	GETNRMMAX("артикул", "код склада")	Функция возвращает максимальный нормативный запас указанного артикула для указанного склада
4	Дать минимальный нормативный запас	GETNRMMIN("артикул", "код склада")	Функция возвращает минимальный нормативный запас указанного артикула для указанного склада
5	Дать ставку налога номенклатуры	NOM_TAXRATE ("Артикул", "Шифр налога")	Функция возвращает ставку налога, настроенную для указанной номенклатуры
6	Окно выбора номенклатуры	SKLNOMSPRTMC()	Функция выбора номенклатуры
7	Установить максимальный нормативный запас	SETNRMMAX (ID номенклатуры, "код склада", "кол-во запаса")	
8	Установить минимальный нормативный запас	SETNRMMIN (ID номенклатуры, "код заказа", "кол-во запаса")	
9	Установить нормативный запас торгового зала	SETNRMTRD (ID номенклатуры, "код склада", "кл-во запаса")	
10	Функция чтения реквизитов группы по коду	READPRMGRP("Код группы")	Функция чтения реквизитов группы по коду группы
11	Функция чтения реквизитов ЕИ номенклатуры	READPRMEI("Артикул", "Шифр ЕИ")	Функция чтения реквизитов ЕИ номенклатуры по артикулу номенклатуры и шифру ЕИ

12	Функция чтения реквизитов номенклатуры	READPRMNOM ("Артикул")	Функция чтения реквизитов номенклатуры по артикулу.
13	Функция чтения реквизитов партии	READPRMPRT("Шифр партии", "Артикул")	Функция чтения реквизитов партии по шифру партии и артикулу
14	Функция чтения реквизитов следующей тары	READNEXTTARA()	Функция чтения реквизитов следующей тары
15	Функция чтения реквизитов тары номенклатуры	READPRMTARA("Артикул номенклатуры", "Артикул тары")	Функция чтения реквизитов тары номенклатуры. Первый параметр - артикул номенклатуры, второй - артикул искомой тары. Если второй параметр пустой, функция возвратит первую тару, привязанную к номенклатуре. Для получения следующей нужно вызвать READNEXTTARA()
	Функции работы с картотекой взаиморасчетов		

1	Получить обороты по картотеке контрагентов	FILL_OBR_ACCPLTCN T(ДатаОт,ДатаДо, Код валюты,Код котировки)	Получить обороты по картотеке контрагентов Функция возвращает текущее сальдо расчетов по картотеке контрагентов на диапазон дат по указанной котировке и валюте. Данные записываются во временную таблицу TempDocS: TDocS_Station - код станции = GetStationCd() TDocS_TypHdr - тип заголовка = 71 TDocS_Rcd - Rcd контрагента TDocS_FldStr1 - код контрагента TDocS_FldStr2 - наименование контрагента TDocS_UserSum1 - входящее сальдо TDocS_UserSum2 - сумма прихода TDocS_U
2	Получить сальдо расчетов по контрагенту на дату	FPTN_BAL (Код контрагента,Дата,Код котировки,Код валюты)	Функция возвращает текущее сальдо расчетов по контрагенту на дату по указанной котировке и валюте.
	Функции работы с качеств. характеристиками		
1	Получить значение характ. партии	GETPRITITEMSH(Тип, "Шифр парт.", ID товара, "Код")	Тип 0 - складская партия/ 1 - производственная Получить значение характеристики партии товара по шифру партии, ID товара, коду хар-ки партии
2	Получить значение характ. партии по коду	GETPRITITEM_S ("Код")	Функция возвращает симв. значение характеристики партии по трехсимвольному коду

3	Получить значение характ. партии по коду	GETPRTITEM ("Код")	Функция возвращает значение характеристики партии по трехсимвольному коду
4	Получить значение характ. партии по коду и артик.	GETPRTITEMARTS ("Артикул", "Код")	Возвращает симв. значение характеристики партии для указанного артикула по коду
5	Получить значение характ. партии по коду и артикул	GETPRTITEM_ART ("Артикул", "Код")	Возвращает значение характеристики партии для указанного артикула по коду
6	Получить значение характ. по коду и артикулу	GETNOMITEMARTS ("Артикул", "Код")	Возвращает симв. значение качественной характеристики для указанного артикула по коду
7	Получить значение характ. по коду и артикулу	GETNOMITEM_ART ("Артикул", "Код")	Возвращает значение качественной характеристики для указанного артикула по коду
8	Получить значение характеристики по коду	GETNOMITEM ("Код")	Функция возвращает значение качественной характеристики по трехсимвольному коду
9	Получить значение характеристики по коду	GETNOMITEM_S ("Код")	Функция возвращает симв. значение качественной характеристики по трехсимвольному коду
10	Установить значение характ. парт. по коду и артик.	SETPRTITEM ("Артикул", "Код", Значение)	Устанавливает значение указанной характеристики партии
11	Установить значение характ. по коду и артикулу	SETNOMITEM ("Артикул", "Код", Значение)	Устанавливает значение указанной качественной характеристики
	Функции работы с печатной формой 1194		

1	Функция получения единицы измерения	GET_EI_POS(Тип,"Код подр.",ID прод.,ID заказа,Дат нач.,Дат оконч.)	Функция возвращает единицу измерения позиции (продукции/материала). Тип (0-продукция, 1-материал). Для материала ID заказа = 0.
2	Функция получения количества материала	GETQT_PER_M(Тип, "Код подр.",ID ном.,Дат нач.,Дат оконч.)	Функция возвращает количество материала за период. Тип (0-норма, 1-факт).
3	Функция получения количества продукции	GETQT_PER_P(Тип,"Код подр.",ID ном.,ID заказа,Дат нач.,Дат оконч.)	Функция возвращает количество продукции за период. Тип (0-стандарт, 1-брак, 2-задание).
Функции работы с прейскурантами			
1	Дать дату установки цены товара по прейскуранту	PRC_GETDATCN ("Прейс.",ID тов.,"П","Д")	Функция возвращает дату установки цены,полученной функцией PRC_GETCN
2	Дать последнюю цену товара по прейскуранту	PRC_GETCN ("Прейс.",ID Товара,"П",Д)	Функция возвращает цену по прейскуранту на дату. Вход: Шифр прейс., ID товара, Шифр партии,Дата. Выход:Цена
3	Дать складскую цену на дату	PRC_GETSKLCN (ID товара, "СКЛАД", "ПАРТИЯ", Дата)	Функция возвращает складскую цену на дату. ВХОД: ID товара, Шифр склада, Шифр партии, Дата Выход: Цена
4	Настройка прейскурантов по покупателям	PRC_SETGRPCFG ("Тип ном.", "Код группы", "Артикул", Дата, "Прейс.")	Функция создает запись в таблице "Настройки прейскурантов по покупателям", или корректирует существующую.

5	Настройка прейскурантов по покупателям	PRC_SETGRPCFGK(Тип. ном., "Код группы", "Артикул", Дата, "Прейс.", "Покупатель"	Функция создает запись в таблице "Настройки прейскурантов по покупателям", или корректирует существующую.
6	Получить ЕИ позиции прейскуранта	PRC_GETEI ("Прейс.", ID Товара, "П")	Функция возвращает ЕИ позиции прейскуранта. Вход: Шифр прейск., ID товара, Шифр партии. Выход: ЕИ
7	Получить прейскурант настроенный для склада	PRC_GETSTORCFG ("Код склада")	Функция возвращает прейскурант настроенный для склада
8	Получить прейскурант покупателя	PRC_GETGRPCFG ("Тип ном.", "Код группы", "Артикул", Дата, "Код контрагента")	Функция возвращает код прейскуранта покупателя
9	Установить цену по позиции прейскуранта на дату	PRC_SETDATCN ("Прейс.", ID товара, "Партия", Дата, Цена)	Функция устанавливает цену товара из прейскуранта за указанную дату.
10	Установить цену товара из прейскуранта на дату по	PRC_SETDATCN_N("Прейс.", ID товара, "Партия", Дата, Цена)	Функция устанавливает цену товара из прейскуранта за указанную дату по точному совпадению описательного ключа.
11	Чтение тарифа скидки контрагенту на дату	READ_PTN_TRF("Код контрагента", Дата, "Тип тарифа")	Чтение тарифа скидки контрагенту на дату
	Функции работы с регламентированной отчетностью		

1	Выбрать параметры расчета	F_ADJPRMSEL()	<p>Окно выбора реквизитов регламентированной отчетности</p> <p>ВЫХОД: Будут заполнены поля: SRHist_ShblRcd - КОД_ШАБЛОНА SRHist_Rcd - КОД_РАСЧЕТА SRHist_Correction - КОД_КОРРЕКТИРОВКИ</p> <p>Получить значения можно по GetCargoVal(ИМЯ_ПОЛЯ)</p>
Функции работы со складскими остаткам			
1	Загрузить остатки в TempDocS	LOAD_OST(Режим)	<p>Загрузить остатки в TempDocS (TDocS_ТурHdr = /*hTrdCfgSpcObr 53*/)</p> <p>Режим (0 - очистить, 1 - загрузить/*обороты, остатки*/ , 2 - загрузить/*поступление, выбытие*/)</p>
2	Получить остаток на дату	GET_OST_QT(ID Карточки, Дата)	Функция возвращает остаток по складской карточке на заданную дату
3	Получить цену остатка	GET_OST_CN (ID карточки, Дата)	Функция возвращает цену по складской карточке на заданную дату
4	Установка фильтра даты для загрузки остатков в Тем	SET_OST_FLTR_DAT(Начало периода,Конец периода,Начало время,Конец время)	Установка фильтра даты для загрузки остатков в TempDocS
5	Установка фильтра движения для загрузки остатков	SET_OST_FLTR_ADV(вид цены,исключить внутреннее перемещение,вид операции,приход/р	Установка фильтра движения для загрузки остатков
6	Установка фильтра по владельцам для загрузки остат	SET_OST_FLTR_OWNER(поставщик,получатель,ID контрагента,тип контрагента)	Установка фильтра по владельцам для загрузки остатков

7	Установка фильтра по номенклатуре для загрузки ост	SET_OST_FLTR_NO M(номенклатура,группа,партия)	Установка фильтра по номенклатуре для загрузки остатков
8	Установка фильтра по ОУ для загрузки остатков	SET_OST_FLTR_OU(объект учета,тип объекта учета,счет)	Установка фильтра по ОУ для загрузки остатков
9	Установка фильтра по складам для загрузки остатков	SET_OST_FLTR_STORAGE(группа мест хранения,склад)	Установка фильтра по складам для загрузки остатков
	Функции расчёта ведомостей калькуляции		
	Расчётные функции		
1	Вернуть объём продукции по объекту	CALC_OBJ_QNT("Группа","Продукц.,""Подразд","Заказ",ДатаНач,ДатаКон,ЕДИзм,Парам)	Функция возвращает объём выпущенной продукции за период в указанной единице измерения. Параметр: 0-всё,1-стандарт,2-брак.
2	Вернуть плановую стоимость израсх. материалов	MAT_EXP_AMOUNT("КодПродукции","ЕДИзм",Дата,Парам)	Функция возвращает стоимость израсходованных материалов на выпуск единицы продукции по плановым ценам на указанную дату.
3	Вернуть полную сумму трудозатрат по нарядам	SAL_EXP_ALL(ДатаНач,ДатаКон,"КодГр","КодПрод","КодПодр","КодЗак")	Вернуть полную сумму заработной платы по нарядам за период по заданной комбинации кодов группы и типа продукции, подразделения и заказа.
4	Вернуть стоим. остатков материалов в производстве	MAT_BALANCE("КодПодр","КодГрНом","КодНом",Дата,ТипЦены,Параметр)	Вернуть стоимость остатков материалов по заданному подразделению на дату. Тип цены: 0-склад.себестоим.,1-плановая цена Параметр: 0-все,1-незавершен.произв.,2-натур.остатки.

5	Вернуть сумму затрат на материалы	МАТ_EXP(ТипОбъекта,"Код",ДатаНач,ДатаКон,ТипЦены,ТипСырья)	Вернуть сумму затрат на материалы по указанному типу объекта с заданным кодом за заданный период.
6	Вернуть сумму затрат на материалы EXP	МАТ_EXPEX("КдГрп", "КдНом", "КдПрд", "КдЗак", ДатаНач, ДатаКон, ТипЦены, ТипСырья)	Вернуть сумму затрат на материалы по указанному кодам группы, номенклатуры, подразделения и заказа по заданому периоду Тип цены - 0-складская себестоимость, 1-плановая цена.
7	Вернуть сумму затрат на материалы на количество	МАТ_EXP_QT(ДатаНач,ДатаКон,ТипОбъекта,"Код",ТипЦены,ВидПрод,"КодЕИ",ТипСырья,Кол-во)	Вернуть сумму затрат на материалы по указанному типу объекта с заданным кодом за заданный период для заданного количества продукции.
8	Вернуть сумму затрат на отходы	WAST_EXP(ТипОбъекта,"Код",ДатаНач,ДатаКон,ТипЦены)	Вернуть сумму затрат на отходы по указанному типу объекта с заданным кодом за заданный период
9	Вернуть сумму статьи плановой калькуляции из архив	GET_ARCPLNSUM("КодАрхива", "КодНом", "КодСтатьи")	Вернуть сумму статьи плановой калькуляции из архива. В качестве КодНом может передаваться код номенклатуры или ОПА продукции.
10	Вернуть сумму статьи плановой калькуляции по дате	GET_DATPLNSUM(Дата, "КодНом", "КодСтатьи")	Вернуть сумму статьи плановой калькуляции на дату. В качестве КодНом может передаваться код номенклатуры или ОПА продукции.

11	Вернуть сумму трудозатрат	SAL_EXP(ДатаНач,ДатаКон,ТипОбъекта,Код)	Вернуть сумму заработной платы за период по заданному типу объекта с указанным кодом.
12	Вернуть сумму трудозатрат на количество	SAL_EXP_QT(ДатаНач,ДатаКон,ТипОбъекта,"Код",ВидПрод,"КодЕИ",Кол-во)	Вернуть сумму трудозатрат на продукцию по указанному типу объекта с заданным кодом за заданный период.
13	Вернуть сумму трудозатрат по нарядам за период	SAL_EXP_NORM(ДатаНач,ДатаКон,"КодГр","КодПрод","КодПодр","КодЗак")	Вернуть сумму заработной платы по нарядам за период по заданной комбинации кодов группы и типа продукции, подразделения и заказа.
14	Вернуть сумму ШПЗ по видам оплат	GET_PAY_SUM_VO(ДатаПериода,"КодЗаказа","КодПодр",СписокВидовОплат)	Вернуть сумму ШПЗ по указанным параметрам за заданный период по заданному списку видов оплат. Список видов оплат задаются либо строкой с кодами ВО через пробел, либо кодом списка ВО.
15	Вернуть сумму ШПЗ по предприятию	GET_PAY_SUM(ДатаПериода,"КодЗаказа","КодПодр")	Вернуть сумму ШПЗ по указанному подразделению на указанный заказ за заданный период.
	Функции для работы в базах распределения		
1	Вернуть ID базы распределения	GET_BASE_ID("Код базы",ДатаПериода)	Функция возвращает ID базы распределения по заданным параметрам: "Код базы" - строковый код шаблона базы распределения ДатаПериода - дата начала периода искомой базы распределения.

2	Вернуть ID первой базы распределения	GET_BASE_FIRST(НомерЦикла,"Код базы")	Функция возвращает ID первой базы распределения по заданным параметрам: НомерЦикла - число от 0 до 255 включительно для организации вложенных и параллельных циклов перебора записей "Код базы" - строковый код шаблона базы распределения.
3	Вернуть ID следующей базы распределения	GET_BASE_NEXT(НомерЦикла,"Код базы")	Функция возвращает ID следующей базы распределения по заданным параметрам: НомерЦикла - число от 0 до 255 включительно для организации вложенных и параллельных циклов перебора записей "Код базы" - строковый код шаблона базы распределения. Функция должна вызываться после функции GET_BASE_FIRST.
4	Вернуть дату базы распределения	GET_BASE_DAT(ID_базы)	Функция возвращает дату ввода или модификации базы распределения по заданным параметрам: ID_базы - числовой идентификатор базы распределения, возвращаемый функциями GET_BASE_FIRST, GET_BASE_NEXT и GET_BASE_ID.

5	Вернуть дату периода базы распределения	GET_BASE_PER(ID_базы)	Функция возвращает дату периода базы распределения по заданным параметрам: ID_базы - числовой идентификатор базы распределения, возвращаемый функциями GET_BASE_FIRST, GET_BASE_NEXT и GET_BASE_ID.
6	Вернуть значение базы распределения	CALC_BASE_VAL("Код базы",Дата,"Код объекта")	Функция возвращает значение указанной базы распределения на заданный период по заданному объекту применимости.
7	Вернуть итог по базе распределения	GET_BASE_TOTAL("Код базы",Дата)	Функция возвращает итоговую сумму по указанной базе распределения на заданный период.
8	Вернуть код первого объекта базы распределения	GET_BOBJ_FIRST(ID_базы)	Функция возвращает строковый код первого объекта применимости по заданным параметрам: ID_базы - числовой идентификатор базы распределения, возвращаемый функциями GET_BASE_FIRST, GET_BASE_NEXT и GET_BASE_ID.

9	Вернуть код следующего объекта базы распределения	GET_BOBJ_NEXT(ID_базы)	Функция возвращает строковый код следующего объекта применимости по заданным параметрам: ID_базы - числовой идентификатор базы распределения, возвращаемый функциями GET_BASE_FIRST, GET_BASE_NEXT и GET_BASE_ID.
Функции для работы с ведомостями калькуляции			
1	Вернуть сумму по статье калькуляции	CALC_ST_SUM("Код ведомости",Дата,"Код объекта","Код статьи")	Функция возвращает сумму по статье калькуляции для заданного объекта по указанной ведомости за заданный период.
Функции расчёта оборотов и баланса			
1	Дата вступительного баланса	GET_DATE_VBL(Код_книги)	Функция получения даты начала вступительного баланса по указанной бухгалтерской книге.
2	Дата начала расчётов	GET_DATE_BEGIN()	Функция получения даты начала расчётов (дату, на которую введены вступительные остатки).
Функции расчётных таблиц			
Дополнительные функции			
1	Пересчитать таблицу	FTBL_RUN(ИД_ФинТаблицы)	Запускает перерасчет финансовой таблицы с указанным ID. Возвращает 1 в случае удачного завершения операции и 0 в случае ошибки. ВНИМАНИЕ! Функция не предназначена для использования в модуле "Бюджеты".

2	Получить сумму указанных ячеек колонки	FTBL_C_SUM(ID_таблицы,"КодКол","Шаблон")	Возвращает сумму всех ячеек указанной колонки, по заданному шаблону. Шаблон включает в себя перечень кодов строк (через запятую) или диапазонов строк ("начало...конец").
3	Получить сумму указанных ячеек строки	FTBL_L_SUM(ID_таблицы,"КодСтр","Шаблон")	Возвращает сумму всех ячеек указанной строки, по заданному шаблону. Шаблон включает в себя перечень кодов колонок (через запятую) или диапазонов колонок ("начало...конец").
4	Суммировать колонки таблицы	FTBL_C_TOT(ID_таблицы,"КодКол","Шаблон")	Суммирует соответствующие ячейки всех колонок по заданному шаблону и помещает результаты в ячейки указанной колонки. Шаблон включает в себя перечень кодов колонок (через запятую) или диапазонов колонок ("начало...конец").
5	Суммировать строки таблицы	FTBL_L_TOT(ID_таблицы,"КодСтр","Шаблон")	Суммирует соответствующие ячейки всех строк по заданному шаблону и помещает результаты в ячейки указанной строки. Шаблон включает в себя перечень кодов строк (через запятую) или диапазонов строк ("начало...конец").
	Функции для колонок и строк фин. таблиц		

1	Блокировать шаблон	FTBL_EDT_BEG(ID_или_Код, Номер_Цикла_или_ 0)	<p>Блокирует указанный шаблон фин. таблиц, а также все построенные таблицы по указанному шаблону. По завершению работы с шаблоном необходимо обязательно вызвать FTBL_EDT_END.</p> <p>Возврат : 1 - удачно, 0 - ошибка.</p>
2	Добавить колонку в шаблон фин. таблиц	FTBL_ADD_COL("Код", "Наим", ТипДанн, ДлнТчн, Ручн, Тип, НомерЦикла)	<p>Добавить колонку в ранее заблокированный шаблон. Параметры : "Код" - код добавляемой колонки "Наим" - наименование ТипДанн - тип данных : 0 - строка, 1 - число, 2 - дата, 3 - время; ДлнТчн - длина/точность в зависимости от типа данных Ручн - флаг пересчета данных, введенных вручную Тип - тип колонки : 0 - расчетная,</p>

3	Добавить строку в шаблон фин. таблиц	FTBL_ADD_ROW("Код", "Наим", Вал, ТипСтр, КодАнл, КонтрСч, Номер_Цикла)	Добавляет строку в ранее заблокированный шаблон финансовых таблиц. Требует FTBL_EDT_BEG. Параметры : "Код" - код добавляемой строки "Наим" - наименование строки Вал - код или ID валюты строки ТипСтр - тип строки : 0 - детальная, 1 - итоговая КодАнл - код аналитической картотеки или 0 КонтрСч - код или ID контрольного счета Номер_Цикла - номер цикла или 0 Возвращает : 1 - удачно, 0 - ошибка.
4	Получить ID аналитической картотеки строки таблицы	FTBL_L_TYP(ID_таблицы,"КодСтроки")	Функция возвращает тип строки: 0 - не итоговая 1 - итоговая
5	Получить код последней ошибки	FTBL_GETLERCD()	Возвращает код последней ошибки семейства функций для строк и колонок шаблонов фин. таблиц.
6	Разблокировать шаблон	FTBL_EDT_END(Номер_Цикла_или_0)	Функция производит разблокировку шаблона и таблиц в ранее заблокированном цикле. Требует FTBL_EDT_BEG. Возврат : 1 - удачно, 0 - ошибка.
7	Сбросить признак ручного изменение ячейки таблицы	FTBL_C_CLRCNGP("ID_таблицы", "КодСтроки", "КодКолонки", IDАналитики)	Функция возвращает 1, если признак модификации успешно сброшен. Иначе - 0.

8	Удалить аналит. карточку из строки фин. таблицы	FTBL_A_DEL_ROW(ИД_Таблицы, "Код_Строки", ИД_АнлКарточки, ИД_Цикла)	<p>Функция удаляет аналитическую карточку из строки финансовой таблицы.</p> <p>Параметры : ИД_Таблицы - ИД финансовой таблицы "Код_Строки" - строковый код обрабатываемой строки ИД_АнлКарточки - ИД аналитической карточки ИД_Цикла - номер цикла в пределах которого выполняется операция</p> <p>Возврат : 1 - успешное завершение, 0 - ошибка.</p> <p>В случае ошибки можно воспользоваться функцией FTBL_GETLERCD для получения разъяснен</p>
9	Удалить колонку из шаблона фин. таблиц	FTBL_DEL_COL(КодИлиНомерПоПорядку, НомерЦикла)	<p>Удаляет колонку из ранее заблокированного шаблона и всех фин. таблиц, которые были по этому шаблону построены.</p> <p>Возвращает : 1 - удачно, 0 - ошибка.</p> <p>Требуется FTBL_EDT_BEG.</p>
10	Удалить строку из шаблона фин. таблиц	FTBL_DEL_ROW(КодИлиНомерПоПорядку, НомерЦикла)	<p>Удаляет строку по указанному строковому коду или номеру по порядку из ранее заблокированного шаблона.</p> <p>Возвращает : 1 - удачно, 0 - ошибка.</p> <p>Требуется FTBL_EDT_BEG.</p>
	Функции доступа к ячейкам и свойствам таблицы		

1	Дать код аналитики (из кэша)	FTBL_ANL_CD(ИД таблицы, ИД аналитики, Уровень)	Возвращает код аналитики из кэша таблицы
2	Дать наименование аналитики (из кэша)	FTBL_ANL_NM(ИД таблицы, ИД аналитики, Уровень)	Возвращает наименование аналитики из кэша таблицы
3	Подготовить среду к вычислениям	FTBL_PREPARE()	Подготовить среду к вычислениям. Функция рекомендована для каждого запуска программы фин. таблиц.
4	Получить ID аналитики строки таблицы по кодам	FTBL_ANL_ID(Таб,"Стр","K1","K2","K3","K4","K5","K6","K7","K8","K9","K10")	Возвращает ID аналитической карточки по заданной строке указанной таблицы и кодами аналитики.
5	Получить ID аналитической картотеки строки таблицы	FTBL_L_ANL(ID_таблицы,"КодСтроки")	Функция возвращает ID аналитической картотеки, привязанной к строке таблицы.
6	Получить ID контрольного счета строки таблицы	FTBL_L_SCH(ID_Таблицы, "Код_Строки")	Функция возвращает ID контрольного счета строки шаблона финансовых таблиц.
7	Получить год таблицы	FTBL_GET_YEAR(ID_таблицы)	Функция возвращает год таблицы (если этот параметр включен для таблицы).
8	Получить дату создания таблицы	FTBL_GET_DATE(ID_таблицы)	Функция возвращает дату создания указанной таблицы.
9	Получить значение ячейки таблицы	FTBL_GET_SM(ID_таблицы,"КодСтр","КодКол",ID_аналит)	Функция возвращает значение ячейки из расчётной таблицы по указанным параметрам.
10	Получить значение ячейки таблицы	FTBL_GET_SMCASH(ID_таблицы,"КодСтр","КодКол",ID_аналит)	Функция возвращает значение ячейки из кэша расчётной таблицы по указанным параметрам.

11	Получить квартал таблицы	FTBL_GET_QRT(ID_таблицы)	Функция возвращает номер квартала таблицы (если этот параметр включен для таблицы).
12	Получить множитель строки таблицы	FTBL_L_MUL(ID_таблицы, "КодСтроки")	Функция возвращает значение множителя строки таблицы.
13	Получить наименование колонки таблицы	FTBL_C_NM(ID_Таблицы, "КодКолонки")	Получить наименование колонки таблицы по ее коду или "".
14	Получить наименование строки таблицы	FTBL_L_NM(ID_таблицы, "КодСтроки")	Функция возвращает наименование строки таблицы.
15	Получить номер таблицы	FTBL_GET_NMR(ID_таблицы)	Функция возвращает номер таблицы (если этот параметр включен для таблицы).
16	Получить период таблицы	FTBL_GET_PER(ID_таблицы)	Функция возвращает дату начала периода таблицы (если этот параметр включен для таблицы).
17	Получить подразделение таблицы	FTBL_GET_PDR(ID_таблицы)	Функция возвращает код подразделения таблицы (если этот параметр включен для таблицы).
18	Получить строку комментария к таблице	FTBL_GET_NOTE(ID_таблицы)	Функция возвращает строку комментария к таблице.
19	Получить тип данных колонки таблицы	FTBL_C_TP(ИД_Таблицы, "Код_Колонки")	Функция возвращает тип данных колонки таблицы. Коды возврата: 1 - строковый тип 2 - числовой тип 3 - тип даты 4 - тип времени

20	Проверить колонку таблицы на вводимость	FTBL_C_ISINP(ID_таблицы,"КодКолонки")	Функция возвращает 1 если колонка вводимая, и 0 - если расчётная.
21	Проверить ручное изменение ячейки в аналитике	FTBL_C_CNGANL("КодСтроки", "КодКолонки", IDАналитики)	Функция возвращает 1, если значение менялось. Иначе - 0.
22	Проверить строку таблицы на итоговость	FTBL_L_ISTOT(ID_таблицы,"КодСтроки")	Функция возвращает 1 если указанная строка итоговая, и 0 - если строка детальная.
23	Установить дату создания таблицы	FTBL_SET_DATE(ID_таблицы, Дата)	Функция устанавливает дату создания указанной таблицы.
24	Установить значение внешней таблицы	FTBL_SET_SMX(ID_таблицы,"КодСтр","КодКол",ID_аналит,Значение)	Функция устанавливает значение ячейки заданной расчётной таблицы по указанным параметрам.
25	Установить значение ячейки таблицы	FTBL_SET_SM("КодСтр","КодКол",ID_аналит,Значение)	Функция устанавливает значение ячейки текущей расчётной таблицы по указанным параметрам.
26	Установить строку комментария к таблице	FTBL_SET_NOTE(ID_таблицы, Строка комментария)	Функция устанавливает строку комментария к таблице.
Функции навигации по таблице			
1	Получить код первой колонки таблицы	FTBL_C_FIRST(ID_таблицы)	Функция возвращает код первой колонки расчётной таблицы.
2	Получить код первой строки таблицы	FTBL_L_FIRST(ID_таблицы)	Функция возвращает код первой строки расчётной таблицы.
3	Получить код следующей колонки таблицы	FTBL_C_NEXT(ID_таблицы)	Функция возвращает код следующей колонки расчётной таблицы. Требуется предварительного вызова функции FTBL_C_FIRST.

4	Получить код следующей строки таблицы	FTBL_L_NEXT(ID_таблицы)	Функция возвращает код следующей строки расчётной таблицы. Требуется предварительного вызова функции FTBL_L_FIRST.
Функции перебора аналитических карточек			
1	Получить ID первой аналит. карточки по строке	FTBL_A_FIRST(ID_таблицы,"КодСтр")	Возвращает ID первой аналитической карточки по заданной строке указанной таблицы.
2	Получить ID следующей аналит. карточки по строке	FTBL_A_NEXT(ID_таблицы,"КодСтр")	Возвращает ID следующей аналитической карточки по заданной строке указанной таблицы.
Функции справочника "Структурные единицы"			
1	Показать диалог выбора структурной ед. с фильтром	F_STEDLG_FLT("НАИМЕНОВАНИЕ ПОЛЯ КОДА ФИЛЬТРА","ТАБЛИЦА ФИЛЬТРА",ВЫБОР_НЕСКОЛЬКИХ	Показать диалог выбора структурной единицы с фильтром доступа ВХОД: "НАИМЕНОВАНИЕ ПОЛЯ КОДА ФИЛЬТРА" - имя поля в переданной таблице фильтра(игнорируется если "ТАБЛИЦА ФИЛЬТРА"=NULL); "ТАБЛИЦА ФИЛЬТРА"- имя таблицы содержащей список доступных для выбора структурных единиц (остальные будут невыбираемы и засерены) Если NULL - то все записи активны ; "НАИМЕНОВАНИЕ ПОЛЯ КОДА МАРОК" - имя поля в переданной таблице марок(игнорируется если "ТАБЛИЦА МАРОК"=NULL);

2	Получение структурной единицы по Cd или Rcd	F_SYSSTE_GET("Rcd", "Код", "Наименование", "Код родителя", "Флаг поиска")	Получение структурной единицы по Cd или Rcd (флаг поиска 0 - по Rcd / 1 - по коду) ВЫХОД : Код/Наименование/Код родителя ВОЗВРАТ: Rcd структурной единицы или 0 в случае неудачи
3	Функции чтения и установки атрибутов		
4	Дать значение атрибута по наименованию	GETNOMATRNM (ID товара, "Наименование")	Функция возвращает значение атрибута по наименованию атрибута.
5	Дать значение атрибута по номеру	GETNOMATR (ID товара, Номер атрибута)	Функция возвращает значение атрибута товара по номеру.
6	Дать значение характеристик. арт. по наименованию	GETNOMARTDSC(ID товара, "Наименование")	Дать значение характеристики артикула по наименованию
7	Дать наименование атрибута по номеру	GETATRNM (ID товара, Номер атрибута)	Функция возвращает наименование атрибута по номеру.
8	Получить ID - товара	GETRCDOPA("Артикул", "Атрибут1,Знач. атрибута1...")	Функция GETRCDOPA возвращает ID товара для артикула с указанными атрибутами.
9	Установить значение атрибута по наименованию	SETNOMATRNM (ID Товара, "Наименование", "Значение")	Функция устанавливает значение атрибута товара по наименованию атрибута и возвращает новый ID товара
10	Установить значение атрибута по номеру	SETNOMATR (ID Товара, Номер атрибута, "Значение")	Функция устанавливает значение атрибута товара по номеру и возвращает новый ID товара.
	Функции чтения реквизитов договоров		

1	Зачитать реквизит операции договора	FR_DOGGETRIVAL(ИД договора, номер операции, Тип реквизита)	Функция возвращает значение указанного реквизита операции договора.
2	Получить значение реквизита операции договора по н	FR_DOGGETOPRIVAL(ИД договора, ИД операции, наименование реквизита)	Функция возвращает значение указанного реквизита операции договора.
3	Получить код группы договора по ИД	FR_GETDOGGRPCD(ИД договора)	Функция по ИД группы договора возвращает ее код.
Функции чтения реквизитов торговых документов			
1	Читать реквизиты заказа по ИД	READ_ZKG(ИД заказа)	Читать реквизиты заказа по ИД.
2	Читать реквизиты накладной по ИД	READ_NAK(ИД накладной)	Читать реквизиты накладной по ИД.
3	Читать реквизиты счета по ИД	READ_ACC(ИД счета)	Читать реквизиты счета по ИД.
Строк: 1237			