

<b>11</b>	<b>ПРИЛОЖЕНИЯ</b> .....	<b>2</b>
11.1	ПРИМЕР СОЗДАНИЯ ТИПОВОЙ ПРОВОДКИ ПРОГРАММНЫМ СПОСОБОМ .....	2
11.2	ПРИМЕР СОЗДАНИЯ ПЕЧАТНОЙ ФОРМЫ ДОКУМЕНТА .....	6
11.3	ПРОГРАММНЫЕ ЭЛЕМЕНТЫ, ДОСТУПНЫЕ ИЗ <i>VBSCRIPT</i> .....	11
11.3.1	<i>Классы общего назначения</i> .....	11
11.3.2	<i>Классы системы, используемые при написании формулы расчета прейскуранта</i> .....	12
11.3.3	<i>Классы, используемые при написании алгоритмов типовых проводок</i> .....	14
11.3.4	<i>Классы системы, используемые при написании алгоритмов расчета в заработной плате</i> .....	27
11.3.5	<i>Классы системы, используемые при написании алгоритмов расчета амортизации в подсистеме Имущество</i> .....	32
11.3.6	<i>Классы системы, используемые при написании алгоритмов расчета налогов в Мастере расчета суммы</i> 34	
11.4	НАСТРОЙКА ВИДОВ ОПЛАТ «Ссуда», «ПРОЦЕНТЫ ПО ССУДЕ», «МАТЕРИАЛЬНАЯ ВЫГОДА» .....	38
11.4.1	<i>Ссуда</i> .....	38
11.4.2	<i>Проценты по ссуде</i> .....	40
11.4.3	<i>Материальная выгода</i> .....	40
11.5	СИСТЕМНЫЕ АЛГОРИТМЫ РАСЧЕТОВ ЕНВД и УСН .....	41
11.5.1	<i>Системный алгоритм расчета единого налога на вмененный доход</i> .....	41
11.5.2	<i>Системный алгоритм расчета единого налога по упрощенной системе налогообложения</i> .....	41

## 11 Приложения

### 11.1 Пример создания типовой проводки программным способом

Наряду с табличным способом создания типовых проводок можно использовать программный способ. По сравнению с табличным, программный способ является более трудоемким и требует знания основ программирования на языке *Basic*, однако дает большую свободу при описании типовых проводок. С другой стороны, при использовании программного способа возрастает вероятность появления ошибок, поэтому рекомендуется по возможности пользоваться табличным способом и прибегать к программному способу только в сложных случаях, когда возможностей табличного способа оказывается недостаточно<sup>1</sup>.

В качестве примера рассмотрим типовую системную проводку *Предоплата партнерам (Т)*. Буква Т в скобках указывает на то, что проводка формировалась табличным способом. Форма строки этой табличной проводки имеет вид:

Окно строки типовой проводки.

Дебетовый счет взят из документа, в качестве кредитового используется счет кассы/банка. По дебету настроены три уровня аналитики: по партнеру из документа, по документу-основанию и по статьям затрат. Формула расчета суммы имеет вид:

$$\text{Posting.SummaPosting} = \text{Payment.SummaDoc}$$

<sup>1</sup> Табличное представление проводки имеет программный эквивалент. В этом легко убедиться, изменив для табличной проводки вид с *Табличного* на *Программный*.

Таким образом, форма строки проводки содержит все необходимые данные для формирования проводки: дебетовый и кредитовый счета, сумму проводки и аналитику. При этом счета и аналитика задаются из окон выбора. Программирование сводится к работе с Мастером формул для задания алгоритма расчета суммы проводки.

При программном способе формирования проводки все приходится программировать самостоятельно. Для данной типовой проводки рассмотрим программу создания проводки. Для этого вернемся в форму *Типовая проводка* и изменим вид проводки с *Табличной* на *Программную*. При этом вкладка *Проводки* изменит свое название на *Программа*, и в появившемся окне будет отображен программный код, соответствующий рассмотренному табличному представлению.

Разберем полученный код. Предварительно отметим, что конечным результатом работы кода является задание дебетового и кредитового счетов и формулы для вычисления суммы проводки. Также должна быть описана аналитика и определены некоторые свойства экземпляра класса Posting (этот класс служит для описания проводок в системе).

```
' объявляется вспомогательная переменная mCodeAccount
dim mCodeAccount

' инструкция With позволяет избежать повторного написания имени класса Posting
With Posting

    ' метод Clear очищает проводку
    .Clear

    ' остатков нет
    .IsRest = False

    ' счет не является забалансовым
    .IsOutBalance = False

    ' курс валюты проводки устанавливается равным курсу валюты документа к валюте
    ' учета (класс Payment служит для описания документа)
    .RatePosting = Payment.RateAccount

    ' признак деления на курс устанавливается равным признаку деления на курс для
    ' суммы в валюте учета
    .IsDivide = Payment.IsDivideAccount

    ' дата проводки устанавливается равной дате проведения документа
    .DatePosting = Payment.DateOper

    ' валюта проводки устанавливается как валюта документа
    .IDCurrency = Payment.IDCurrency

    ' примечание к проводке
    .CommentPosting = "Перечисление денежных средств"

    ' сумма проводки берется из соответствующего документа
    Posting.SummaPosting = Payment.SummaDoc

    ' сумма в валюте учета (SummaAccount) рассчитывается в зависимости от того, как
    ' был установлен признак деления на курс в документе
    If .IsDivide Then
        .SummaAccount = .SummaPosting / .RatePosting
    Else
        .SummaAccount = .SummaPosting * .RatePosting
    End If

    ' счет дебета проводки устанавливается равным корр. счету из документа
    ' (по условию)
    .DT_IDAccount = Payment.idAccount

    ' обрабатывается дебетовая аналитика
    ' в переменную mCodeAccount записывается код корр. счета из документа
    mCodeAccount = Payment.AccountCode

    ' проверка, установлен ли для этого кода признак ведения аналитического учета
    ' в бухгалтерии (бухгалтерию описывает класс Bookkeeping)
    if Bookkeeping.Analyst(mCodeAccount) then
```

```

' обрабатывается четыре уровня аналитики
' проверяется, присутствует ли аналитика первого уровня
if Bookkeeping.Center1(mCodeAccount) then
    ' по условию на первом уровне установлена аналитика по партнеру
    ' если существует код этого партнера в документе, то код аналитики
    ' первого уровня (DT_IDCenter1) берется из документа, в противном
    ' случае - из бухгалтерии по умолчанию
    if Payment.Partner.IdPartner > 0 then
        .DT_IDCenter1 = Payment.Partner.IdPartner
    else
        .DT_IDCenter1 = Bookkeeping.Center1Def(mCodeAccount)
    end if
end if

' проверяется, присутствует ли аналитика второго уровня
if Bookkeeping.Center2(mCodeAccount) then
    ' по условию на втором уровне установлена аналитика по документу
    ' если существует код этого документа-основания в документе,
    ' то аналитика второго уровня (DT_IDCenter2) берется из документа,
    ' в противном случае - из бухгалтерии по умолчанию
    if Payment.Dogovor.IdDogovor > 0 Then
        .DT_IDCenter2 = Payment.Dogovor.IdDogovor
    else
        .DT_IDCenter2 = Bookkeeping.Center2Def(mCodeAccount)
    end if
end if

' на третьем уровне установлена аналитика по умолчанию, поэтому проверяется
' только наличие аналитики на этом уровне, и если она есть - формируется
' аналитика по бухгалтерии
if Bookkeeping.Center3(mCodeAccount) then
    .DT_IDCenter3 = Bookkeeping.Center3Def(mCodeAccount)
end if

' проверяется, присутствует ли аналитика четвертого уровня
if Bookkeeping.Center4(mCodeAccount) then
    ' на четвертом уровне установлена аналитика по статьям затрат
    ' если существует код статей затрат в документе,
    ' то аналитика второго уровня (DT_IDCenter3) берется из статей
    ' затрат, в противном случае - из бухгалтерии по умолчанию
    if Payment.IDExpenses > 0 Then
        .DT_IDCenter4 = Payment.IDExpenses
    else
        .DT_IDCenter4 = Bookkeeping.Center3Def(mCodeAccount)
    end if
end if
end if

' кредитовый счет берется как счет учета кассы/банка (по условию)
.KT_IDAccount = Payment.MCenterIdAccount

' обрабатывается кредитовая аналитика
' переменной mCodeAccount присваивается значение кода счета учета кассы/банка
mCodeAccount = Payment.MCenterAccountCode

' кредитовая аналитика на всех уровнях установлена по умолчанию, поэтому
' проверяется только наличие аналитики в бухучете, и если она там присутствует -
' кредитовая аналитика для формируемой проводки берется из бухгалтерии
if Bookkeeping.Analyst(mCodeAccount) then
    if Bookkeeping.Center1(mCodeAccount) then
        .KT_IDCenter1 = Bookkeeping.Center1Def(mCodeAccount)
    end if
    if Bookkeeping.Center2(mCodeAccount) then
        .KT_IDCenter2 = Bookkeeping.Center2Def(mCodeAccount)
    end if
    if Bookkeeping.Center3(mCodeAccount) then
        .KT_IDCenter3 = Bookkeeping.Center3Def(mCodeAccount)
    end if

```

```
        if Bookkeeping.Center4(mCodeAccount) then
            .KT_IDCenter4 = Bookkeeping.Center4Def(mCodeAccount)
        end if
    end if
    ' сохранение типовой проводки
    .Save
End With
```

Для многострочной проводки программа будет состоять из нескольких последовательных фрагментов кода, каждый из которых должен быть составлен по правилам, рассмотренным выше для однострочной проводки.

Информацию о классах, их свойствах и методах, используемых при формировании типовых проводок, можно получить из нижнего окна Мастера формул (см. также Приложение 3).

## 11.2 Пример создания печатной формы документа

Создание любой новой печатной формы в БЭСТ-ОФИС можно разбить на 2 этапа:

1. Создание в *MS Excel* внешнего оформления печатной формы документа (шаблона).
2. "Привязка" полей созданного в *Excel* шаблона к параметрам программы БЭСТ-Офис.

Рассмотрим принцип создания любых печатных форм документов на примере счета на продажу.

### *Создание нового вида документа и печатной формы для него*

Для создания шаблона в *Excel* необходимо создать нужную форму, поименовав поля, в которые должны подтягиваться данные из программы.

Откройте программу *MS Excel*, войдите в режим создания нового документа и создайте следующую простейшую форму для счета:

1	A	B	C	D	E	F	G	H	I	J	K
2		Общество с ограниченной ответственностью "Свет"									
3		Адрес									
4		Телефон									
5		ИНН									
6											
7		Покупатель									
8		Адрес									
9		Телефон									
10		ИНН									
11											
12		Счет №				от					
13											
14		Наименование	Ед.изм.	Кол-во	Цена	Ст-ть					
15											
16			Сумма			0					
17											
18		Валюта									
19		Итого к оплате:									
20											
21		Менеджер									
22											
23											
24											
25											
26											
27											

Теперь можно начинать присвоение имен нужным полям этой формы. Осуществляется это с использованием комбинации клавиш **<Ctrl+F3>**. Поля могут именоваться произвольным образом, но не должно включать пробелов и специфических символов.

Например, чтобы присвоить имя полю C3, в которое из программы предполагается подтягивать адрес предприятия, установите курсор в эту ячейку C3, нажмите комбинацию клавиш **<Ctrl+F3>** и в появившемся окне *Присвоение имени* внесите первую запись: *ПродавецАдрес*.

Нажмите **ОК**.

**Примечание:**

Если имя поля состоит из нескольких слов, записывать его (имя) нужно без пробелов, начиная каждое слово с большой буквы или используя "подчеркивание" как знак разделения слов.

В ячейку С4 из программы должен подтягиваться телефон предприятия. Поэтому логично назвать его *ПродавецТелефон*, точно также установив курсор в ячейку С4 и нажав клавиши <Ctrl+F3>.

Введите наименования остальных полей формы в соответствии с таблицей:

Ячейка	Наименование поля
С5	<i>ПродавецИНН</i>
С7	<i>Покупатель</i>
С8	<i>ПокупательАдрес</i>
С9	<i>ПокупательТелефон</i>
С10	<i>ПокупательИНН</i>
С12	<i>СчетНомер</i>
G12	<i>СчетДата</i>
A15	<i>НомерПП</i>
B15	<i>Наименование</i>
B18	<i>Валюта</i>
С15	<i>НоменклатураЕДИзм</i>
D15	<i>НоменклатураКоличество</i>
E15	<i>НоменклатураЦена</i>
F15	<i>НоменклатураСтоимость</i>
С21	<i>Менеджер</i>
Строка 15	<i>Подробности</i>

Для того чтобы в дальнейшем в счет попадали все позиции спецификации из программы, необходимо строке 15 тоже присвоить имя. Для этого выделите ее (можно немного длиннее), нажмите комбинацию клавиш <Ctrl+F3> и внесите имя *Подробности*.

В ячейку F16 должна попадать суммарная стоимость всех позиций спецификации. Чтобы этого достичь, установите курсор в эту ячейку (F16) и нажмите комбинацию клавиш <ALT+>. В этой ячейке появится запись =СУММ(...) и ячейка будет отмечена пунктиром. Отметьте область суммирования, используя клавишу <Shift> и клавиши управления курсором "стрелка вверх/вниз". Эта область обязательно должна включать помимо ячейки F15 и ячейку F14 (с наименованием столбца *Ст-ть*). Отметив таким образом область суммирования, нажмите <Enter>.

В результате, если установить курсор в ячейку F16, в строке редактирования должна появиться запись: =СУММ(F14:F15).

**Созданный таким образом шаблон должен быть сохранен в специальной директории для документов системы *Template\Document*.**

В нашем случае эта директория содержится в папке *C:\Program Files\Intellect-Service\БЭСТ-ОФИС\*, в которую была установлена программа. Имя файла для сохранения шаблона может быть произвольным, но рекомендуется начинать его с префикса *User*, для того, чтобы в дальнейшем при обновлении системы эта форма не потерялась и не затерлась. Сохраните шаблон, назвав его, например, *User\_barSchet.xls*.

На этом работу по созданию шаблона простейшей печатной формы в *Excel* можно считать завершенной и переходить к следующему этапу – "привязке" именованных полей созданного таким образом шаблона к параметрам программы.

Для этого в подсистеме **Настройка** откройте пункт **Виды документов системы** и войдите в режим создания нового документа.

Заполните поля **Карточки вида документа** следующим образом:

В поле **Журнал** из справочника выберите запись **Счета на оплату**.

**Примечание:**

*Отнесение нового документа к тому или иному журналу определяет набор видов операций, которые могут быть связаны с создаваемым документом, и набор параметров программы БЭСТ-Офис, которые будут доступны для созданной в Excel печатной формы этого документа.*

В поле **Наименование** внесите запись *Образец*.

В поле **Код** – *ОБР*.

На страничке **Виды операций** отметьте позицию *Счет на оплату (продажа)*, поставив слева напротив этой позиции "галочку". Также в этой строке установите "галочки" для позиций **Автонумерация** и **Уникальность**.

Далее откройте страничку **Печатные формы** и нажмите пиктограмму "Создать строку".

На экране появляется окно **Настройка печатной формы**.

В поле **Наименование** запишите *Образец*.

В поле **Код** – *ОБР*.

Обратите внимание на "галочку" в позиции **Активная**. Она должна быть установлена.

**Примечание:**

*Печатная форма создаваемого документа с такой "галочкой" будет автоматически отмечена "флажком" на страничке **Печатные формы**, и, следовательно, именно она по умолчанию будет вызываться при печати. Это особенно актуально в том случае, если документ имеет несколько печатных форм, т.к. активной может быть только одна из них.*

Далее, в поле **Шаблон** выберите из предложенного списка созданную ранее для этого документа печатную форму – *User\_barSchet.xls*. После этого программа автоматически заполнит колонку **Поле шаблона** поименованными полями из созданного ранее шаблона *Excel*. Напротив каждого такого поля программа устанавливает значение *Без параметра*. Для того чтобы осуществить "привязку" **Полей шаблона** к **Параметрам программы**, необходимо в соответствующей позиции колонки **Параметр программы** открыть список доступных параметров и для конкретного **Поля шаблона** из списка выбрать тот **Параметр программы**, который должен импортироваться в это поле. Такую операцию необходимо последовательно проделать для каждого поля шаблона.

Осуществите эту "привязку" в соответствии с таблицей:

Поле шаблона	Параметр программы
Валюта	<i>Валюта</i>
Менеджер	<i>Менеджер</i>
Наименование	<i>Спецификация. Наименование</i>
НоменклатураЕдИзм	<i>Спецификация. Единица измерения</i>
НоменклатураКоличество	<i>Спецификация. Количество</i>
НоменклатураСтоимость	<i>Спецификация. Стоимость</i>
НоменклатураЦена	<i>Спецификация. Цена со скидкой</i>



НомерПП	Спецификация. НомерПП
Подробности	Спецификация.
Покупатель	Партнер.
ПокупательАдрес	Партнер. Адрес
ПокупательИНН	Партнер. ИНН
ПокупательТелефон	Партнер. Телефон
ПродавецАдрес	Предприятие. Адрес (по документу)
ПродавецИНН	Предприятие. ИНН
ПродавецТелефон	Предприятие. Телефон
СчетДата	Дата счета
СчетНомер	Номер документа

Нажмите кнопку **Настройка формы**. В открывшемся шаблоне на панели инструментов *MS Excel* откройте редактор *Visual Basic* (меню **Сервис – Макрос – Редактор Visual Basic**). С помощью команды **File – Import file** добавьте к проекту файл модуля *VBA Template\SystemTemplate\bowPropis.bas*. Данный файл находится в директории, в которой установлена программа БЭСТ-ОФИС (в рассматриваемом примере это *C:\Program Files\Intellect-Servise\*). После подключения указанного модуля становится доступной функция *SummaPropis*, позволяющая записывать суммы прописью. Сохраните внесенные изменения, воспользовавшись соответствующей пиктограммой в редакторе *Visual Basic*. Закройте окно *Visual Basic*.

На экране остался шаблон печатной формы, созданный в *MS Excel*. В ячейке C19 запишите следующую формулу:

`=SummaPropis(F16;Валюта)`

Нажмите **Enter**<sup>2</sup>. Сохраните изменения в шаблоне. Вернитесь в карточку настройки печатной формы, обновите поля, сохраните код, сохраните и закройте карточку.

#### **Примечание:**

При написании формулы в качестве разделителя аргументов необходимо использовать разделитель, указанный в региональной настройке *Windows*, в качестве разделителя элементов списка.

Сохраните и закройте **Карточку вида документа**.

Проверить работоспособность созданной таким образом печатной формы можно, если в подсистеме **Продажи** создать счет и в качестве вида документа выбрать созданный *Образец*. В качестве партнера выберите *ТОО "Восток"*. Заполните спецификацию, введя 2 позиции: *Диван – 1шт.* и *Блокнот – 100шт.* Пример такой печатной формы представлен на рисунке:

<sup>2</sup> Если при этом откроется окно *VBA* и появится сообщение *Compile error: User-defined type not defined*, то это не является признаком ошибки. Необходимые библиотеки типов будут подключены автоматически при формировании печатной формы. Нажмите **Ok** и закройте окно *VBA*. Если Вы все же хотите избежать появления этого сообщения, можно перед использованием функции *SummaPropis* подключить нужную библиотеку вручную. Для этого необходимо в редакторе *VBA* выбрать команду **Tools – References**, в появившемся окне установить галочку напротив библиотеки *Суммы/даты прописью (Версия 3.x)* и нажать **Ok**.

The screenshot shows a Microsoft Excel spreadsheet with the following data:

№п/п	Наименование	Ед.изм.	Кол-во	Цена	Ст-ть
1	Краска белая	шт	1	29,17	29,17
2	Краска синяя	кор.	1	1464,75	1464,75
		Сумма			1493,92

Additional data from the spreadsheet:

- Company: Общество с ограниченной ответственностью "Свет"
- Address: 121156 г. Москва, ул. Напрудная, д. 4
- Phone: (095) 365-94-90
- TIN: 7714169005
- Buyer: ООО "ДЕЛЬФА"
- Buyer Address: 129347, МОСКВА Г, Абрикосовский пер, д.1, корп.2, кв.3
- Buyer Phone: 7712045630
- Buyer TIN: 7712045630
- Account No: 1, Date: 06.06.02
- Currency: РУБ
- Total to pay: Одна тысяча четыреста девяносто три рубля 92 копейки
- Manager: Иванов П. В.

### Создание новой печатной формы для существующего вида документа

В целях экономии времени для выполнения этой операции воспользуемся созданным ранее шаблоном для счета на продажу.

Войдя в режим **Виды документов системы** подсистемы **Настройка**, откройте карточку того документа, для которого создается новая печатная форма (в нашем случае – *Сч/П Счет на продажу*).

Далее войдите в режим редактирования и откройте страничку **Печатные формы**.

Нажмите пиктограмму "Создать строку", клавишу **F7** или воспользуйтесь правой кнопкой мышки. Заполните форму **Настройка формы печати** как описано выше, в качестве шаблона выбрав созданный в Excel файл *User\_barSchet.xls*. Осуществите "привязку" полей шаблона и параметров программы (в соответствии с таблицей I-2). Сохраните введенную информацию.

После осуществления проделанных операций в списке печатных форм режима **Счета на продажу** подсистемы **Продажи** появилась новая запись – *Образец*. Чтобы распечатать счет в таком виде из списка печатных форм выбирается эта запись.

## 11.3 Программные элементы, доступные из *VBScript*

В тех случаях, когда пользователю предоставляется возможность создания собственных расчетных алгоритмов на языке *VBScript*, используются классы, дающие доступ из скрипта пользователя к программным средствам БЭСТ-ОФИС. Ниже приводится описание прикладных классов БЭСТ-ОФИС с доступными функциями (свойствами, методами)<sup>3</sup>.

### 11.3.1 Классы общего назначения

#### VBScript – стандартные функции *Visual Basic*

Свойство	Описание
Left(string, length)	Вырезает подстроку слева
Right(string, length)	Вырезает подстроку справа
Mid(string, start[, length])	Вырезает подстроку из строки
InStr([start, ]string1, string2)	Позиция подстроки string2 в строке string1
Trim(string)	Удаляет пробелы в строке слева и справа
IIF(expr, truepart, falsepart)	Условный оператор
Val(string)	Преобразует строку в число
Str(number)	Преобразует число в строку
FirstYearDate(YourDate)	Дата начала года (формат #mm/dd/yyyy#)
FirstQuarterDate(YourDate)	Дата начала квартала (формат #mm/dd/yyyy#)
FirstMonthDate(YourDate)	Дата начала месяца (формат #mm/dd/yyyy#)
mRound(number, decimal places)	Округление
mDiv(number, decimal places)	Округление в меньшую сторону

#### GlobVars – общие сведения о предприятии

Свойство	Описание
CodeYear	Год начала работы
NumMonth	Месяц начала работы
GlobalAcctCurrency	Валюта учета. ID
GlobalAcctCurrencyCode	Валюта учета. Код
FullName	Наименование полное
INN	ИНН
ОКПО	ОКПО
IsBookKeeping	Ведение бухгалтерии
Name	Наименование
NameBoss	Фамилия директора
PostBoss	Должность директора

<sup>3</sup> Жирным шрифтом выделены объекты, которые сами содержат функции (методы, события). Описания таких объектов вынесены в отдельные таблицы.

NameFBoss	Фамилия фин.директора
PostFBoss	Должность фин.директора
KPP	КПП
OKDP	ОКДП
OKFS	ОКФС
OKOPF	ОКОПФ
StartDate	Дата начала работы

### clsPayDocTax – метод инициализации внешней отчетности

Свойство	Описание
USN	учитывать признак УСН в документах (если USN=0, выбираются все документы без учета признака УСН; USN=1 – выбираются документы, у которых установлен признак УСН; USN=1 – выбираются документы, у которых не установлен признак УСН)
CONT	учитывать статус документа (CONT=1 – выбираются только документы в статусе контрирован; CONT=2 – выбираются документы в статусе контрирован и проведен)

### clsAddressStruc – структурированный адрес

Свойство	Описание
Init	создание временной таблицы, строки которой содержат строки адресов предприятия (из настройки предприятия)
MoveFirst	перемещает указатель записи в таблице на первую строку
MoveLast	перемещает указатель записи в таблице на последнюю строку
MoveNext	перемещает указатель записи в таблице на следующую строку
MovePrevious	перемещает указатель записи в таблице на предыдущую строку
BOF	вышли за первую строку
EOF	вышли за последнюю строку
RecordCount	количество строк в таблице
IsActive	признак активного адреса
IDAddressCtg	категория адреса (5 — юридический, 6 — фактический)
NameAddressCtg	наименование категории адреса
Address	строка сцепленного адреса
PostIndex	почтовый индекс
CodeRegion	код региона
NameRegion	наименование региона
Area	район
City	город
Item	населенный пункт
Street	улица
House	дом
Building	корпус
Apartment	квартира (офис)
CommentAddress	комментарий к адресу

### 11.3.2 Классы системы, используемые при написании формулы расчета прейскуранта.

#### PriceList – прейскурант

Свойство	Описание
RateBasePrice	Курс валюты базового прейскуранта
RatePrice	Курс валюты текущего прейскуранта
DateCalcul	Дата_расчета
Discount	Скидка

#### Stock – артикул

Свойство	Описание
BasePriceStock	Цена базового прейскуранта
LastPriceBuy	Цена последней закупки
CodeStock	Код
FullCodeStock	Артикул
NameStock	Наименование
NameStockAlt	Альтернативное наименование
QtyStockCurrent	Остаток
PriceAccount	Цена средняя
CostAccount	Стоимость
TradingMargin	Торговая надбавка
PricePrice	Цена прейскуранта
PointQuote	Точка заказа
StatusStock	Статус
CodeUnit	Основная единица измерения
IsPrice	Входит в прейскурант
StockCtg	Категория
CodeStockGroup	Код группы
IsDateCondition	Учитывать срок годности
TaxTariff("Код налога")	Ставка налога
minPrice(DatePrice, nDay)	Минимальная цена за период
maxPrice(DatePrice, nDay)	Максимальная цена за период
avgPrice(DatePrice, nDay)	Средняя цена за период
minPriceCurrency(DatePrice, nDay)	Минимальная цена за период в валюте прейскуранта
maxPriceCurrency(DatePrice, nDay)	Максимальная цена за период в валюте прейскуранта
avgPriceCurrency(DatePrice, nDay)	Средняя цена за период в валюте прейскуранта
LastPriceCurrencyBuy	Цена последней закупки в валюте прейскуранта

### 11.3.3 Классы, используемые при написании алгоритмов типовых проводок

#### Posting – проводка

Свойство	Описание
IDAccountType	Идентификатор системы учета
DatePosting	Дата
IsOutBalance	Баланс/Забаланс
IsRest	Остатки
DT_IDAccount	Счет ДБ
KT_IDAccount	Счет КТ
SummaPosting	Сумма проводки
CommentPosting	Примечание
DT_IDCenter1	Аналитика1
DT_IDCenter2	Аналитика2
DT_IDCenter3	Аналитика3
DT_IDCenter4	Аналитика4
KT_IDCenter1	Аналитика1
KT_IDCenter2	Аналитика2
KT_IDCenter3	Аналитика3
KT_IDCenter4	Аналитика4
Save	Записать
Clear	Очистить
RatePosting	Курс валюты проводки
IsDivide	Признак деления на курс
SummaAccount	Сумма в валюте учета
IDCurrency	Идентификатор валюты проводки

#### AccountDoc – бухгалтерская справка

Свойство	Описание
Partner	Это объект – партнер, позволяющий добраться до любого свойства этого объекта
Dogovor	Это объект – договор, позволяющий добраться до любого свойства этого объекта
IdIncomeCosts	ID вида доходов/расходов
Expenses	Статья затрат
ExpensesCode	Статья затрат. Код
Account	Корр. счет
AccountCode	Код корр. счета
CurrencyDoc	Валюта
SummaDoc	Сумма в валюте документа
OperTypeCode	Вид операции. Код
DocType	Документ. Код
Comment	Назначение

Number	Номер
DateDoc	Дата
Fields("Наименование поля")	Дополнительное поле
RateAccount	Курс валюты документа к валюте учета

### Dogovor – договор (DogovorTransf – договор трансферта)

Свойство	Описание
DogovorType	Тип (1-приход, 2-расход)
IdDogovor	ID договора (для второй аналитики)
DogovorJournal	Журнал
DogovorDoc	Вид документа
DogovorSumma	Сумма
DogovorCurrency	Валюта
DogovorAccount	Корр. счет (ID)
DogovorPartner	Партнер
DogovorManager	Менеджер
DogovorAccountCode	Код. Корр. счета
DogovorRate	Курс валюты преискуранта к валюте сделки

### Partner – партнер (PartnerTransf – партнер трансферта)

Свойство	Описание
Group	Группа
IdPartner	ID
Code	Код
Ctg	Категория
INN	ИНН

### Payment – платежный документ

Свойство	Описание
<b>Dogovor</b>	Это объект – договор, позволяющий добраться до любого свойства этого объекта
<b>Partner</b>	Это объект – партнер, позволяющий добраться до любого свойства этого объекта
<b>TaxDoc</b>	Налоги документа
Fields("Наименование поля")	Дополнительное поле
IsDivideAccount	Признак деления на курс для суммы в валюте учета
IsDivideBagrain	Признак деления на курс для суммы в валюте расчетов
MCenterAccountCode	Касса/Банк. Код счета учета
IDExpenses	Статья затрат
IDExpenses	Статья затрат

IDAccount	Корп. счет
AccountCode	Корп. счет. Код
IDCurrency	Валюта
SummaDoc	Сумма в валюте документа
OperTypeCode	Вид операции. Код
OperTypeIsArrival	Вид операции. Тип
IdIncomeCosts	ID вида дохода/расхода
DocTypeCode	Документ. Код
Number	Номер
DateDoc	Дата
DateOper	Дата проведения
IDMCenter	Касса/Банк
MCenterCode	Касса/Банк. Код
MCenterIDAccount	Касса/Банк. Счет учета
Comment	Назначение
PaymentCodeCode	Шифр платежа. Код
RateBargain	Кросс-курс валюты расчетов к валюте документа
RateAccount	Курс валюты документа к валюте учета
SummaBargain	Сумма в валюте расчетов
SummaAccount	Сумма в валюте учета
TaxModelCode	Налоговая модель. Код
SummTax("код налога")	Налог. Сумма
SummTaxTariff("код налога", "ставка")	Налог. Сумма ставки
<b>Payment.TaxPaymentDoc</b>	Налоговый платеж

### **Payment.TaxPaymentDoc – налоговый платеж**

<b>Свойство</b>	<b>Описание</b>
IdTaxPay	Id уплачиваемого налога (чтение)
CodeTaxPay	код уплачиваемого налога (чтение)
IdPeriodPay	Id периодичности уплаты налога (чтение)
CodePeriodPay	код периодичности уплаты налога (чтение)
IdPeriodTax	Id налогового периода (чтение)
MonthPeriodTax	номер месяца налогового периода (от 01 до 12) (чтение)
QuartPeriodTax	номер квартала налогового периода (от 01 до 04) (чтение)
HalfYearPeriodTax	номер полугодия налогового периода ( 01 или 02) (чтение)
YearPeriodTax	год налогового периода (чтение)
KBK	код бюджетной классификации (чтение)
POP	показатель основания платежа (чтение)
PTP	показатель типа платежа (чтение)
PNP	показатель налогового периода (чтение/запись)
PND	показатель номера документа (чтение/запись)
PDD	показатель даты документа (чтение/запись)



## DocStock - накладная

Свойство	Описание
<b>DocStockDoc</b>	Объект - строка накладной, позволяющий добраться до любого свойства этого объекта
<b>Dogovor</b>	Это объект – договор, позволяющий добраться до любого свойства этого объекта
<b>Partner</b>	Это объект – партнер, позволяющий добраться до любого свойства этого объекта
<b>TaxDoc</b>	Налоги накладной
AccountCode	Счет расчетов. Код
Comment	Назначение
DateDoc	Дата
DocTypeCode	Документ. Код
ExpensesCode	Статья затрат. Код
Fields("наименование поля")	Дополнительное поле
IDAccount	Счет расчетов
IdCurrency	Валюта
IDExpenses	Статья затрат
IdSclad	Склад ID
IdIncomeCosts	ID вида доходов/расходов
IsDivideAccount	Признак деления на курс в сумме для суммы в валюте учета
IsDivideBagrain	Признак деления на курс в сумме для суммы в валюте расчетов
Number	Номер
OperTypelsArrival	Вид операции. Тип
RateAccount	Курс валюты документа к валюте учета
RateBagrain	Курс валюты расчетов к валюте учета
SummaAccount	Сумма в валюте учета
SummaDoc	Сумма в валюте документа
SummaTotal	Сумма к оплате
TaxModelCode	Налоговая модель. Код
SummaTax("код налога")	Сумма налога
SummaWhouse	Сумма списания со склада

## DocStockDoc – строка накладной

Свойство	Описание
<b>TaxStr</b>	Налоги артикула
CodeAccount	Счет Код
CodeAccountTransf	Счет внутреннего перемещения Код
CodeStock	Код товара
CodeStockGroup	Код группы товара
CostAccount	Сумма в валюте учета
CostDoc	Стоимость в валюте документа
CostTotal	Сумма к оплате
CostTotalDoc	Стоимость в валюте документа со всеми налогами

CostWhouse	Сумма складская
CostWithoutTax	Сумма без налогов
FullCodeStock	Товар Артикул
IDAccount	Счет ID
IDAccountTransf	Счет внутреннего перемещения ID
IDOwner	Собственник ID
IDOwnerDog	Договор консигнации ID
IDStockGroup	Идентификатор группы товара
IDResource	Вид ресурса
NameStockGroupe	Товар Наименование группы
NumStock	Количество строк накладной
PriceBalance	Цена бухгалтерская (себестоимость)
PriceDoc	Цена в валюте документа
PriceWhouse	Цена складская
QntyUnitBase	Количество в основных единицах измерения
StockGroupCtg	Категория номенклатуры
StockName	Товар Наименование
SummTax("Код налога")	Налог Сумма
TaxRate("Код налога")	Ставка налога
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись
SharingCostStr	Распределенные затраты

### TaxStr – налоги артикула

Свойство	Описание
IDTax	ID налога
TaxCode	Код налога
IDRate	ID ставки налога
TaxRate	Ставка налога
TaxSum	Сумма налога
EOF	Вышли за последнюю запись
BOF	Вышли за первую запись
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись
Find("код налога", ставка)	Найти запись по условию

**DocObligation – справка о зачете**

Свойство	Описание
<b>Dogovor</b>	Это объект – договор, позволяющий добраться до любого свойства этого объекта
<b>DogovorTransf</b>	Это объект – Договор трансф, позволяющий добраться до любого свойства этого объекта
<b>Partner</b>	Это объект – партнер, позволяющий добраться до любого свойства этого объекта
<b>PartnerTransf</b>	Это объект – Партнер трансф, позволяющий добраться до любого свойства этого объекта.
<b>TaxDoc</b>	Налоги справки о зачете
IDExpenses	Статья затрат
ExpensesCode	Статья затрат. Код
IDAccount	Счет расчетов
AccountCode	Счет расчетов. Код
SummaDoc	Сумма в валюте документа
OperTypeCode	Вид операции. Код
OperTypelsArrival	Вид операции. Тип
DocTypeCode	Документ. Код
IdIncomeCosts	ID вида дохода/расхода
Number	Номер
DateDoc	Дата
Comment	Назначение
RateBargain	Курс валюты расчетов
SummaBargain	Сумма в валюте расчетов
SummaTransf	Сумма трансф
TaxModelCode	Налоговая модель. Код
SummTax("код налога")	Налог. Сумма
OperTypeCodeTransf	Вид операции трансф. Код
OperTypelsArrivalTransf	Вид операции трансф. Тип
RateBargainTransf	Курс валюты расчетов трансф
SummaBargainTransf	Сумма в валюте расчетов трансф
IDCurrency	Валюта
Fields{"Наименование поля"}	Дополнительное поле

**TaxDoc – налоги справки о зачете (акта выполненных работ, акта приемки-передачи ОС)**

Свойство	Описание
IDTax	ID налога
TaxCode	Код налога
IDRate	ID ставки налога
TaxRate	Ставка налога
TaxSum	Сумма налога
BOF	Вышли за первую запись
EOF	Вышли за последнюю запись

MoveFirst	Стать на первую запись
MovePrevious	Стать на предыдущую запись
MoveNext	Стать на следующую запись
MoveLast	Стать на последнюю запись
Find("код налога", ставка)	Найти запись по условию

### Bookkeeping – бухгалтерия

Свойство	Описание
ObjCurrency	Валюта
Analytic	Аналитика в проводках
Balance	Баланс
SubAccount	Субсчета
IdAccountType("код")	Возврат ID Плана счетов по его коду
Account("", IdAccountType)	Возврат ID счета по его коду для заданного Плана счетов
Code([ID])	Возврат кода счета по его ID
Analyst1("категория", "код")	Возврат ID аналитики первого уровня по коду
Analyst3("")	Возврат ID артикула по его коду
Analyst4("")	Возврат ID статьи затрат
Status("", IdAccountType)	Статус счета
Balance("", IdAccountType)	Отношение к балансу
Analyst("", IdAccountType)	Наличие аналитики
Center1("", IdAccountType)	Наличие аналитики 1-го уровня
Center2("", IdAccountType)	Наличие аналитики 2-го уровня
Center3("", IdAccountType)	Наличие аналитики 4-го уровня
Center4("", IdAccountType)	Наличие аналитики 4-го уровня
Storage("", IdAccountType)	Накопительный
Center1Def("", IdAccountType)	Возврат ID аналитики 1-го уровня по умолчанию
Center2Def("", IdAccountType)	Возврат ID аналитики 2-го уровня по умолчанию
Center3Def("", IdAccountType)	Возврат ID аналитики 3-го уровня по умолчанию
Center4Def("", IdAccountType)	Возврат ID аналитики 4-го уровня по умолчанию
IsCurrency("", IdAccountType)	Признак валютного счета
TaxAcc(IDTax, IDResource, IDTaxRate, Flag123)	Возврат ID счета (расчетов, учета или затрат)
TaxAcc2( sCodeTax , sTypeResource, Rate, Flag123)	Возврат ID счета (расчетов, учета или затрат)
DBKT( IdAccountType , "50","78",[dBeg],[dEnd])	Оборот по двум счетам (дебет и кредит)

### ObjCurrency – валюта

Свойство	Описание
IDCurrency	Идентификатор валюты

CodeCurrency	Код валюты
RecordCount	Количество валют
BOF	Вышли за первую запись
EOF	Вышли за последнюю запись
MoveFirst	Стать на первую запись
MovePrevious	Стать на предыдущую запись
MoveNext	Стать на следующую запись
MoveLast	Стать на последнюю запись
RateAccount("Код валюты", Дата)	Курс валюты

### Analytic – аналитика в проводках

Свойство	Описание
NumAnalytic	Количество аналитических карточек
Init(IDAccount, BegDate, EndDate, IdCurrency, IsNotValToCurrency)	Инициализация аналитики (формат даты #mm/dd/yyyy#)
FindCard(id1, id2, id3, id4)	Поиск карточки
IdPartner	Идентификатор партнера
Partner	Наименование партнера по первой аналитике
IdBagrain	IdBagrain
Bagrain	Код документа + номер + дата сделки
IdStock	IdStock
NameStock	Наименование третьей аналитики
IdExpenses	IdExpenses
Expences	Наименование счета затрат
SaldoBegDB	Сальдо на начало (дебет)
SaldoBegKT	Сальдо на начало (кредит)
OborotDB	Обороты за период (дебет)
OborotKT	Обороты за период (кредит)
SaldoEndDB	Сальдо на конец (дебет)
SaldoEndKT	Сальдо на конец (кредит)
MoveFirst	Стать на первую запись
MovePrevious	Стать на предыдущую запись
MoveNext	Стать на следующую запись
MoveLast	Стать на последнюю запись
EOF	Вышли за последнюю запись

### Balance – баланс

Свойство	Описание
Init(DateBeg, DateEnd, IdCurrency, IsNotValToCurrency, aidAccountType)	Расчет оборотов (формат даты #mm/dd/yyyy#)

SaldoDB(sCode)	Дебетовое сальдо счета по его коду
SaldoKT(sCode)	Кредитовое сальдо счета по его коду
SaldoBegDB(sCode)	Начальное дебетовое сальдо счета по его коду
SaldoBegKT(sCode)	начальное кредитовое сальдо счета по его коду
OborotDB(sCode)	Дебетовый оборот
OborotKT(sCode)	Кредитовый оборот

### SubAccount – субсчета

Свойство	Описание
InitSubAccount(aCode, aldAccountType)	Инициализация
CodeAccount	Возвращает субсчет
MoveFirst	Стать на первую запись
MovePrevious	Стать на предыдущую запись
MoveNext	Стать на следующую запись
MoveLast	Стать на последнюю запись
EOF	Вышли за последнюю запись

### DocAssets – акт приемки-передачи ОС

Свойство	Описание
OperTypeCode	Код вида операции
IsArrival	Тип движения
IsTransfert	Признак внутреннего перемещения
DocTypeCode	Вид документа
Number	Номер
DateDoc	Дата
Comment	Комментарий
SummaAccount	Сумма документа
SummaTotal	Сумма к оплате
Staff	МОЛ (ID)
Division	Подразделение (ID)
StaffTransf	МОЛ по трансферту(ID)
Expenses	Статья затрат (ID)
ExpensesCode	Код статьи затрат
Account	Счет расчетов (ID)
AccountCode	Код счета расчетов
IdIncomeCosts	ID вида доходов/расходов
TaxModelCode	Код налоговой модели
SummaTax("Код налога")	Общая сумма налога по документу
Fields("Наименование поля")	Дополнительное поле

CurrencyBagrain	Валюта акта
RateBagrain	Курс валюты акта
<b>Dogovor</b>	Договор (объект)
<b>Partner</b>	Партнер (объект)
<b>TaxDoc</b>	Налоги акта приемки-передачи ОС
<b>DocAssetsStr</b>	Строка акта приемки-передачи ОС

### DocAssetsStr – строка акта приемки-передачи ОС

Свойство	Описание
AccNew	Новый счет учета ОС (ID) (для перемещения ОС)
AccCodeNew	Новый код счета учета ОС (для перемещения ОС)
AccAmortNew	Новый счет начисления амортизации (ID) (для перемещения ОС)
AccCodeAmortNew	Новый код счета начисления амортизации (ID) (для перемещения ОС)
AccChargesNew	Новый счет списания затрат при начислении амортизации (ID) (для перемещения ОС)
AccCodeChargesNew	Новый код счета списания затрат при начислении амортизации (ID) (для перемещения ОС)
AccOld	Старый счет учета ОС (ID)
AccCodeOld	Код старого счета учета ОС (ID)
AccAmortOld	Старый счет начисления амортизации (ID)
AccCodeAmortOld	Код старого счета начисления амортизации (ID)
AccChargesOLd	Старый счет списания затрат при начислении амортизации (ID)
AccCodeChargesOLd	Код старого счета списания затрат при начислении амортизации (ID)
Stock	Артикул (ID)
StockName	Наименование артикула
QntyUnitBase	Количество в основных единицах измерения
CostAccount	Стоимость
CostWithoutTax	Сумма без налогов
CostTotal	Сумма к оплате
CostWhouse	Начальная (балансовая) стоимость ОС
PriceWhouse	Цена ОС
AmortInIt	Начальный износ
AmortComulate	Накопленный износ
QntyUnitBaseNew	Количество после движения
CostWhouseNew	Начальная стоимость после движения
AmortInItNew	Начальный износ после движения
AmortComulateNew	Накопленный износ после движения
AmortMountOut	Списываемый износ за месяц
SummaTax("Код налога")	Сумма налога
TaxRate	Ставка налога
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись

MovePrevious	Встать на предыдущую запись
NumStr	Количество строк документа
CostBagrain	Сумма в валюте документа
<b>Assets</b>	Карточка ОС

### AdvPaid – справка по подотчету

Свойство	Описание
Number	Номер
DateDoc	Дата справки по подотчету
Comment	Содержание документа
SummaDoc	Сумма в валюте документа
RateBagrain	Курс в валюте расчетов
SummaBagrain	Сумма в валюте расчетов
RateAccount	Курс валюты документа к валюте учета
SummaAccount	Сумма в валюте учета
IdIncomeCosts	ID вида доходов/расходов
IdCurrency	Ссылка на валюту документа
CurrencyCode	Код валюты документа
IdExpenses	Ссылка на статью затрат
ExpensesCode	Код статьи затрат
IdAccount	Ссылка на счет расчетов
AccountCode	Код счета расчетов
Fields("наименование поля")	Значение дополнительного поля
<b>AdvPaidTransfert</b>	Справка по подотчету (трансферт)
<b>AdvReport</b>	Авансовый отчет

### AdvPaidTransfert – справка по подотчету (трансферт)

Свойство	Описание
<b>AdvReport</b>	Авансовый отчет
Fields("наименование поля")	Значение дополнительного поля
AccountCode	Код счета расчетов
IdAccount	Ссылка на счет расчетов
ExpensesCode	Код статьи затрат
IdExpenses	Ссылка на статью затрат
CurrencyCode	Код валюты документа
IdCurrency	Ссылка на валюту документа
SummaAccount	Сумма в валюте учета
RateAccount	Курс валюты документа к валюте учета
SummaBagrain	Сума в валюте расчетов
RateBagrain	Курс к валюте расчетов



SummaDoc	Сумма в валюте документа
Comment	Содержание документа
DateDoc	Дата справки по подотчету
Number	Номер
IdIncomeCosts	Id вида доходов/расходов

### AdvReport – авансовый отчет

Свойство	Описание
IDAdvReport	Идентификатор авансового отчета
CodeDocType	Код вида документа
Number	Номер
DateDoc	Дата авансового отчета
IdStaff	Ссылка на сотрудника
IdDivision	Ссылка на подразделение
Comment	Наименование документа
SummaDoc	Сумма в валюте документа
Rate	Курс на дату отчета
Saldo	Остаток
IdCurrency	Ссылка на валюту документа
CurrencyCode	Код валюты документа
IdExpenses	Ссылка на статью затрат
ExpensesCode	Код статьи затрат
IdAccount	Ссылка на счет расчетов
AccountCode	Код счета расчетов
SummaTax("код")	Сумма по налогу
Fields("наименование поля")	Значение дополнительного поля
IdIncomeCosts	Id вида доходов/расходов

### ActWork – акт выполненных работ

Свойство	Описание
ActWorkStr	Строки акта
Dogovor	Договор
Partner	Партнер
TaxDoc	Налоги акта выполненных работ
AccountCode	Код счета расчетов
Comment	Назначение
DateDoc	Дата
DocTypeCode	Код документа
ExpensesCode	Код статьи затрат
Fields("Наименование поля")	Дополнительное поле

IDAccount	Счет расчетов
IdCurrency	Валюта
IDExpenses	Статья затрат
IsDivideAccount	Признак деления на курс для суммы в валюте учета
IsDivideBagrain	Признак деления на курс для суммы в валюте расчетов
Number	Номер
OperTypeCode	Код вида операции
OperTypeIsArrival	Тип вида операции
IdIncomeCosts	ID вида доходов/расходов
RateAccount	Курс валюты документа к валюте учета
RateBagrain	Курс валюты расчетов к валюте учета
SummaAccount	Сумма в валюте учета
SummaDoc	Сумма в валюте документа
SummaTotal	Сумма к оплате
TaxModelCode	Код налоговой модели
SummTax("код налога")	Сумма налога

### ActWorkStr – строка акта выполненных работ

Свойство	Описание
TaxStr	Налоги по строке акта выполненных работ
CodeWork	Код вида работ
CostAccount	Сумма в валюте учета
CostDoc	Стоимость в валюте документа
CostTotalDoc	Стоимость в валюте документа со всеми налогами
CostTotal	Сумма к оплате
CostWithoutTax	Сумма без налогов
FullCodeWork	Вид работ Артикул
IDWork	Вид работ Идентификатор
IDResource	Вид ресурса
NumWork	Количество строк акта
PriceDoc	Цена в валюте документа
QntyUnitBase	Количество в основных единицах измерения
StockGroupCtg	Категория номенклатуры
WorkName	Вид работ Наименование
SummTax("код налога")	Сумма налога
TaxRate("код налога")	Ставка налога
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись

### 11.3.4 Классы системы, используемые при написании алгоритмов расчета в заработной плате

#### Salary – зарплата

Свойство	Описание	Пример
SumIn("")	Сумма начислений для В/О	SumIn("2000.01")
SumOut("")	Сумма удержаний для В/О	SumOut("2000.01")
AddPeriod("")	Изменить период	AddPeriod("2000.01",-2)
CurPeriod	Текущий период	
SumInOutCode("", "", "", "")	Сумма по видам оплат за период	
PercentInOutCode("", "", "", "")	Процент по видам оплат за период	
BaseSumInOutCode("", "", "", "")	Базовая сумма по видам оплат за период	
SumInOutCol("", "", "", "")	Сумма по колонке ВО за период	
PercentInOutCol("", "", "", "")	Процент по колонке ВО за период	
BaseSumInOutCol("", "", "", "")	Базовая сумма по колонке ВО за период	
MinSalary("")	Минимальная зарплата за период	
Coefficient("", "")	Сумма коэффициента за период	
SumInOutCodeAll	Сумма по колонке вида оплаты	
Salary.SocialStaff _Woman("", "", "", "")	Этот метод будет рассчитывать количество женщин, годовые доходы которых попали в указанную вилку доходов по указанному фонду	Salary.SocialStaff_Woman ("2001.01", "ФСС", 100000.01, 300000)
Salary.SocialStaff _ENVD("период")	рассчитывать общее количество сотрудников, у которых есть доходы с видом налогообложения ЕНВД. Обрабатываются данные с января по месяц, указанный в параметре <период>.	Salary.SocialStaff _ENVD("2001.01")
Salary.SocialStaff _Woman_ENVD("период")	количество женщин, у которых есть доходы с видом налогообложения ЕНВД. Обрабатываются данные с января по месяц, указанный в параметре "период".	Salary.SocialStaff_Woman_ENVD ("2001.01")
Salary.RegressScale.NumberAverageInv("период")	выводит среднесписочную численность инвалидов с начала года по указанный период. Инвалидность сотрудника определяется по наличию в Личной карточке/Документы строки с признаком «И» (инвалидность)	
Salary.FondFSSt. Init "период", "код фонда"	формирует данные для выполнения описанных ниже функций	
Salary.FondFSSt.Summa("Период начала", "Период окончания")	выводит сумму налогооблагаемой базы для фонда, указанного в методе Init, для всех сотрудников с периода начала по период окончания	
Salary.FondFSSt.SummaInv("Период начала", "Период окончания")	выводит сумму налогооблагаемой базы для фонда, указанного в методе Init, для сотрудников инвалидов с периода начала по период окончания	

**Staff – сотрудник**

Свойство	Описание
Salary	Оклад
Tariff	Ставка налога
Advance	Аванс
IsAdvProc	Аванс процентом
PrHospital	Процент больничного
CodeDivision	Код подразделения
CodeTitul	Код должности
CodeWorkGraphic	Код графика
Tnum	Табельный номер
Birthday	Дата рождения
DateReceipt	Дата приема на работу
DateTrial	Дата окончания испытательного срока
DateDismissal	Дата увольнения
Dependents	Количество иждивенцев
IDAccount	Счет лицевого счета
IDExpenses	Статья затрат
KindStaff	Вид найма
IDStaff	ID сотрудника
IDDivision	ID подразделения
DayPlan(****)	Количество плановых дней
HourPlan(****)	Количество плановых часов
DatWithSatPlan(****)	Количество плановых дней, включая субботы
DayFact(****)	Количество фактических дней
HourFact(****)	Количество фактических часов
DatWithSatFact(****)	Количество фактических дней, включая субботы
IDAccountTax	Счет налогового учета
Coefficient1, Coefficient2, Coefficient3	Дополнительные коэффициенты

**SlipCalc – расчетный лист**

Свойство	Описание
Period	Период
DayFact	Дней отработано
HourFact	Часов отработано
DayWithSatFact	Дней с субботаами отработано
DayPlan	Дней по графику
HourPlan	Часов по графику
DayWithSatPlan	Дней с субботаами по графику
SaldIn	Сальдо на начало

TotalIn	Итого начислено
TotalOut	Итого удержано
TotalPay	Итого выдано
SaldoOut	Сальдо на конец
IdStaff	Сотрудник

**InOut – вид оплаты**

Свойство	Описание
CodeInOut	Код
NameInOut	Наименование
IsConst	Признак постоянного ВО
IsReal	Признак входимости в итог РЛ
InOutCtg	Тип (1-нач.,2-удерж.,3-выплата)
NumSpecialAlg	Номер спецалгоритма
CodeIncome	Вид дохода (вычета)
Priority	Приоритет
Account	ID Счет учета
AccountCode	Счет учета
Correction	Возможность сторно
IDExpenses	Статья затрат
IDAccountTax	Счет налогового учета

**SheetPay – платежная ведомость**

Свойство	Описание
Period	Период создания
DocType	Вид документа
Number	Номер
Date	Дата создания
CodeDivision	Код подразделения
PeriodClose	Период закрытия

**SheetCalc – расчетная ведомость**

Свойство	Описание
Year	Год
Period	Период
MoveFirst	Встать на первый Р/Л
MoveNext	Встать на следующий Р/Л
DateEndMonth	Последний день месяца

**SlipCalcStr – строка расчетной ведомости**

Свойство	Описание
Day	Дней
Hour	Часов
SummaBase	Базовая сумма
Summa	Сумма
Percent	Процент
Period	Период
IsManualCorr	Ручная правка
AccountCode	Счет учета
IDAccountTax	Счет налогового учета

**CalcTax – справка о начислении налога**

Свойство	Описание
IDPeriod	Идентификатор периода
Period	Период
DocType	Вид документа
IdIncomeCosts	ID вида доходов/расходов
Number	Номер
DateDoc	Дата
IdCurrency	Валюта
IDExpenses	Статья затрат
Summa	Начислено
<b>Tax</b>	Налог
Fields("Наименование поля")	Дополнительное поле
<b>SumAccountFundSoc</b>	Фонд социального налога (счет из расчетного листа)
<b>SumAccountFund</b>	Фонд (счет из расчетного листа)
<b>SumAccountFundStaffSoc</b>	Фонд социального налога (счет из лицевого счета)
<b>SumAccountFundStaff</b>	Фонд (счет из лицевого счета)
<b>SumAccountFundPF</b>	Фонд пенсионного страхования (счет из расчетного листа)
<b>SumAccountFundStaffPF</b>	Фонд пенсионного страхования (счет из лицевого счета)
SumTaxPF	Сумма страховых взносов на обязательное пенсионное страхование, рассчитанных с заработной платы, полученной при выполнении деятельности, облагаемой ЕНВД

**Tax – налог**

Свойство	Описание
IDTax	Идентификатор налога
Code	Код
Name	Наименование

IsSalary	Признак зарплатного
IsStock	Признак номенклатурного
SaldoKT	Долг бюджету
IsSystem	Системный
ActiveProc	Активный процент
IDTaxRate	ID активного процента
GroupTax	ID группы налога
NameGroup	Наименование группы налога
IsSocial	Признак социального налога
IsFixed	Признак фиксированного налога
IsOpen	Признак используемого налога
IdPartner	ID партнера
IsPF	Признак Пенсионного фонда
SumTaxAbleFund ("период")	Рассчитывает налогооблагаемую базу для фондов ЕСН
SumTaxAbleFundPF ("период")	Рассчитывает налогооблагаемую базу для пенсионного фонда
SumTaxAbleFundPod ("период")	Рассчитывает налогооблагаемую базу для подоходного налога
SumFundPod ("период")	Сумма подоходного налога

### **SumAccountFundSoc, SumAccountFundStaffSoc, SumAccountFundPF, SumAccountFundStaffPF – фонды**

<b>Свойство</b>	<b>Описание</b>
Init(<ID налога>, <ID периода>, <Категория центра учета >)	Инициализация
Account	Идентификатор счета
Summa	Сумма
Staff	Идентификатор сотрудника
Division	Идентификатор подразделения
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись
CodeAccount	Код счета учета
IDExpenses	ID статьи затрат

### **SumAccountFund, SumAccountFundStaff – фонды**

<b>Свойство</b>	<b>Описание</b>
Init(<ID налога>, <ID периода>, <Категория центра учета >)	Инициализация
Account	Идентификатор счета
Summa	Сумма
Staff	Идентификатор сотрудника

Division	Идентификатор подразделения
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись
CodeAccount	Код счета учета
IsDisabled	Признак инвалидности
IDExpenses	ID статьи затрат

### 11.3.5 Классы системы, используемые при написании алгоритмов расчета амортизации в подсистеме Имущество

#### Assets – карточка ОС

Свойство	Описание
Stock	Артикул (ID)
CodeStock	Код артикула
FullCodeStock	Полный код артикула
NameStock	Наименование артикула
StockGroup	Группа артикула (ID)
StockGroupCode	Код группы артикула
TypeOc	Тип имущества
FunctionType	Назначение
UseType	Использование
ThingsType	Принадлежность
OKOF	ОКОФ
NumCard	Номер карточки ОС
NumInvent	Инвентарный номер
CostInit	Начальная (балансовая) стоимость
CostInitNS	Начальная (балансовая) стоимость для списания
CostInitN	Начальная (балансовая) стоимость (Н)
OstCostN	Остаточная стоимость (Н)
Qty	Количество по карточке
AmortInit	Начальный износ
AmortInitN	Начальный износ (Н)
AmortInitNS	Начальный износ для списания
AmortComulate	Накопленный износ на начало месяца
AmortComulateN	Накопленный износ на начало месяца (Н)
AmortComulateNS	Накопленный износ на начало месяца для списания
Amort	Износ за месяц
AmortN	Износ за месяц (Н)
AmortRate	Норма амортизации за год
AmortRateN	Норма амортизации за год (Н)



Life	Срок полезного использования
LifeRest	Срок полезного использования (оставшийся)
KoeffAmort	Коэффициент ускоренной амортизации
IsAmort50	Признак расчета амортизации по норме 50%
DateStart50	дата начала начисления 50% амортизации
DateEnd50	дата окончания начисления 50% амортизации
OutPutAll	Предполагаемая выработка (пробег)
OutPutInit	Начальная выработка (пробег)
IsAmort	Признак расчета амортизации
StatusAssets	Статус карточки ОС
Division	Подразделение (ID)
Staff	МОЛ (ID)
Account	Счет учета ОС (ID)
AccountN	Счет доходов ID (H)
CodeAccount	Код счета учета ОС
CodeAccountN	Код счета доходов (H)
AccAmort	Счет начисления амортизации (ID)
AccAmortN	Счет начисления амортизации ID (H)
CodeAccAmort	Код счета начисления амортизации
CodeAccAmortN	Код счета начисления амортизации (H)
AccCharges	Счет списания затрат при начислении амортизации (ID)
AccChargesN	Счет первоначальной стоимости ОС ID (H)
CodeAccCharges	Код счета списания затрат при начислении амортизации
CodeAccChargesN	Код счета первоначальной стоимости ОС (H)
MethodAmort	Код метода расчета амортизации
IsOutPut	Признак метода расчета по выработке (пробегу)
NumGroupsAmort	Номер группы амортизации

### DocAmort – ведомость амортизации

Свойство	Описание
DocTypeCode	Вид документа
DateDoc	Дата
<b>Dogovor</b>	Это объект – договор, позволяющий добраться до любого свойства этого объекта
<b>Partner</b>	Это объект – партнер, позволяющий добраться до любого свойства этого объекта
SummaTotal	Сумма по документу
Fields("Наименование поля")	Дополнительное поле
<b>DocAmortStr</b>	Строка ведомости амортизации

**DocAmortStr – строка ведомости начисления амортизации**

Свойство	Описание
Amort	Сумма амортизации по строке
IsManual	Признак ручной правки
OutPut	Величина месячной выработки (пробега)
<b>Assets</b>	Карточка ОС
CostRestBeg	Базовая стоимость (на начало периода)
AccountType	Тип системы учета
AmortRate	Норма амортизации
PercWearing	Процент износа
MoveFirst	Встать на первую запись
MoveLast	Встать на последнюю запись
MoveNext	Встать на следующую запись
MovePrevious	Встать на предыдущую запись
NumStr	Количество строк документа
DatePay	Дата оплаты
DateDocEx	Дата ввода в эксплуатацию
IsUSN	1 – по УСН, 0 – не по УСН

**CalcAmort – расчет ведомости амортизации**

Свойство	Описание
<b>DocAmortStr</b>	Строка ведомости амортизации

**11.3.6 Классы системы, используемые при написании алгоритмов расчета налогов в Мастере расчета суммы****CalcTax - справка начисления налога**

Свойство	Описание
CalcTax.TaxableSumm	Налогооблагаемая сумма (заполняется при расчете налога)
CalcTax.Summa	Начислено (сумма рассчитанного налога заполняется при расчете налога)
CalcTax.GetRateByName("Наименование")	функция, возвращающая значения параметра из карточки налога по наименованию параметра
CalcTax.IDTaxPeriod	идентификатор налогового периода справки начисления налога
CalcTax.TaxPeriod	наименование налогового периода справки начисления налога
CalcTax.TaxPeriodDate	дата начала налогового периода справки (если периодичность налога "Ежеквартально", то это первое число последнего месяца квартала)
CalcTax.TaxPeriodDateEnd	дата окончания налогового периода справки (если периодичность налога "Ежеквартально", то это последнее число последнего месяца квартала)
CalcTax.SumTax (<Id Периода c>, <Id периода по>, <Код налога>)	функция возвращает общую сумму "Начислено" по всем справкам начисления указанного налога, отобранным за указанный интервал НАЛОГОВЫХ периодов

	справок
CalcTax.SumTaxPay1(<Id периода с>, <Id периода по>, <Код налога>, <УСН>)	функция возвращает сумму налога, уплаченного за указанный интервал периодов (т.е. сумму из платежных документов, дата проведения которых попадает в заданный интервал периодов; параметр <УСН> = {0 — все платежи, 1 — платежи с признаком "Сделать запись в КДР", 2 — платежи без признака "Сделать запись в КДР"})
CalcTax.SumTaxPay2(<Id периода с>, <Id периода по>, <Код налога>, <УСН>)	функция возвращает сумму налога, уплаченного в счет налогового периода (т.е. сумму из платежных документов, дата налогового периода которых попадает в заданный интервал периодов; параметр <УСН> = {0 — все платежи, 1 — платежи с признаком "Сделать запись в КДР", 2 — платежи без признака "Сделать запись в КДР"})
CalcTax.SumTaxPay3(<Id периода с>, <Id периода по>, <Id налогового периода с>, <Id налогового периода по>, <Код налога>, <УСН>)	функция возвращает сумму налога, уплаченного в указанный интервал дат за указанный интервал налоговых периодов (т.е. сумму из платежных документов, дата проведения которых попадает в заданный интервал периодов и дата налогового периода которых попадает в заданный интервал налоговых периодов; параметр <УСН> = {0 — все платежи, 1 — платежи с признаком "Сделать запись в КДР", 2 — платежи без признака "Сделать запись в КДР"})

### CalcTax.Tax - налог

Свойство	Описание
CalcTax.Tax.PeriodCalc	периодичность расчета налога (0 — не определена, 1 — ежемесячно, 2 — ежеквартально)

### GlobVars - предприятие

Свойство	Описание
GlobVars.TaxRegime	вид налогообложения (0 — общий, 1 — УСН, 2 — ЕНВД)
GlobVars.ObjectTax	объект налогообложения для УСН (0 — доход, 1 — Доход — Расход)
GlobVars.DateBegSimplyTax	дата начала действия УСН
GlobVars.DateEndSimplyTax	дата окончания действия УСН
GlobVars.DateToPeriod(<Дата>)	функция возвращает идентификатор периода для заданной даты

### BookInkome – книга доходов/расходов

Свойство	Описание
BookInkome.InitDateBetween(DateBeg, DateEnd)	функция инициализации, выбирает из Книги доходов/расходов (КДР) документы за указанный интервал дат и создает временную таблицу этих записей
BookInkome.MoveFirst	функция перемещения на первую запись временной таблицы
BookInkome.MoveLast	функция перемещения на последнюю запись временной таблицы
BookInkome.MoveNext	функция перемещения на следующую запись временной таблицы
BookInkome.MovePrevious	функция перемещения на предыдущую запись временной таблицы
BookInkome.EOF	признак выхода за последнюю запись временной таблицы
BookInkome.BOF	признак выхода за первую запись временной таблицы
BookInkome.NumStr	количество отображенных записей
BookInkome.SummaDoc	сумма документа в валюте документа
BookInkome.SummaTotalDoc	сумма к оплате (со всеми налогами) в валюте документа

BookInkome.RateAccount	курс валюты документа к валюте учета
BookInkome.SummaAccount	стоимость отпускная в валюте учета
BookInkome.SummaTotal	сумма к оплате (со всеми налогами) в валюте учета
BookInkome.DateDoc	дата документа
BookInkome.DateOper	дата перевода документа в статус «Проведен»
BookInkome.Status	статус документа (2 — проведен, 3 — контрирован)
BookInkome.Period	идентификатор периода документа
BookInkome.Arrival	признак приходной/расходной операции (1 — приход, 2 — расход)
BookInkome.SumInc(<Доход/Расход>)	функция возвращает общую сумму (SummaTotal) по всем отображенным во временную таблицу документам. Параметр <Доход/Расход> = {0 — Сумма, рассчитывается в соответствии с настройками предприятия ("Объект налогообложения" для УСН); 1 — Сумма дохода (суммируются только приходные документы); 2 — Сумма расхода (суммируются только расходные документы)}

### ТaxBook – книга покупок/продаж

Свойство	Описание
TaxBook.InitDateBetween(DateBeg,DateEnd)	функция инициализации, выбирает из Книги покупок/продаж записи за указанный интервал дат и создает временную таблицу этих записей
TaxBook.MoveFirst	функция перемещения на первую запись временной таблицы
TaxBook.MoveLast	функция перемещения на последнюю запись временной таблицы
TaxBook.MoveNext	функция перемещения на следующую запись временной таблицы
TaxBook.MovePrevious	функция перемещения на предыдущую запись временной таблицы
TaxBook.EOF	признак выхода за последнюю запись временной таблицы
TaxBook.BOF	признак выхода за первую запись временной таблицы
TaxBook.NumStr	количество отображенных записей
TaxBook.DateArrival	дата поставки
TaxBook.IsArrival	тип записи (1 — книга покупок, 2 — книга продаж)
TaxBook.DocTaxDate	дата счета — фактуры
TaxBook.DatePayment	дата платежа
TaxBook.DateLog	дата регистрации
TaxBook.DateWhouse	дата оприходования на склад
TaxBook.SummaTotal	сумма со всеми налогами
TaxBook.SummaNDS	сумма НДС
TaxBook.SummaCash	сумма НП
TaxBook.TypeBookTax	тип оплаты (1 — полная, 2 — частичная, 3 — сторнирующая запись)
TaxBook.SumBuy	сумма НДС по всем записям книги покупок за проинициализированный период
TaxBook.SumSell SumBuy	сумма НДС по всем записям книги продаж за проинициализированный период

**TaxBook.TaxBookSum – строки книги покупок/продаж** (отдельные строки в записи книги покупок/продаж соответствуют различным ставкам НДС из одного счета — фактуры)

Свойство	Описание
TaxBook.TaxBookSum.MoveFirst	функция перемещения на первую строку записи книги покупок/продаж
TaxBook.TaxBookSum.MoveLast	функция перемещения на последнюю строку записи книги покупок/продаж
TaxBook.TaxBookSum.MoveNext	функция перемещения на следующую строку записи книги покупок/продаж
TaxBook.TaxBookSum.MovePrevious	функция перемещения на предыдущую строку записи книги покупок/продаж
TaxBook.TaxBookSum.EOF	признак выхода за последнюю строку записи книги покупок/продаж
TaxBook.TaxBookSum.BOF	признак выхода за первую строку записи книги покупок/продаж
TaxBook.TaxBookSum.NumStr	количество строк в строк записи книги покупок/продаж
TaxBook.TaxBookSum.SummaTotal	сумма со всеми налогами
TaxBook.TaxBookSum.SummaTotal	сумма НДС
TaxBook.TaxBookSum.ProcTaxTariff	ставка НДС
TaxBook.TaxBookSum.SummaCash	сумма НП

## 11.4 Настройка видов оплат «Ссуда», «Проценты по ссуде», «Материальная выгода»

### 11.4.1 Ссуда

Для расчета суммы ссуды введены два вида оплаты:

Удержание СДО – начальный остаток по ссуде;

Удержание ССД – сумма возвращенной ссуды.

#### 11.4.1.1 СДО (остаток по ссуде)

При выдаче сотруднику ссуды в его расчетном листке должно быть указано удержание СДО (остаток по ссуде). Это удержание является постоянным и не увеличивает сумму удержаний работника (в карточке вида оплаты не стоит отметка «Увеличивает итоги р/л сотрудника»).

Сумму выданной ссуды необходимо ввести в колонку «Базовая сумма». В следующих месяцах остаток ссуды будет определяться автоматически как остаток ссуды на начало прошлого месяца за вычетом выплаченной в прошлом месяце части ссуды.

Когда сумма остатка ссуды станет равной нулю, удержание СДО нужно удалить из расчетного листка.

Алгоритм вида оплаты СДО выглядит следующим образом:

```

S1      =      Salary.SumInOutCode("СДО",Salary.AddPeriod(Salary.CurPeriod,-1),
      Salary.AddPeriod(Salary.CurPeriod,-1),"")
S2      =      S1-      Salary.SumInOutCode("ССД",Salary.AddPeriod(Salary.CurPeriod,-1),
      Salary.AddPeriod(Salary.CurPeriod,-1),"")
if S1>0 then
      SlipCalcStr.Summa = S2
else
' если ссуда выдана в текущем месяце, то тогда ее нужно ввести вручную в колонку "Базовая
сумма"
      SlipCalcStr.Summa = SlipCalcStr.SummaBase
end if

```

#### 11.4.1.2 ССД (сумма возвращенной ссуды)

Это удержание увеличивает итоги расчетного листка. Является ли оно постоянным или нет - нужно определить в зависимости от того, как сотрудники рассчитываются по ссуде (каждый месяц или нет).

#### 11.4.1.3 Ежемесячная выплата ссуды

Если ссуда выплачивается ежемесячно, то в карточке вида оплаты ССД необходимо поставить соответствующую отметку. В том месяце, в котором сотрудник начинает возвращать ссуду необходимо указать это удержание в его расчетном листке.

#### 11.4.1.4 Выплата ссуды постоянной суммой

В случае если ссуда выплачивается сотрудником постоянной суммой, сумма ежемесячной выплаты ссуды вводится в колонку "Базовая сумма" один раз в том месяце, в котором начинается выплата ссуды.

Алгоритм вида оплаты ССД в этом случае будет выглядеть следующим образом:

```
SS = SlipCalcStr.SummaBase
SumRL = Salary.SumIn("") - Salary.SumOut("")
CDO = Salary.SumInOutCode("СДО","","","")
' Если сумма начислений расчетного листка меньше выплачиваемой суммы ссуды, то
сумма ссуды уменьшается.
if SS > SumRL then
SS = SumRL
end if
' если сумма выплачиваемой ссуды меньше остатка ссуды, то берется остаток ссуды
if SS > CDO then
SlipCalcStr.Summa = CDO
else
SlipCalcStr.Summa = SS
end if
```

В данном алгоритме происходит проверка – если сумма выплаты больше, чем разница между суммой начислений и удержаний лицевого счета, то выплачивается не вся часть ссуды. Чтобы эта проверка работала, в карточке ССД необходимо отметить все виды начислений (кроме тех, которые не увеличивают итоги р/л) и все виды удержаний, которые должны выплачиваться раньше, чем рассчитывается ссуда.

Если такая проверка не нужна, то алгоритм можно сократить до строки:

```
SlipCalcStr.Summa = SlipCalcStr.SummaBase
```

#### 11.4.1.5 Выплата ссуды как процента от суммы начислений

Если сумма ссуды определяется как процент от начисленной сотруднику суммы, то в строке расчетного листка с видом оплаты ССД необходимо указать этот процент.

Алгоритм в этом случае будет выглядеть так:

```
SlipCalcStr.SummaBase = Salary.SumIn(Salary.CurPeriod)
SlipCalcStr.Summa = SlipCalcStr.SummaBase * SlipCalcStr.Percent / 100
```

В карточке вида оплаты ССД должны быть отмечены виды оплат, пропорционально которым начисляется ссуда.

#### 11.4.1.6 Выплата ссуды не каждый месяц

Если ссуда сотрудником выплачивается не каждый месяц, то вид оплаты ССД не должен быть постоянным. При выплате сотрудником части ссуды можно указывать это удержание в его расчетном листке или (особенно, если сотрудников несколько) создавать документ на выплату ссуды (в журнале Документов по видам оплат) с видом оплаты ССД. Алгоритм удержания описан выше.

### 11.4.2 Проценты по ссуде

Вид оплаты ПрС (постоянное удержание) вводится в расчетный листок сотрудника одновременно с видом оплаты СДО (остаток ссуды на начало месяца). В колонке «Процент» указывается процент по ссуде.

Процент по ссуде определяется как сумма остатка ссуды на начало месяца, умноженная на процент (в данном случае процент - годовой, если нужно указывать ежемесячный процент по ссуде, необходимо перенастроить алгоритм – не делить сумму на 12).

Предполагается, что ссуда была выдана в первый день месяца и погашается в последний день.

В этом случае алгоритм вида оплаты ПрС будет выглядеть следующим образом:

```
SlipCalcStr.SummaBase = Salary.SumOut("")
```

```
SlipCalcStr.Summa = SlipCalcStr.SummaBase * SlipCalcStr.Percent /12/100
```

В карточке вида оплаты нужно отметить удержание СДО.

### 11.4.3 Материальная выгода

Вид оплаты МВ. Начисление постоянное, не увеличивает итоги расчетного листка.

Для того чтобы материальная выгода рассчитывалась автоматически нужно, чтобы в расчетном листке сотрудника были удержания СДО - остаток ссуды на начало и ССД - выплата ссуды за месяц (если ссуда уже возвращается).

#### **ВНИМАНИЕ!**

---

В поле "Процент" строки расчетного листка для вида оплаты МВ нужно ввести годовой процент по ссуде. Процент ставки рефинансирования вводится непосредственно в этот алгоритм

---

Считаем, что ссуда выдается в первый день месяца, а погашение ссуды происходит в последний день месяца, поэтому определяем материальную выгоду, исходя из остатка ссуды на начало месяца.

Алгоритм вида оплаты МВ выглядит следующим образом:

```
St_Ref = 25 ' Процент ставки рефинансирования
```

```
if SlipCalcStr.Percent < St_Ref*3/4 then
```

```
SlipCalcStr.SummaBase = Salary.SumInOutCode("СДО", "", "", "")
```

```
SlipCalcStr.Summa = SlipCalcStr.SummaBase * (St_Ref*3/4 - SlipCalcStr.Percent)/12/100
```

```
end if
```

#### **ВНИМАНИЕ!**

---

Так как при расчете Расчетного листка сначала рассчитываются начисления, а потом удержания, т.е. начисление МВ (материальная выгода) рассчитывается раньше, чем удержание СДО (остаток ссуды на начало месяца), от которого МВ зависит, то на кнопку «Расчет» необходимо нажать дважды – сначала рассчитывается удержание СДО, а затем МВ.

---



## 11.5 Системные алгоритмы расчетов ЕНВД и УСН

### 11.5.1 Системный алгоритм расчета единого налога на вмененный доход

Системный алгоритм расчета Единого налога на вмененный доход настроен в соответствии требованиями Налогового кодекса РФ (глава 26.3. Система налогообложения в виде единого налога на вмененный доход для отдельных видов деятельности) со всеми изменениями и дополнениями.

```

Dim NMes, mDateBeg, PFR, NalogTmp, K1, SAll, SVd
if CalcTax.GetRateByName("ВД") > 0 then ' Для каждой карточки налога на ВД, т.е. по каждому
виду деятельности и по каждому объекту
K1      =      FormatNumber(      (1000+CalcTax.GetRateByName("Коф"))      /
(1000+CalcTax.GetRateByName("Ком")), 2)
CalcTax.TaxableSumm = ( CalcTax.GetRateByName("Ф1")+ CalcTax.GetRateByName("Ф2")+
CalcTax.GetRateByName("Ф3"))* CalcTax.GetRateByName("БД") * K1*
CalcTax.GetRateByName("К2") * CalcTax.GetRateByName("К3")
NalogTmp = CalcTax.TaxableSumm * CalcTax.GetRateByName("%") / 100
CalcTax.Summa = NalogTmp
else ' Здесь надо настроить суммирование своих налогов по ВД
NalogTmp =CalcTax.SumTax (CalcTax.IDTaxPeriod, CalcTax.IDTaxPeriod, "ЕНВД04_A") ' и т.д.
прибавить значения из всех своих карточек и/или изменить название налога "ЕНВД04_A" на свое
' Сумма единого налога, исчисленная за налоговый период, уменьшается на сумму страховых
взносов на обязательное пенсионное страхование, уплаченных за этот же период времени при
выплате налогоплательщиками вознаграждений своим работникам, занятым в тех сферах
деятельности налогоплательщика, по которым уплачивается единый налог... При этом сумма
единого налога не может быть уменьшена более чем на 50 процентов. (Статья 346.32)
PFR =CalcTax.SumTaxPay1(CalcTax.IDTaxPeriod-2 , CalcTax.IDTaxPeriod, "ПФРнак",0) +
CalcTax.SumTaxPay1(CalcTax.IDTaxPeriod-2 , CalcTax.IDTaxPeriod, "ПФРстр",0)
SAll= CalcTax.SumTax (CalcTax.IDTaxPeriod-2 , CalcTax.IDTaxPeriod, "ПФРнак")
+CalcTax.SumTax (CalcTax.IDTaxPeriod-2 , CalcTax.IDTaxPeriod, "ПФРстр") ' общая сумма
начисленных взносов (суммы «ПФРстр» и «ПФРнак») во всем видам деятельности (по ЕНВД и по
общему режиму налогообложения)
SVd = CalcTax.SumTaxPF (CalcTax.IDTaxPeriod-2 , CalcTax.IDTaxPeriod, 1, "", 0,"") ' сумма
взносов в пенсионный фонд (суммы «ПФРстр» и «ПФРнак»), начисленная с видов оплат,
облагаемых ЕНВД
IF SAll<>0 then
PFR = PFR*SVd/SAll
end if
if PFR <=NalogTmp * 50 /100 then
NalogTmp = NalogTmp - PFR
else
NalogTmp =NalogTmp * 50 /100
End if
' сумма налога уменьшается на сумму пособий по временно нетрудоспособности, выплаченных за
счет собственных средств работодателя (Федеральный закон №190-ФЗ от 31.12.2002)
NalogTmp = NalogTmp - CalcTax.SumInOutCodeAll ("БОЛ", CalcTax.IDTaxPeriod-2 ,
CalcTax.IDTaxPeriod, "")

```

```

if NalogTMP > 0 then
    CalcTax.Summa = NalogTmp
else
    CalcTax.Summa = 0
end if
end if

```

### 11.5.2 Системный алгоритм расчета единого налога по упрощенной системе налогообложения

Системный алгоритм расчета Единого налога по упрощенной системе налогообложения настроен в соответствии с требованиями Налогового кодекса РФ (глава 26.2. Упрощенная система налогообложения) со всеми изменениями и дополнениями.

В соответствии с Налоговым Кодексом расчет суммы налога в течение налогового периода (года) осуществляется кассовым методом нарастающим итогом. При расчете налога его сумма уменьшается на суммы “ранее уплаченных сумм квартальных авансовых платежей по налогу” (п.3 Статья 346.21. Порядок исчисления и уплаты налога). Поэтому сумма Справки начисления налога за каждый последующий отчетный налоговый период (в течение одного налогового периода (года)) учитывает произведенные в течение этого отчетного налогового периода платежи по налогу.

```

Dim TaxMin, mDateBeg, NalogTmp, PFR, LastYrPer, Defc
Do while True
If Year(CalcTax.TaxPeriodDateEnd) = Year(GlobVars.DateBegSimplyTax) Then 'для переходного
периода добавляем документы прошлого года; для вновь созданного предприятия дата начала
работы должна совпадать с датой начала применения упрощенной системы
    mDateBeg = DateAdd("yyyy", -1, GlobVars.DateBegSimplyTax)
ElseIf Year(CalcTax.TaxPeriodDateEnd) > Year(GlobVars.DateBegSimplyTax) then
    mDateBeg = FirstYearDate(CalcTax.TaxPeriodDateEnd)
else
    msgbox "Неверно задан налоговый период справки"
    Exit Do
end if
BookInkome.InitDateBetween mDateBeg, CalcTax.TaxPeriodDateEnd ' нарастающим итогом с
начала года
if GlobVars.ObjectTax = 0 then ' Доход
    CalcTax.TaxableSumm =BookInkome.SumInc(0)
    NalogTmp = CalcTax.TaxableSumm * CalcTax.Tax.ActiveProc / 100
    mDateBeg = FirstYearDate(CalcTax.TaxPeriodDateEnd) ' дата начала года
' сумма налога уменьшается на сумму страховых взносов на обязательное пенсионное
страхование, уплачиваемых за этот же период времени... При этом сумма налога (квартальных
авансовых платежей по налогу) не может быть уменьшена более чем на 50 процентов.
    PFR =CalcTax.SumTaxPay1(GlobVars.DateToPeriod( mDateBeg), CalcTax.IDTaxPeriod,
"ПФРнак",1) + CalcTax.SumTaxPay1(GlobVars.DateToPeriod( mDateBeg), CalcTax.IDTaxPeriod,

```

```
"ПФРстр",1) ' платежи в ПФР за периоды, предшествующие переходу на упрощенную систему
налогообложение не должны иметь признак "Сделать запись в КДР"
  if PFR <=NalogTmp * 50 /100 then
    NalogTmp = NalogTmp - PFR
  else
    if NalogTmp >= 0 then
      NalogTmp =NalogTmp * 50 /100
    end if
  end if
' сумма налога уменьшается за сумму пособий по временной нетрудоспособности, выплаченных
за счет собственных средств работодателя (Федеральный закон " 190-ФЗ от 31.12.2002)
NalogTmp = NalogTmp - CalcTax.SumInOutCodeAll ("БОЛ", GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod, "")
' сумма налога уменьшается на сумму ранее уплаченных сумм квартальных авансовых платежей
по налогу (Статья 346.21. п. 3)
  NalogTmp = NalogTmp - CalcTax.SumTaxPay1(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod , "ЕУСН",0)
' в текущем году не надо учитывать платежи по ЕУСН по окончательным расчетам за прошлые
года или по начальным остаткам
  LastYrPer = DateAdd("y", -1, mDateBeg)
  NalogTmp = NalogTmp +CalcTax.SumTaxPay3(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod, 1, GlobVars.DateToPeriod(LastYrPer), "ЕУСН", 0)
  if NalogTMP > 0 then
    CalcTax.Summa = NalogTmp
  else
    CalcTax.Summa = 0
  end if
else ' 1 Доход-Расход
  CalcTax.TaxableSumm =BookInkome.SumInc(0)
'if DatePart("Q", CalcTax.TaxPeriodDateEnd)=4 then
  Defc = Val(InputBox ("Сумма убытка предыдущих налоговых периодов " , , 0))
  if CalcTax.TaxableSumm > 0 AND CalcTax.TaxableSumm*30/100<Defc then
    CalcTax.TaxableSumm = CalcTax.TaxableSumm*70/100
  else
    if CalcTax.TaxableSumm > 0 then
      CalcTax.TaxableSumm = CalcTax.TaxableSumm - Defc
    end if
  end if
'end if
  NalogTmp = CalcTax.TaxableSumm * CalcTax.Tax.ActiveProc / 100
' Сумма минимального налога исчисляется в размере 1 процента налоговой базы, которой
являются доходы... Минимальный налог уплачивается в случае, если сумма исчисленного в
общем порядке налога меньше суммы исчисленного минимального налога. (Статья 346.18)
  TaxMin = BookInkome.SumInc(1) / 100 ' 1 % от дохода
```

```

if DatePart("Q", CalcTax.TaxPeriodDateEnd)=4 AND NalogTmp < TaxMin then
    CalcTax.Summa = TaxMin
    msgbox "Начисляется минимальный налог"
    NalogTmp = TaxMin
' сумма налога уменьшается на сумму ранее уплаченных сумм квартальных авансовых платежей
по налогу (Статья 346.21. п. 3)
    mDateBeg = FirstYearDate(CalcTax.TaxPeriodDateEnd) ' дата начала года
    NalogTmp = NalogTmp - CalcTax.SumTaxPay1(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod , "ЕУСН",0)
' в текущем году не надо учитывать платежи по ЕУСН по окончательным расчетам за прошлые
года или по начальным остаткам
    LastYrPer = DateAdd("y", -1, mDateBeg)
    NalogTmp = NalogTmp +CalcTax.SumTaxPay3(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod, 1, GlobVars.DateToPeriod(LastYrPer), "ЕУСН", 0)
    If NalogTmp >0 then
        CalcTax.Summa = NalogTmp
    else
        CalcTax.Summa = 0
    end if
else
    if NalogTmp > 0 then
' сумма налога уменьшается на сумму ранее уплаченных сумм квартальных авансовых платежей
по налогу (Статья 346.21. п. 3)
        mDateBeg = FirstYearDate(CalcTax.TaxPeriodDateEnd) ' дата начала года
        NalogTmp = NalogTmp - CalcTax.SumTaxPay1(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod , "ЕУСН",0)
' в текущем году не надо учитывать платежи по ЕУСН по окончательным расчетам за прошлые
года или по начальным остаткам
        LastYrPer = DateAdd("y", -1, mDateBeg)
        NalogTmp = NalogTmp +CalcTax.SumTaxPay3(GlobVars.DateToPeriod( mDateBeg) ,
CalcTax.IDTaxPeriod, 1, GlobVars.DateToPeriod(LastYrPer), "ЕУСН", 0)
        CalcTax.Summa = NalogTmp
    else
        CalcTax.Summa = 0
    end if
end if
end if
Exit Do
Loop

```

**Система управления предприятием БЭСТ-ОФИС, версия 3.5.**

**© Интеллект-Сервис, 2002**