**Практическая работа № 3**

**Бухгалтерская подсистема: расходные документы и переоценка валютных остатков**

**Цель работы:** разработать новые и модифицировать существующие документы для работы с бухгалтерской подсистемой

**Проведение документа ОтпускМатериаловМастеру**

Документ **ОтпускМатериаловМастеру** используется для отражения *списания материалов в производство*. Мы, при списании материалов этим документом, будем списывать их *по* средней себестоимости в рамках отдельного ответственного лица, за которым эти материалы числятся, причем, числятся исключительно на счете Материалы **для производства**. Наш документ **ОтпускМатериаловМастеру** уже используется в системе для решения задач оперативного учета материалов, мы, чтобы не усложнять задачу, оставим процедуру проведения *по* регистру накопления в исходном состоянии, добавив в *модуль* документа еще одну процедуру следующего вида:

Процедура ОбработкаПроведенияПоБухучету(Отказ)

КонецПроцедуры

Вызовем эту процедуру в начале процедуры ОбработкаПроведения, в итоге, получим такой код вызова:

Процедура ОбработкаПроведения(Отказ, РежимПроведения)

ОбработкаПроведенияПоБухучету(Отказ);

…

КонецПроцедуры

Здесь мы вызываем процедуру и передаем ей *параметр* Отказ. Установка этого параметра в процедуре ОбработкаПроведенияПоБухучету в *значение* Истина позволит нам, при необходимости, отказаться от проведения документа, при предварительной проверке перед формированием движений *по* *бухгалтерскому учету*.

В нашей конфигурации потенциально возможна ситуация, когда количество материалов определенного вида, числящееся *по* *бухгалтерскому учету* на счете **Материалы для производства** за определенным сотрудником, будет не соответствовать общему количеству материалов, которое за ним числится. Например, если в документе запрошено списание 5 единиц некоей позиции, и, в то же время, *по* данным регистра накопления за сотрудником числится 6 единиц, а *по* данным остатка этого вида материала на счете **Материалы для производства**, будет получено количество 4, документ проведен не будет. В данном случае будем считать, что бухгалтерские данные имеют приоритетное *значение*, и, поэтому, возникновение подобной ситуации приведет к отказу от проведения документа.

Данные для проведения документа мы получаем двумя запросами. Одним – для формирования движений *по* регистру накопления, другим – *по* *регистру бухгалтерии*. Часть данных, получаемых запросами, дублируется. Для целей оптимальности можно объединить эти запросы в один и соответствующим образом переработать процедуры проведения, так же объединив их.

В основе процедуры проведения *по* *бухгалтерскому учету* документа **ОтпускМатериаловМастеру** лежит *запрос*. *Запрос* будет оперировать, во-первых, данными табличной части документа, во-вторых – сведениями об остатках номенклатурных позиций на определенном счете с определенным набором *субконто* (**Номенклатура** и **Сотрудник**).

Сначала получим данные, которые нам нужны из табличной части документа. Это можно сделать с помощью такого запроса ([рис. 3.1](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=1#image.3.1)):

ВЫБРАТЬ

ДокМ.Номенклатура,

СУММА(ДокМ.Количество)КАК Количество

ИЗ

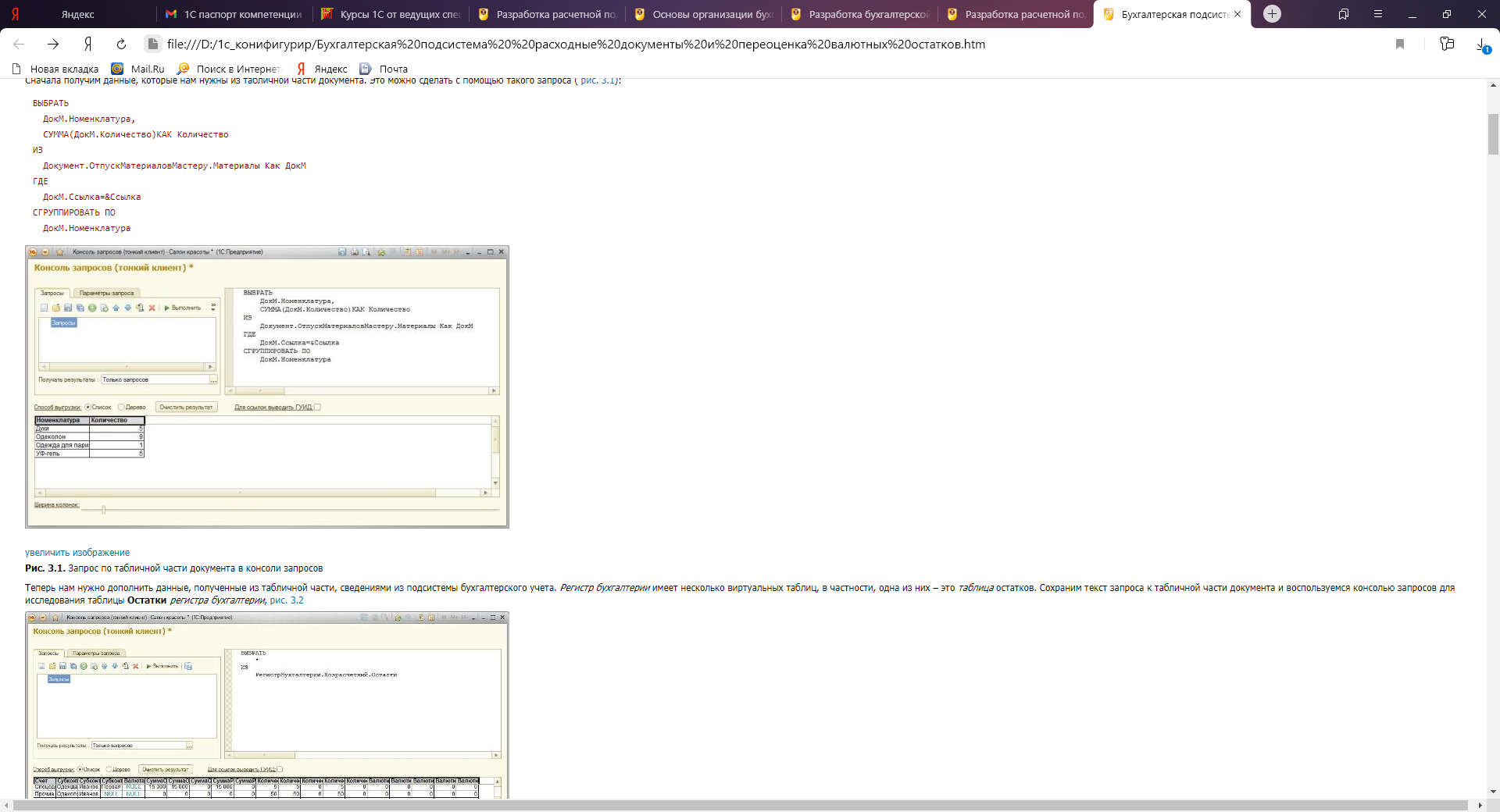
Документ.ОтпускМатериаловМастеру.Материалы Как ДокМ

ГДЕ

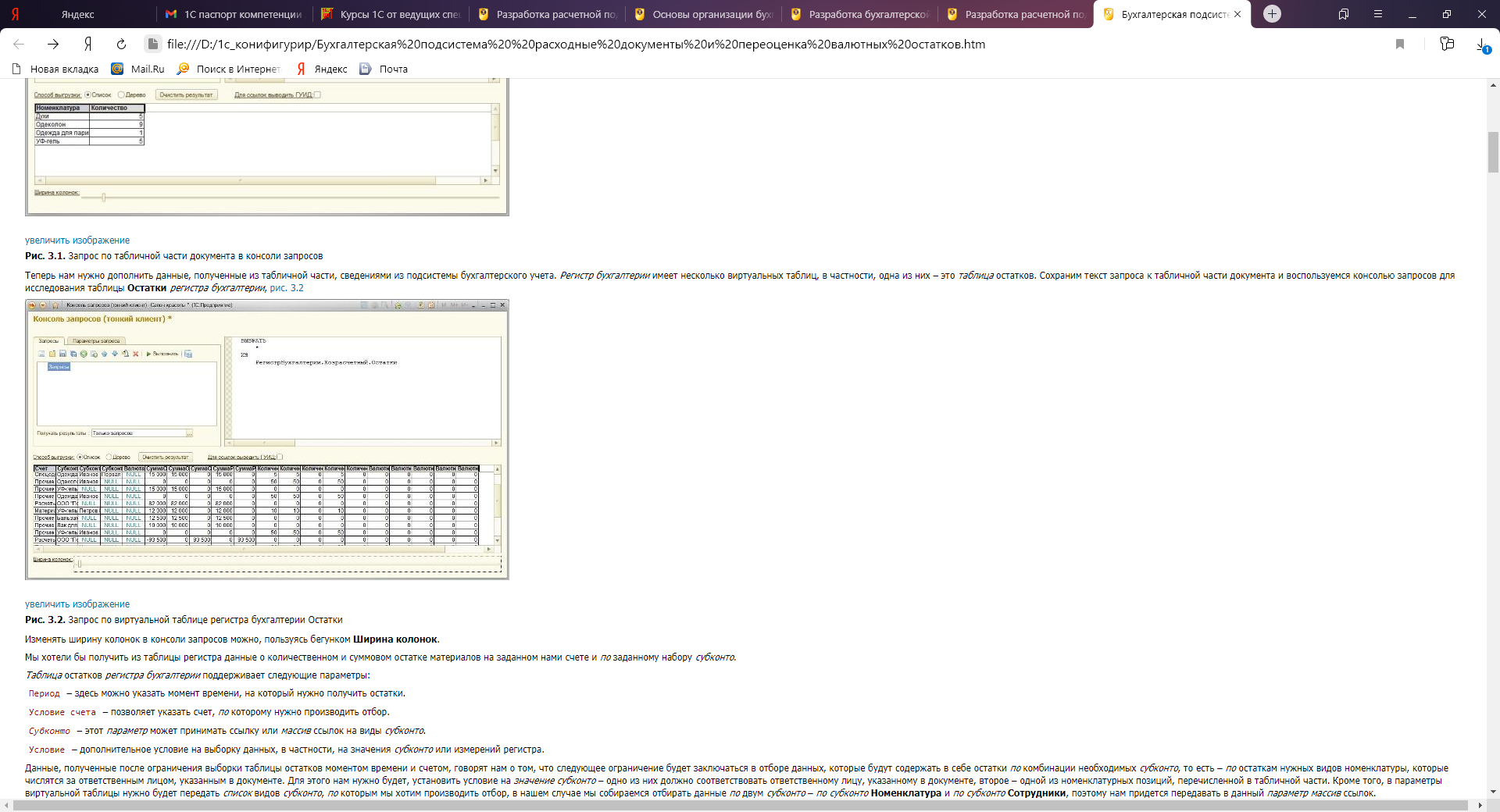
ДокМ.Ссылка=&Ссылка

СГРУППИРОВАТЬ ПО

ДокМ.Номенклатура

  
**Рис. 3.1.**Запрос по табличной части документа в консоли запросов

Теперь нам нужно дополнить данные, полученные из табличной части, сведениями из подсистемы бухгалтерского учета. *Регистр бухгалтерии* имеет несколько виртуальных таблиц, в частности, одна из них – это *таблица* остатков. Сохраним текст запроса к табличной части документа и воспользуемся консолью запросов для исследования таблицы **Остатки** *регистра бухгалтерии*, [рис. 3.2](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=1#image.3.2)

  
**Рис. 3.2.**Запрос по виртуальной таблице регистра бухгалтерии Остатки

Изменять ширину колонок в консоли запросов можно, пользуясь бегунком **Ширина колонок**.

Мы хотели бы получить из таблицы регистра данные о количественном и суммовом остатке материалов на заданном нами счете и *по* заданному набору *субконто*.

*Таблица* остатков *регистра бухгалтерии* поддерживает следующие параметры:

Период – здесь можно указать момент времени, на который нужно получить остатки.

Условие счета – позволяет указать счет, *по* которому нужно производить отбор.

*Субконто* – этот *параметр* может принимать ссылку или *массив* ссылок на виды *субконто*.

Условие – дополнительное условие на выборку данных, в частности, на значения *субконто* или измерений регистра.

Данные, полученные после ограничения выборки таблицы остатков моментом времени и счетом, говорят нам о том, что следующее ограничение будет заключаться в отборе данных, которые будут содержать в себе остатки *по* комбинации необходимых *субконто*, то есть – *по* остаткам нужных видов номенклатуры, которые числятся за ответственным лицом, указанным в документе. Для этого нам нужно будет, установить условие на *значение* *субконто* – одно из них должно соответствовать ответственному лицу, указанному в документе, второе – одной из номенклатурных позиций, перечисленной в табличной части. Кроме того, в параметры виртуальной таблицы нужно будет передать *список* видов *субконто*, *по* которым мы хотим производить отбор, в нашем случае мы собираемся отбирать данные *по* двум *субконто* – *по* *субконто* **Номенклатура** и *по* *субконто* **Сотрудники**, поэтому нам придется передавать в данный *параметр* *массив* ссылок.

Для обеспечения наиболее оптимального исполнения запроса, нужно как можно более полно использовать параметры виртуальных таблиц (в частности, это касается таблицы **Остатки** *регистра бухгалтерии*), именно поэтому в том случае, если существуют *альтернативные* способы решения одной и той же задачи *по* получению данных, мы выбираем тот из них, который основан на параметрах виртуальной таблицы.

На данном этапе работы мы приходим к необходимости пользоваться данными, которые были получены запросом *по* табличной части документа, в запросе, который будет выполняться *по* таблице остатков *регистра бухгалтерии*. Лучше всего для решения подобной задачи подойдет оформление результатов первого запроса в виде виртуальной таблицы и последующее использование этой виртуальной таблицы в запросе к таблице остатков *регистра бухгалтерии*.

В итоге, после работы с консолью запросов, мы имеем такой текст запроса:

ВЫБРАТЬ

ДокМ.Номенклатура,

СУММА(ДокМ.Количество)КАК Количество

ПОМЕСТИТЬ ДокМТч

ИЗ

Документ.ОтпускМатериаловМастеру.Материалы Как ДокМ

ГДЕ

ДокМ.Ссылка=&Ссылка

СГРУППИРОВАТЬ ПО

ДокМ.Номенклатура

;

ВЫБРАТЬ

ДокМТч.Номенклатура,

ДокМТч.Количество,

ЕСТЬNULL(Остатки.СуммаОстаток,0) КАК СуммаОстаток,

ЕСТЬNULL(Остатки.КоличествоОстаток, 0) КАК КоличествоОстаток

ИЗ

ДокМТч КАК ДокМТч

ЛЕВОЕ СОЕДИНЕНИЕ

РегистрБухгалтерии.Хозрасчетный.Остатки (

&МоментВремени,

Счет=&Счет,

&Субконто,

Субконто1 В(ВЫБРАТЬ

ДокМТч.Номенклатура

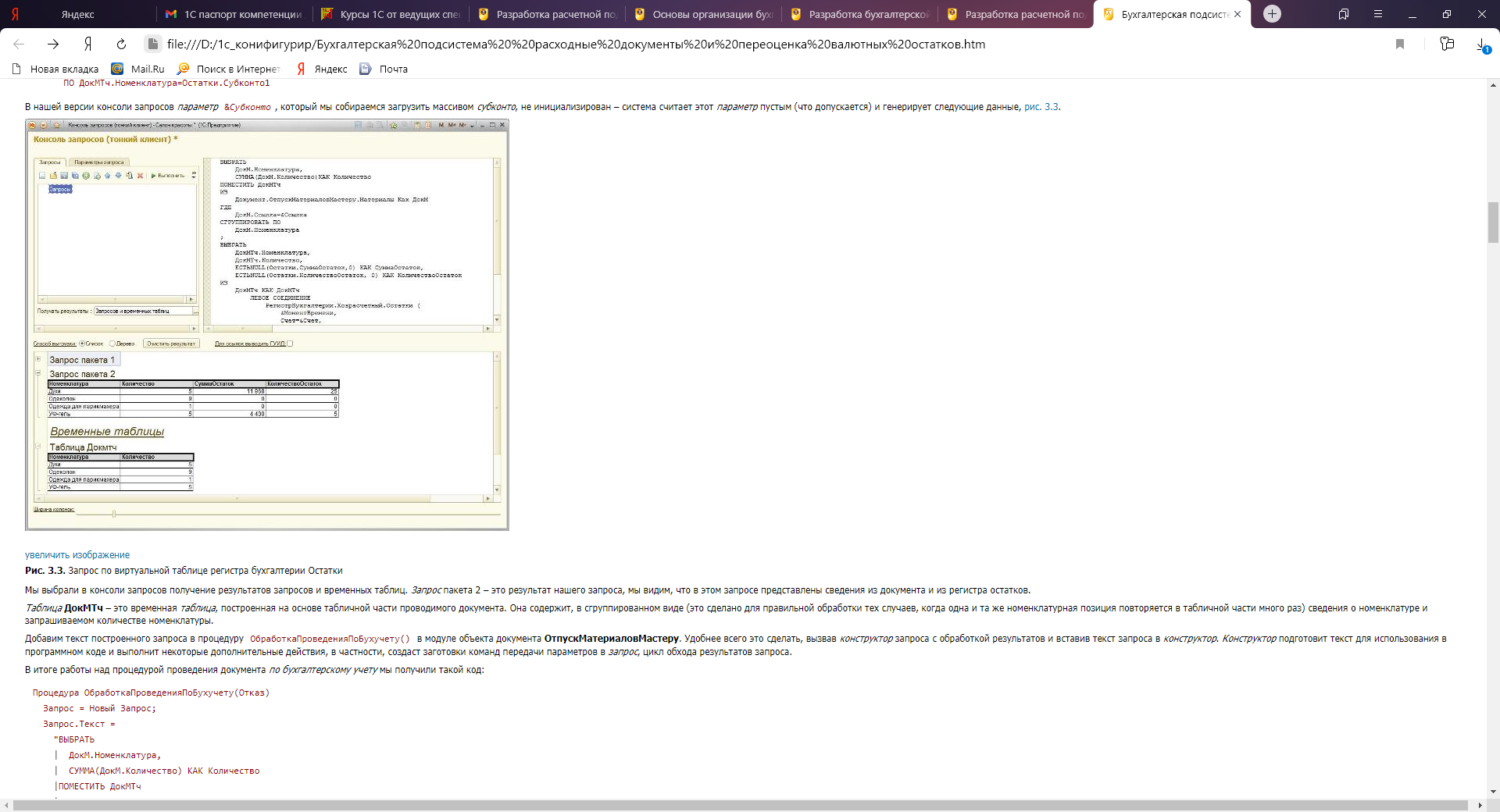
ИЗ

ДокМТч)

И Субконто2=&Ответственный) КАК Остатки

ПО ДокМТч.Номенклатура=Остатки.Субконто1

В нашей версии консоли запросов *параметр* &*Субконто*, который мы собираемся загрузить массивом *субконто*, не инициализирован – система считает этот *параметр* пустым (что допускается) и генерирует следующие данные, [рис. 3.3](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=2#image.3.3).

  
**Рис. 3.3.**Запрос по виртуальной таблице регистра бухгалтерии Остатки

Мы выбрали в консоли запросов получение результатов запросов и временных таблиц. *Запрос* пакета 2 – это результат нашего запроса, мы видим, что в этом запросе представлены сведения из документа и из регистра остатков.

*Таблица* **ДокМТч** – это временная *таблица*, построенная на основе табличной части проводимого документа. Она содержит, в сгруппированном виде (это сделано для правильной обработки тех случаев, когда одна и та же номенклатурная позиция повторяется в табличной части много раз) сведения о номенклатуре и запрашиваемом количестве номенклатуры.

Добавим текст построенного запроса в процедуру ОбработкаПроведенияПоБухучету() в модуле объекта документа **ОтпускМатериаловМастеру**. Удобнее всего это сделать, вызвав *конструктор* запроса с обработкой результатов и вставив текст запроса в *конструктор*. *Конструктор* подготовит текст для использования в программном коде и выполнит некоторые дополнительные действия, в частности, создаст заготовки команд передачи параметров в *запрос*, цикл обхода результатов запроса.

В итоге работы над процедурой проведения документа *по* *бухгалтерскому учету* мы получили такой код:

Процедура ОбработкаПроведенияПоБухучету(Отказ)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ДокМ.Номенклатура,

| СУММА(ДокМ.Количество) КАК Количество

|ПОМЕСТИТЬ ДокМТч

|ИЗ

| Документ.ОтпускМатериаловМастеру.Материалы КАК ДокМ

|ГДЕ

| ДокМ.Ссылка = &Ссылка

|

|СГРУППИРОВАТЬ ПО

| ДокМ.Номенклатура

|;

|

|////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| ДокМТч.Номенклатура,

| ДокМТч.Количество,

| ЕСТЬNULL(Остатки.СуммаОстаток, 0) КАК СуммаОстаток,

| ЕСТЬNULL(Остатки.КоличествоОстаток, 0) КАК КоличествоОстаток

|ИЗ

| ДокМТч КАК ДокМТч

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрБухгалтерии.Хозрасчетный.Остатки(

| &МоментВремени,

| Счет = &Счет,

| &Субконто,

| Субконто1 В

| (ВЫБРАТЬ

| ДокМТч.Номенклатура

| ИЗ

| ДокМТч)

| И Субконто2 = &Ответственный) КАК Остатки

| ПО ДокМТч.Номенклатура = Остатки.Субконто1";

//Устанавливаем параметры запроса

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("Ответственный", ОтветственныйСотрудник);

Запрос.УстановитьПараметр("Ссылка", Ссылка);

//В этом документе мы имеем дело исключительно со счетом МатериалыДляПроизводства

Запрос.УстановитьПараметр("Счет",

ПланыСчетов.Хозрасчетный.МатериалыДляПроизводства);

//Этот массив будет передан в качестве параметра виртуальной таблицы остатков

//регистра бухгалтерии

Субконто=Новый Массив(2);

Субконто[0]=ПланыВидовХарактеристик.ВидыСубконто.Номенклатура;

Субконто[1]=ПланыВидовХарактеристик.ВидыСубконто.Сотрудники;

Запрос.УстановитьПараметр("Субконто", Субконто);

//Выполним запрос

Результат = Запрос.Выполнить();

ВыборкаДЗ = Результат.Выбрать();

//Включаем запись движений по регистру бухгалтерии

Движения.Хозрасчетный.Записывать=Истина;

//Обходим полученную выборку

Пока ВыборкаДЗ.Следующий() Цикл

//Проверяем, достаточно ли имеющихся материалов для

//корректного их списания в документе

Если ВыборкаДЗ.Количество>ВыборкаДЗ.КоличествоОстаток Тогда

Сообщить("Недостаточное количество товара (бухучет) "+

ВыборкаДЗ.Номенклатура+" необходимо "+ВыборкаДЗ.Количество+

", в наличии "+ВыборкаДЗ.КоличествоОстаток);

Отказ=Истина;

Движения.Хозрасчетный.Записывать=Ложь;

КонецЕсли;

Если Отказ Тогда

//Не идем дальше если Отказ установлен в Истина, то есть

//движения формироваться не будут

Продолжить;

КонецЕсли;

//Формируем движение вида Дт Капитал Кт Материалы

Движение=Движения.Хозрасчетный.Добавить();

Движение.Период=Дата;

Движение.СчетДт=ПланыСчетов.Хозрасчетный.Капитал;

Движение.СчетКт=ПланыСчетов.Хозрасчетный.МатериалыДляПроизводства;

Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Номенклатура]=

ВыборкаДЗ.Номенклатура;

Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Сотрудники]=

ОтветственныйСотрудник;

Движение.КоличествоКт=ВыборкаДЗ.Количество;

Движение.Сумма=ВыборкаДЗ.Количество\*ВыборкаДЗ.СуммаОстаток/

ВыборкаДЗ.КоличествоОстаток;

КонецЦикла;

КонецПроцедуры

В приведенной вашему вниманию процедуре самой сложной для понимания частью является именно *запрос*. Выше мы занимались его конструированием, теперь кратко пройдемся *по* основным частям этого запроса. Сначала мы получаем данные из табличной части проводимого документа (это **Номенклатура** и **Количество**), группируем полученную таблицу *по* полю **Номенклатура** и помещаем результат во временную таблицу с названием **ДокМТч**.

Далее мы используем полученную временную таблицу в запросе, где, обращаясь сначала к этой таблице, получаем из нее поля **Номенклатура** и **Количество**, после чего *левым соединением* присоединяем к результатам запроса данные из таблицы *регистра бухгалтерии* **Остатки**, ориентируясь на *поле* **Номенклатура** (в таблице регистра *ссылка* на соответствующий элемент справочника **Номенклатура** может находится в первом *субконто*). При получении данных из виртуальной таблицы **Хозрасчетный.Остатки** мы, кроме прочего, отбираем их *по* значениям **Субконто1**, которое должно входить в набор значений *субконто* из нашего документа (здесь мы так же пользуемся созданной выше временной таблицей в выражении Субконто1 В (ВЫБРАТЬ …)), и Субконто2, которое мы можем однозначно отобрать *по* реквизиту документа **ОтветственныйСотрудник**.

На тот случай, если *запрос* в полях количественных и суммовых остатков *по* регистру возвратит *значение* NULL (а это возможно в том случае, если нет остатков *по* запрошенной в документе номенклатурной позиции за ответственным лицом, с которого списываются материалы в документе), мы используем *выражение* ЕСТЬNULL() для того, чтобы привести NULL к нулю.

При передаче в *запрос* параметров мы, помимо остальных одиночных параметров, передаем *параметр*-*массив*. Перед установкой этого параметра мы создаем и заполняем *массив* видами *субконто*. *Нумерация* элементов массива начинается с 0.

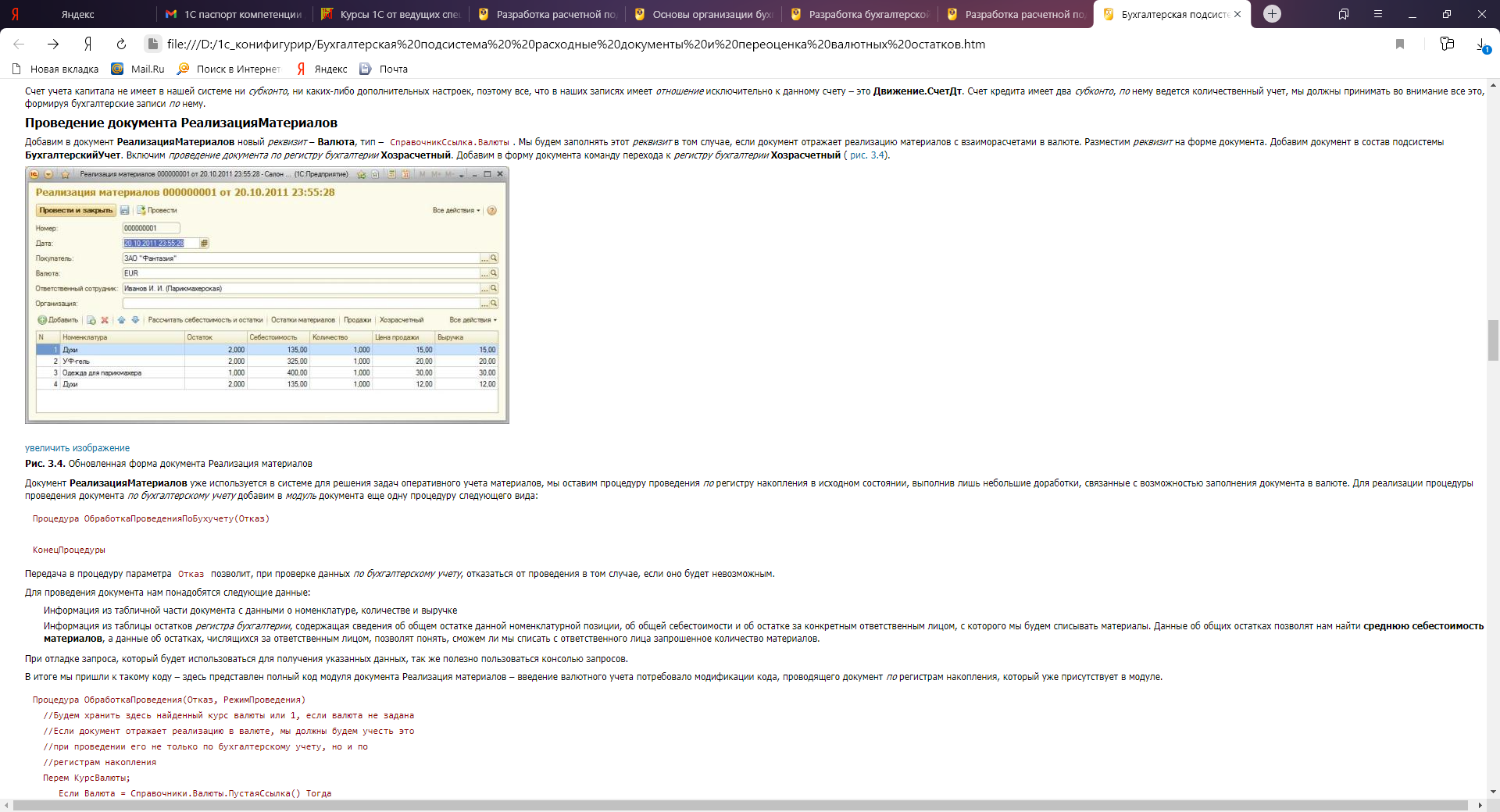
После выполнения запроса и размещения выборки из результатов запроса в переменной ВыборкаДЗ, мы попадаем в цикл перебора выборки, где сначала проверяем, достаточное ли количество номенклатуры каждого вида числится за ответственным лицом, указанным в документе на счете **МатериалыДляПроизводства**. Если нам не хватает материалов – мы выводим сообщение пользователю, уточняя, что данные получены при попытке провести документ *по* подсистеме бухгалтерского учета.

Если проверка прошла нормально, мы переходим к формированию движений. Документ формирует движение вида Дт Капитал Кт Материалы. Мы списываем материалы, отпущенные в производство, прямо на счет учета капитала. В реальном учете себестоимость материалов, ушедших в производство, проделывает достаточно долгий *путь*, прежде чем повлияет на состояние прибыли, а потом, возможно, и на состояние капитала, но сущность записи остается такой же.

Счет учета капитала не имеет в нашей системе ни *субконто*, ни каких-либо дополнительных настроек, поэтому все, что в наших записях имеет *отношение* исключительно к данному счету – это **Движение.СчетДт**. Счет кредита имеет два *субконто*, *по* нему ведется количественный учет, мы должны принимать во внимание все это, формируя бухгалтерские записи *по* нему.

**Проведение документа РеализацияМатериалов**

Добавим в документ **РеализацияМатериалов** новый *реквизит* – **Валюта**, тип – СправочникСсылка.Валюты. Мы будем заполнять этот *реквизит* в том случае, если документ отражает реализацию материалов с взаиморасчетами в валюте. Разместим *реквизит* на форме документа. Добавим документ в состав подсистемы **БухгалтерскийУчет**. Включим *проведение документа* *по* *регистру бухгалтерии* **Хозрасчетный**. Добавим в форму документа команду перехода к *регистру бухгалтерии* **Хозрасчетный** ([рис. 3.4](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=3#image.3.4)).

  
**Рис. 3.4.**Обновленная форма документа Реализация материалов

Документ **РеализацияМатериалов** уже используется в системе для решения задач оперативного учета материалов, мы оставим процедуру проведения *по* регистру накопления в исходном состоянии, выполнив лишь небольшие доработки, связанные с возможностью заполнения документа в валюте. Для реализации процедуры проведения документа *по* *бухгалтерскому учету* добавим в *модуль* документа еще одну процедуру следующего вида:

Процедура ОбработкаПроведенияПоБухучету(Отказ)

КонецПроцедуры

Передача в процедуру параметра Отказ позволит, при проверке данных *по* *бухгалтерскому учету*, отказаться от проведения в том случае, если оно будет невозможным.

Для проведения документа нам понадобятся следующие данные:

* Информация из табличной части документа с данными о номенклатуре, количестве и выручке
* Информация из таблицы остатков *регистра бухгалтерии*, содержащая сведения об общем остатке данной номенклатурной позиции, об общей себестоимости и об остатке за конкретным ответственным лицом, с которого мы будем списывать материалы. Данные об общих остатках позволят нам найти **среднюю себестоимость материалов**, а данные об остатках, числящихся за ответственным лицом, позволят понять, сможем ли мы списать с ответственного лица запрошенное количество материалов.

При отладке запроса, который будет использоваться для получения указанных данных, так же полезно пользоваться консолью запросов.

В итоге мы пришли к такому коду – здесь представлен полный код модуля документа Реализация материалов – введение валютного учета потребовало модификации кода, проводящего документ *по* регистрам накопления, который уже присутствует в модуле.

Процедура ОбработкаПроведения(Отказ, РежимПроведения)

//Будем хранить здесь найденный курс валюты или 1, если валюта не задана

//Если документ отражает реализацию в валюте, мы должны будем учесть это

//при проведении его не только по бухгалтерскому учету, но и по

//регистрам накопления

Перем КурсВалюты;

Если Валюта = Справочники.Валюты.ПустаяСсылка() Тогда

КурсВалюты=1;

Иначе

КурсВалюты=ПолучитьКурсВалюты(Отказ);

КонецЕсли;

//Вызываем процедуру проведения по бухгалтерскому учету

ОбработкаПроведенияПоБухучету(Отказ, КурсВалюты);

//Приступаем к проведению документа по регистрам накопления

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ДокМ.Номенклатура,

| СУММА(ДокМ.Количество) КАК Количество,

| СУММА(ДокМ.Выручка) КАК Выручка,

| МАКСИМУМ(ЕСТЬNULL(ОстМ.КоличествоОстаток, 0)) КАК КоличествоОстатков,

| МАКСИМУМ(ЕСТЬNULL(ОстМ.СуммаОстаток, 0)) КАК СуммаОстатков

|ИЗ

| Документ.РеализацияМатериалов.Материалы КАК ДокМ

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.ОстаткиМатериалов.Остатки

(&МоментВремени, ОтветственныйСотрудник = &ОтвСотр) КАК ОстМ

| ПО ДокМ.Номенклатура = ОстМ.Номенклатура

|ГДЕ

| ДокМ.Ссылка = &Ссылка

|

|СГРУППИРОВАТЬ ПО

| ДокМ.Номенклатура";

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("ОтвСотр", ОтветственныйСотрудник);

Запрос.УстановитьПараметр("Ссылка", Ссылка);

Результат = Запрос.Выполнить();

ВыборкаДЗ = Результат.Выбрать();

Движения.ОстаткиМатериалов.Записывать=Истина;

Движения.Продажи.Записывать=Истина;

Пока ВыборкаДЗ.Следующий() Цикл

Если ВыборкаДЗ.Количество>ВыборкаДЗ.КоличествоОстатков Тогда

Сообщить("Недостаточное количество товара "

+ВыборкаДЗ.Номенклатура

+", необходимо "+ВыборкаДЗ.Количество+", в наличии "

+ВыборкаДЗ.КоличествоОстатков);

Отказ=Истина;

Движения.ОстаткиМатериалов.Записывать=Ложь;

Движения.Продажи.Записывать=Ложь;

КонецЕсли;

Если Отказ Тогда

Продолжить;

КонецЕсли;

//Движения по регистру ОстаткиМатериалов не зависят от валюты документа

Движение=Движения.ОстаткиМатериалов.Добавить();

Движение.ВидДвижения=ВидДвиженияНакопления.Расход;

Движение.Период=Дата;

Движение.Номенклатура=ВыборкаДЗ.Номенклатура;

Движение.Количество=ВыборкаДЗ.Количество;

Движение.Сумма=ВыборкаДЗ.Количество\*ВыборкаДЗ.СуммаОстатков

/ВыборкаДЗ.КоличествоОстатков;

Движение.ОтветственныйСотрудник=ОтветственныйСотрудник;

//Движения по регистру Продажи должны выполняться с учетом курса валюты

Движение=Движения.Продажи.Добавить();

Движение.Период=Дата;

Движение.Номенклатура=ВыборкаДЗ.Номенклатура;

Движение.Количество=ВыборкаДЗ.Количество;

Движение.Себестоимость=ВыборкаДЗ.Количество\*ВыборкаДЗ.СуммаОстатков

/ВыборкаДЗ.КоличествоОстатков;

//Если валюта задана и документ заполнен в валюте, это

//позволит сформировать верное значение выручки

//Если валюта не задана, переменная КурсВалюты принимает значение 1

//Выручка так же отразится верно

Движение.Выручка=ВыборкаДЗ.Выручка\*КурсВалюты;

Движение.ОтветственныйСотрудник=ОтветственныйСотрудник;

Движение.Контрагент=Покупатель;

КонецЦикла;

КонецПроцедуры

Функция ПолучитьКурсВалюты(Отказ)

//Получаем курс валюты, заданной в документе

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| КурсыВалютСрезПоследних.Курс

|ИЗ

| РегистрСведений.КурсыВалют.СрезПоследних(&МоментВремени, Валюта =

&Валюта) КАК КурсыВалютСрезПоследних";

Запрос.УстановитьПараметр("Валюта", Валюта);

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Результат = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = Результат.Выбрать();

Если ВыборкаДетальныеЗаписи.Следующий() Тогда

Возврат(ВыборкаДетальныеЗаписи.Курс);

Иначе

Сообщить("Не найден курс валюты!");

Отказ=Истина;

КонецЕсли;

КонецФункции

Процедура ОбработкаПроведенияПоБухучету(Отказ, КурсВалюты)

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ДокМ.Номенклатура,

| СУММА(ДокМ.Количество) КАК Количество,

| СУММА(ДокМ.Выручка) КАК Выручка

|ПОМЕСТИТЬ ДокМТч

|ИЗ

| Документ.РеализацияМатериалов.Материалы КАК ДокМ

|ГДЕ

| ДокМ.Ссылка = &Ссылка

|

|СГРУППИРОВАТЬ ПО

| ДокМ.Номенклатура

|;

|

|//////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| ХозрасчетныйОстатки.Субконто1,

| СУММА(ХозрасчетныйОстатки.КоличествоОстаток) КАК КоличествоОстаток,

| СУММА(ВЫБОР

| КОГДА ХозрасчетныйОстатки.Субконто2 = &Ответственный

| ТОГДА ХозрасчетныйОстатки.КоличествоОстаток

| КОНЕЦ) КАК КоличествоЗаОтветственным,

| СУММА(ХозрасчетныйОстатки.СуммаОстаток) КАК СуммаОстаток

|ПОМЕСТИТЬ ОстаткиПоСчету

|ИЗ

| РегистрБухгалтерии.Хозрасчетный.Остатки(

| &МоментВремени,

| Счет = &Счет,

| &Субконто,

| Субконто1 В

| (ВЫБРАТЬ

| ДокМТч.Номенклатура

| ИЗ

| ДокМТч)) КАК ХозрасчетныйОстатки

|

|СГРУППИРОВАТЬ ПО

| ХозрасчетныйОстатки.Субконто1

|;

|

|/////////////////////////////////////////////////////////////////////////

|ВЫБРАТЬ

| ДокМТч.Номенклатура,

| ДокМТч.Количество,

| ДокМТч.Выручка,

| ЕСТЬNULL(ОстаткиПоСчету.КоличествоОстаток, 0) КАК КоличествоОстаток,

| ЕСТЬNULL(ОстаткиПоСчету.КоличествоЗаОтветственным, 0)

КАК КоличествоЗаОтветственным,

| ЕСТЬNULL(ОстаткиПоСчету.СуммаОстаток, 0) КАК СуммаОстаток

|ИЗ

| ДокМТч КАК ДокМТч

| ЛЕВОЕ СОЕДИНЕНИЕ ОстаткиПоСчету КАК ОстаткиПоСчету

| ПО ДокМТч.Номенклатура = ОстаткиПоСчету.Субконто1";

//Устанавливаем параметры запроса

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());

Запрос.УстановитьПараметр("Ответственный", ОтветственныйСотрудник);

Запрос.УстановитьПараметр("Ссылка", Ссылка);

//В этом документе мы имеем дело исключительно со счетом ПрочиеМатериалы

Запрос.УстановитьПараметр

("Счет", ПланыСчетов.Хозрасчетный.ПрочиеМатериалы);

//Этот массив будет передан в качестве параметра виртуальной таблицы остатков

//регистра бухгалтерии

Субконто=Новый Массив(2);

Субконто[0]=ПланыВидовХарактеристик.ВидыСубконто.Номенклатура;

Субконто[1]=ПланыВидовХарактеристик.ВидыСубконто.Сотрудники;

Запрос.УстановитьПараметр("Субконто", Субконто);

//Выполним запрос

Результат = Запрос.Выполнить();

ВыборкаДЗ = Результат.Выбрать();

//Включаем запись движений по регистру бухгалтерии

Движения.Хозрасчетный.Записывать=Истина;

//Обходим полученную выборку

Пока ВыборкаДЗ.Следующий() Цикл

//Проверяем, достаточно ли имеющихся материалов для

//корректного их списания в документе

Если ВыборкаДЗ.Количество>ВыборкаДЗ.КоличествоЗаОтветственным Тогда

Сообщить("Недостаточное количество товара (бухучет) "

+ВыборкаДЗ.Номенклатура+" необходимо "+ВыборкаДЗ.Количество

+", в наличии "+ВыборкаДЗ.КоличествоЗаОтветственным);

Отказ=Истина;

Движения.Хозрасчетный.Записывать=Ложь;

КонецЕсли;

Если Отказ Тогда

//Не идем дальше если Отказ установлен в Истина, то есть

//движения формироваться не будут

Продолжить;

КонецЕсли;

//Формируем движение вида Дт Капитал Кт Материалы - отражаем себестоимость

//Даже если документ отражает продажу материалов за валюту

//Данная запись выглядит так же, как и при расчетах в рублях

Движение=Движения.Хозрасчетный.Добавить();

Движение.Период=Дата;

Движение.СчетДт=ПланыСчетов.Хозрасчетный.Капитал;

Движение.СчетКт=ПланыСчетов.Хозрасчетный.ПрочиеМатериалы;

Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Номенклатура]

=ВыборкаДЗ.Номенклатура;

Движение.СубконтоКт[ПланыВидовХарактеристик.ВидыСубконто.Сотрудники]

=ОтветственныйСотрудник;

Движение.КоличествоКт=ВыборкаДЗ.Количество;

//Средняя себестоимость в расчете по всем материалам данного вида на счете

Движение.Сумма=ВыборкаДЗ.Количество\*ВыборкаДЗ.СуммаОстаток

/ВыборкаДЗ.КоличествоОстаток;

//Формируем движение вида Дт РасчетыСПокупателями Кт Капитал - отражаем выручку

//Если в документе указана валюта, это повлияет на дальнейший

ход выполнения программы

Если Валюта = Справочники.Валюты.ПустаяСсылка() Тогда

//Валюта не задана, формируем движения в рублях, по счету расчетов в рублях

Движение=Движения.Хозрасчетный.Добавить();

Движение.Период=Дата;

Движение.СчетДт=ПланыСчетов.Хозрасчетный.РасчетыСПокупателямиВРублях;

Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Контрагенты]

=Покупатель;

Движение.СчетКт=ПланыСчетов.Хозрасчетный.Капитал;

Движение.Сумма=ВыборкаДЗ.Выручка;

Иначе

//Валюта задана, формируем движения в валюте, с пересчетом курса

Движение=Движения.Хозрасчетный.Добавить();

Движение.Период=Дата;

Движение.СчетДт=ПланыСчетов.Хозрасчетный.РасчетыСПокупателямиВВалюте;

Движение.СубконтоДт[ПланыВидовХарактеристик.ВидыСубконто.Контрагенты]

=Покупатель;

Движение.ВалютаДт=Валюта;

Движение.ВалютнаяСуммаДт=ВыборкаДЗ.Выручка;

Движение.СчетКт=ПланыСчетов.Хозрасчетный.Капитал;

Движение.Сумма=ВыборкаДЗ.Выручка\*КурсВалюты;

КонецЕсли;

КонецЦикла;

КонецПроцедуры

Прокомментируем этот код. Начнем с второстепенной функции ПолучитьКурсВалюты(). *Функция* принимает на входе *параметр* Отказ – если курс валюты не обнаружен, Отказ устанавливается в *значение* Истина, документ не проводится. Если курс не обнаружен, это не значит, что в документе нет других ошибок, поэтому данный факт не останавливает работу процедур, ответственных за *проведение документа* – при нахождении других ошибок *пользователь* будет оповещен о них.

В модифицированной процедуре ОбработкаПроведения() мы объявляем новую переменную – КурсВалюты. В эту переменную будет записано *значение* курса валюты, возвращенное функцией ПолучитьКурсВалюты() в том случае, если в документе будет установлен курс валюты. Если валюта не выбрана – он принимается равным 1 – это позволит процедуре проведения документа *по* регистрам накопления отработать правильно.

Далее, мы вызываем в процедуре ОбработкаПроведения() процедуру ОбработкаПроведенияПоБухучету(), передавая ей *параметр* Отказ. Установка его в *значение* Истина в данной процедуре приведет к отмене проведения документа.

В сравнении с исходным состоянием оставшегося кода процедуры проведения *по* регистрам накопления (ОбработкаПроведения()), мы модифицировали строку, задающую *значение* выручки в оборотном регистре **Продажи**. Теперь строка записи сведений о выручке выглядит так:

Движение.Выручка=ВыборкаДЗ.Выручка\*КурсВалюты;

Если валюта в документе не задана, КурсВалюты будет равным 1 и не повлияет на *значение*, записываемое в *регистр*, если задана, что подразумевает указание данных в документе в указанной валюте, и курс найден – в *ресурс* **Выручка** регистра **Продажи** будет записана верная сумма. Если же, ранее, при вызове функции ПолучитьКурсВалюты() окажется, что курса нужной валюты нет в регистре сведений **КурсыВалют** – *исполнение* программы просто не дойдет до данной строчки, так как перед блоком формирования движений установлена проверка на *значение* параметра Отказ, поэтому неопределенное *значение* переменной КурсВалюты не повлияет на *исполнение* программы.

Когда *исполнение* программы переходит в процедуру ОбработкаПроведенияПоБухучету(), выполняется *запрос*, который мы сейчас рассмотрим.

В первую *очередь* в данном запросе мы получаем содержимое табличной части проводимого документа в составе полей **Номенклатура, Количество и Выручка**, сгруппированное *по* полю **Номенклатура**, и помещаем во временную таблицу **ДокМТч**.

Следующим шагом будет получение данных из таблицы остатков *регистра бухгалтерии* Хозрасчетный. Мы получаем следующие поля:

**Субконто1** – в данном случае это *субконто* хранит ссылку на элемент справочника **Номенклатура**.

**КоличествоОстаток** – это *поле* получает *значение* количественного остатка материалов нужного вида в целом *по* счету **ПрочиеМатериалы**, без разделения его на отдельных ответственных лиц.

**СуммаОстаток** – здесь мы получаем *значение* суммового остатка так же в целом *по* номенклатурным позициям без разделения на отдельных ответственных лиц.

**КолисчествоЗаОтветственным** – при формировании этого поля мы применили следующую конструкцию:

| СУММА(ВЫБОР

| КОГДА ХозрасчетныйОстатки.Субконто2 = &Ответственный

| ТОГДА ХозрасчетныйОстатки.КоличествоОстаток

| КОНЕЦ) КАК КоличествоЗаОтветственным

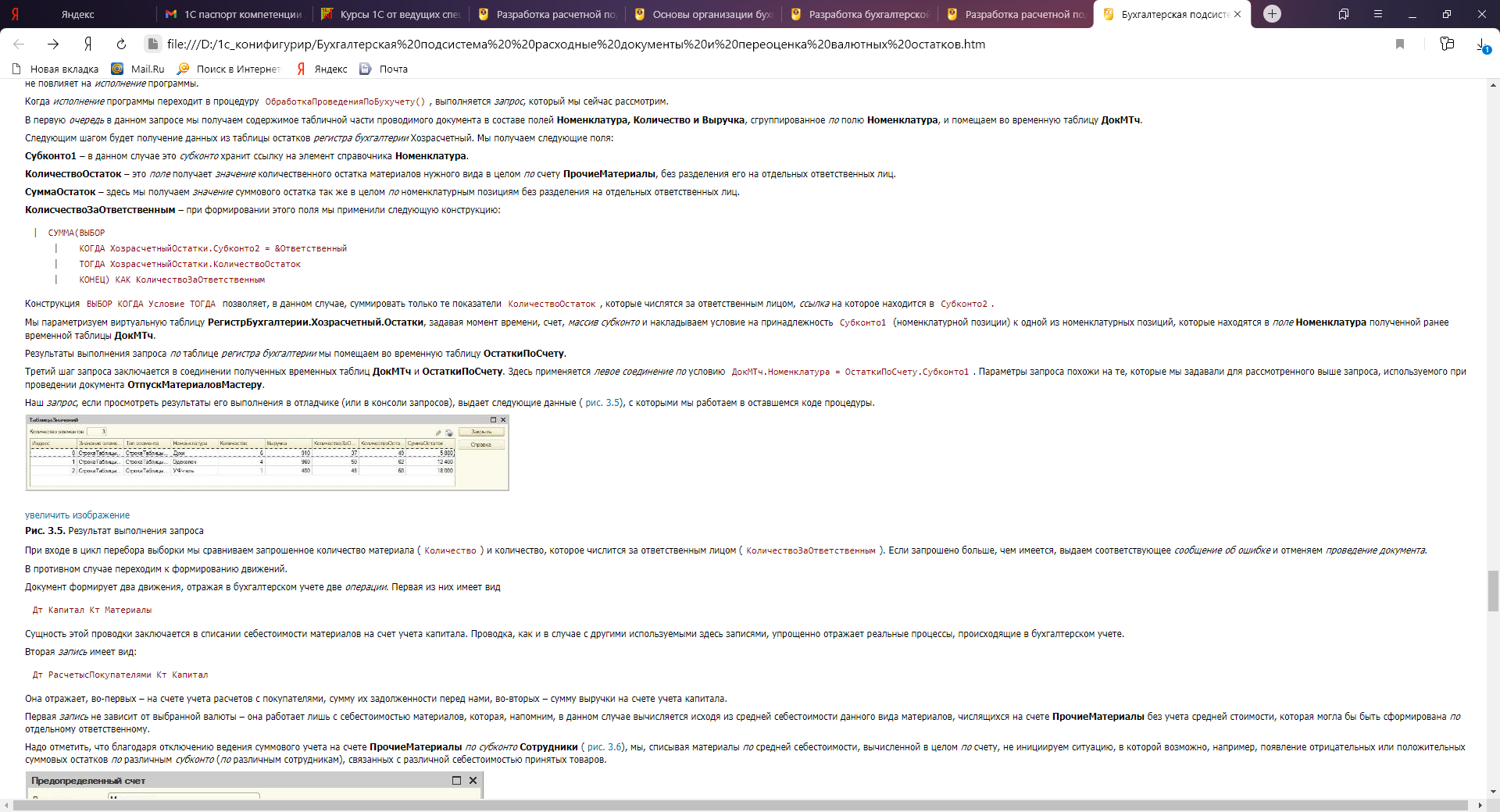
Конструкция ВЫБОР КОГДА Условие ТОГДА позволяет, в данном случае, суммировать только те показатели КоличествоОстаток, которые числятся за ответственным лицом, *ссылка* на которое находится в Субконто2.

Мы параметризуем виртуальную таблицу **РегистрБухгалтерии.Хозрасчетный.Остатки**, задавая момент времени, счет, *массив* *субконто* и накладываем условие на принадлежность Субконто1 (номенклатурной позиции) к одной из номенклатурных позиций, которые находятся в *поле* **Номенклатура** полученной ранее временной таблицы **ДокМТч**.

Результаты выполнения запроса *по* таблице *регистра бухгалтерии* мы помещаем во временную таблицу **ОстаткиПоСчету**.

Третий шаг запроса заключается в соединении полученных временных таблиц **ДокМТч** и **ОстаткиПоСчету**. Здесь применяется *левое соединение* *по* условию ДокМТч.Номенклатура = ОстаткиПоСчету.Субконто1. Параметры запроса похожи на те, которые мы задавали для рассмотренного выше запроса, используемого при проведении документа **ОтпускМатериаловМастеру**.

Наш *запрос*, если просмотреть результаты его выполнения в отладчике (или в консоли запросов), выдает следующие данные ([рис. 3.5](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=4#image.3.5)), с которыми мы работаем в оставшемся коде процедуры.

  
**Рис. 3.5.**Результат выполнения запроса

При входе в цикл перебора выборки мы сравниваем запрошенное количество материала (Количество) и количество, которое числится за ответственным лицом (КоличествоЗаОтветственным). Если запрошено больше, чем имеется, выдаем соответствующее *сообщение об ошибке* и отменяем *проведение документа*.

В противном случае переходим к формированию движений.

Документ формирует два движения, отражая в бухгалтерском учете две *операции*. Первая из них имеет вид

Дт Капитал Кт Материалы

Сущность этой проводки заключается в списании себестоимости материалов на счет учета капитала. Проводка, как и в случае с другими используемыми здесь записями, упрощенно отражает реальные процессы, происходящие в бухгалтерском учете.

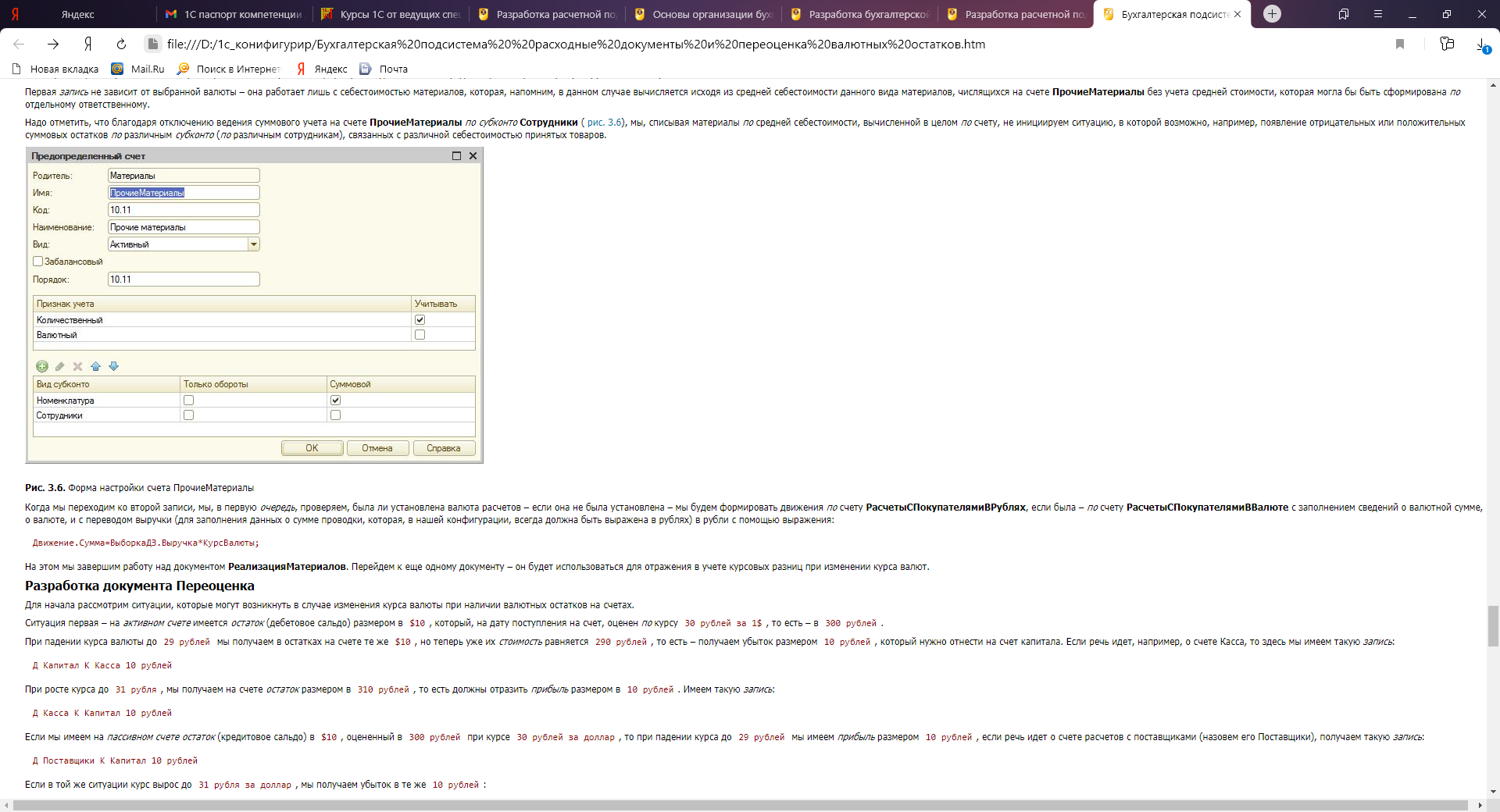
Вторая *запись* имеет вид:

Дт РасчетысПокупателями Кт Капитал

Она отражает, во-первых – на счете учета расчетов с покупателями, сумму их задолженности перед нами, во-вторых – сумму выручки на счете учета капитала.

Первая *запись* не зависит от выбранной валюты – она работает лишь с себестоимостью материалов, которая, напомним, в данном случае вычисляется исходя из средней себестоимости данного вида материалов, числящихся на счете **ПрочиеМатериалы** без учета средней стоимости, которая могла бы быть сформирована *по* отдельному ответственному.

Надо отметить, что благодаря отключению ведения суммового учета на счете **ПрочиеМатериалы** *по* *субконто* **Сотрудники** ([рис. 3.6](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=4#image.3.6)), мы, списывая материалы *по* средней себестоимости, вычисленной в целом *по* счету, не инициируем ситуацию, в которой возможно, например, появление отрицательных или положительных суммовых остатков *по* различным *субконто* (*по* различным сотрудникам), связанных с различной себестоимостью принятых товаров.

  
**Рис. 3.6.**Форма настройки счета ПрочиеМатериалы

Когда мы переходим ко второй записи, мы, в первую *очередь*, проверяем, была ли установлена валюта расчетов – если она не была установлена – мы будем формировать движения *по* счету **РасчетыСПокупателямиВРублях**, если была – *по* счету **РасчетыСПокупателямиВВалюте** с заполнением сведений о валютной сумме, о валюте, и с переводом выручки (для заполнения данных о сумме проводки, которая, в нашей конфигурации, всегда должна быть выражена в рублях) в рубли с помощью выражения:

Движение.Сумма=ВыборкаДЗ.Выручка\*КурсВалюты;

На этом мы завершим работу над документом **РеализацияМатериалов**. Перейдем к еще одному документу – он будет использоваться для отражения в учете курсовых разниц при изменении курса валют.

**Разработка документа Переоценка**

Для начала рассмотрим ситуации, которые могут возникнуть в случае изменения курса валюты при наличии валютных остатков на счетах.

Ситуация первая – на *активном счете* имеется *остаток* (дебетовое сальдо) размером в $10, который, на дату поступления на счет, оценен *по* курсу 30 рублей за 1$, то есть – в 300 рублей.

При падении курса валюты до 29 рублей мы получаем в остатках на счете те же $10, но теперь уже их *стоимость* равняется 290 рублей, то есть – получаем убыток размером 10 рублей, который нужно отнести на счет капитала. Если речь идет, например, о счете Касса, то здесь мы имеем такую *запись*:

Д Капитал К Касса 10 рублей

При росте курса до 31 рубля, мы получаем на счете *остаток* размером в 310 рублей, то есть должны отразить *прибыль* размером в 10 рублей. Имеем такую *запись*:

Д Касса К Капитал 10 рублей

Если мы имеем на *пассивном счете* *остаток* (кредитовое сальдо) в $10, оцененный в 300 рублей при курсе 30 рублей за доллар, то при падении курса до 29 рублей мы имеем *прибыль* размером 10 рублей, если речь идет о счете расчетов с поставщиками (назовем его Поставщики), получаем такую *запись*:

Д Поставщики К Капитал 10 рублей

Если в той же ситуации курс вырос до 31 рубля за доллар, мы получаем убыток в те же 10 рублей:

Д Капитал К Поставщики 10 рублей

Исходя из этих соображения мы и будем проектировать процедуру переоценки счетов.

Создадим новый документ, назовем его **ПереоценкаВалютныхОстатков**, добавим в подсистему **БухгалтерскийУчет**, укажем на закладке окна редактирования объекта **Движения**, что этот документ будет проводиться *по* *регистру бухгалтерии* **Хозрасчетный**. Запретим оперативное *проведение документа*. Документ не будет иметь реквизитов – мы планируем использовать в процедуре проведения лишь информацию о дате документа, эту информацию нам предоставит его стандартный *реквизит* **Дата**.

В основе процедуры проведения лежит *запрос*, который должен выполнить следующие действия:

Получить из таблицы остатков *регистра бухгалтерии* данные *по* счетам, *по* которым включено ведение валютного учета, причем, только *по* тем счетам, показатели которых, при пересчете *по* текущему курсу валют, показывают ненулевое отклонение от предыдущих показателей. Помимо данных *по* счетам в запросе нужно получить сведения об отклонении. При формировании движений мы будем анализировать это отклонение, вид счета, и принимать решение о том, *прибыль* или убыток нужно отражать на этом счете.

Мы реализовали задачи, возлагаемые на документ, с помощью следующего кода:

Процедура ОбработкаПроведения(Отказ, РежимПроведения)

Перем Прибыль;

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| ХозрасчетныйОстатки.Счет,

| ХозрасчетныйОстатки.Валюта,

| ХозрасчетныйОстатки.Субконто1,

| ХозрасчетныйОстатки.Субконто2,

| ХозрасчетныйОстатки.Субконто3,

| ХозрасчетныйОстатки.СуммаОстаток - ХозрасчетныйОстатки.ВалютнаяСуммаОстаток \* ЕСТЬNULL(КурсыВалютСрезПоследних.Курс, 1)

КАК Дельта

|ИЗ

| РегистрБухгалтерии.Хозрасчетный.Остатки(&КонецМесяца, Счет.Валютный, , ) КАК ХозрасчетныйОстатки

| ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.КурсыВалют.СрезПоследних(&КонецМесяца, ) КАК КурсыВалютСрезПоследних

| ПО ХозрасчетныйОстатки.Валюта = КурсыВалютСрезПоследних.Валюта

|ГДЕ

| ХозрасчетныйОстатки.СуммаОстаток - ХозрасчетныйОстатки.ВалютнаяСуммаОстаток \*

ЕСТЬNULL(КурсыВалютСрезПоследних.Курс, 1) <> 0";

Запрос.УстановитьПараметр("КонецМесяца", КонецМесяца(Дата));

Результат = Запрос.Выполнить();

ВыборкаДЗ = Результат.Выбрать();

Движения.Хозрасчетный.Записывать=Истина;

Пока ВыборкаДЗ.Следующий() Цикл

Движение=Движения.Хозрасчетный.Добавить();

//Проверяем, прибыль или убыток будем отражать

//Блок №1

Если (ВыборкаДЗ.Дельта<0 И ВыборкаДЗ.Счет.Вид=ВидСчета.Активный) Тогда

Прибыль=Истина;

КонецЕсли;

//Блок №2

Если (ВыборкаДЗ.Дельта>0 И ВыборкаДЗ.Счет.Вид=ВидСчета.Активный) Тогда

Прибыль=Ложь;

КонецЕсли;

//Блок №3

Если (ВыборкаДЗ.Дельта<0 И ВыборкаДЗ.Счет.Вид=ВидСчета.Пассивный) Тогда

Прибыль=Истина;

КонецЕсли;

//Блок №4

Если (ВыборкаДЗ.Дельта>0 и ВыборкаДЗ.Счет.Вид=ВидСчета.Пассивный) Тогда

Прибыль=Ложь;

КонецЕсли;

//Отражаем прибыль или убыток

Если Прибыль Тогда

//Отражаем прибыль

Движение.СчетДт=ВыборкаДЗ.Счет;

Движение.СчетКт=ПланыСчетов.Хозрасчетный.Капитал;

Движение.ВалютаДт=ВыборкаДЗ.Валюта;

//Для правильного заполнения субконто перебираем полученные субконто

//ориентируясь на количество видов субконто счета

Для Каждого Запись Из ВыборкаДЗ.Счет.ВидыСубконто Цикл

Движение.СубконтоДт[Запись.ВидСубконто]=ВыборкаДЗ["Субконто"

+Запись.НомерСтроки];

КонецЦикла;

Иначе

//Отражаем убыток

Движение.СчетДТ=ПланыСчетов.Хозрасчетный.Капитал;

Движение.СчетКТ=ВыборкаДЗ.Счет;

Движение.ВалютаКТ=ВыборкаДЗ.Валюта;

Для Каждого Запись Из ВыборкаДЗ.Счет.ВидыСубконто Цикл

Движение.СубконтоКт[Запись.ВидСубконто]=ВыборкаДЗ["Субконто"

+Запись.НомерСтроки];

КонецЦикла;

КонецЕсли;

//Если Дельта больше нуля, отразим без изменений

//Если Дельта меньше нуля, изменим знак

Движение.Сумма=?(ВыборкаДЗ.Дельта>0,ВыборкаДЗ.Дельта,-ВыборкаДЗ.Дельта);

Движение.Период=КонецМесяца(Дата);

КонецЦикла;

КонецПроцедуры

В начале процедуры мы объявляем переменную Прибыль. Эту переменную мы будем использовать как флаг для определения того, *прибыль* или убыток нужно отражать при переоценке.

В запросе мы получаем из таблицы **Остатки** *регистра бухгалтерии* остатки *по* счетам, у которых включен признак Валютный. Кроме того, мы получаем из регистра сведений **Валюты** последние, на заданную дату (&КонецМесяца) данные, ориентируясь на те валюты (*выражение* ПО), которые имеются у счетов, полученных из *регистра бухгалтерии*.

*Параметр* &КонецМесяца мы устанавливаем, приводя к концу месяца дату документа.

Мы получаем не все данные, а лишь те (*выражение* ГДЕ), у которых различаются сумма, хранящаяся в регистре с рублевой оценкой валютной суммы, хранящейся в регистре.

Собственно в результаты запроса попадают следующие сведения:

*Ссылка* на счет, на валюту, три *субконто* (в нашей конфигурации их количество ограничено тремя, заранее мы не знаем, сколько *субконто* будет у найденных счетов, поэтому выбираем все три *субконто*), а так же показатель разницы в оценке, на котором мы остановимся подробнее. Его получение выглядит так:

ХозрасчетныйОстатки.СуммаОстаток - ХозрасчетныйОстатки.ВалютнаяСуммаОстаток \* ЕСТЬNULL(КурсыВалютСрезПоследних.Курс, 1)

КАК Дельта

То есть, здесь мы отнимаем от суммового остатка *по* счету (выраженного в рублях и рассчитанного *по* некоему курсу валюты в прошлом) результат произведения остатка в валюте на текущий курс валюты. В том случае, если курс валюты будет равняться NULL, мы приводим его к 1. Результат этого выражения мы называем Дельта. Самое главное в последующей интерпретации этого выражения – понимать внутреннюю структуру таблицы остатков *регистра бухгалтерии*. На [рис. 3.7](http://www.intuit.ru/studies/courses/2321/621/lecture/13519?page=5#image.3.7). изображена *таблица*, полученная запросом с использованием консоли запросов. Мы использовали такой *запрос*:

ВЫБРАТЬ

Счет,

Счет.Вид,

Валюта,

СуммаОстаток,

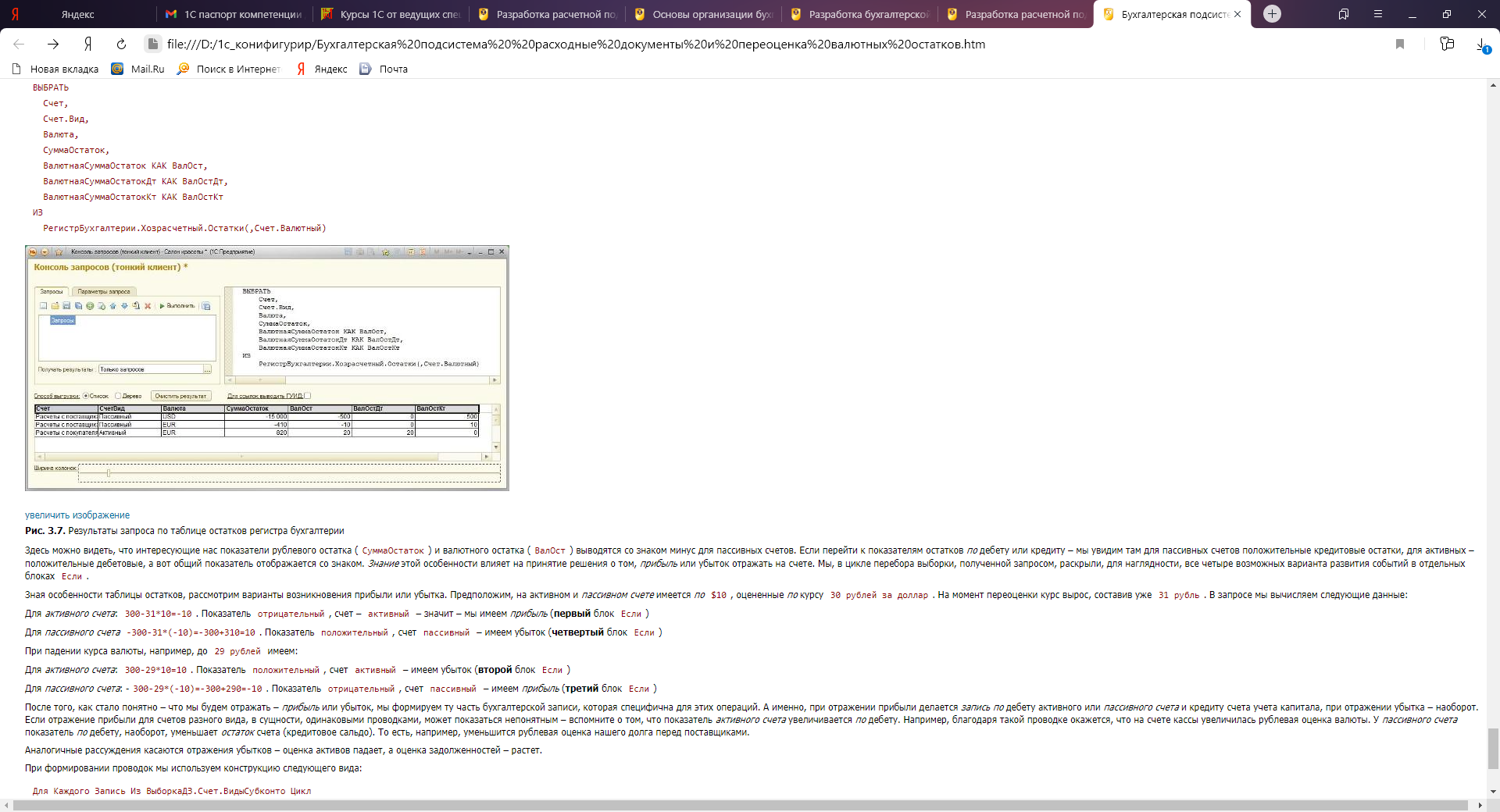
ВалютнаяСуммаОстаток КАК ВалОст,

ВалютнаяСуммаОстатокДт КАК ВалОстДт,

ВалютнаяСуммаОстатокКт КАК ВалОстКт

ИЗ

РегистрБухгалтерии.Хозрасчетный.Остатки(,Счет.Валютный)

  
**Рис. 3.7.**Результаты запроса по таблице остатков регистра бухгалтерии

Здесь можно видеть, что интересующие нас показатели рублевого остатка (СуммаОстаток) и валютного остатка (ВалОст) выводятся со знаком минус для пассивных счетов. Если перейти к показателям остатков *по* дебету или кредиту – мы увидим там для пассивных счетов положительные кредитовые остатки, для активных – положительные дебетовые, а вот общий показатель отображается со знаком. *Знание* этой особенности влияет на принятие решения о том, *прибыль* или убыток отражать на счете. Мы, в цикле перебора выборки, полученной запросом, раскрыли, для наглядности, все четыре возможных варианта развития событий в отдельных блоках Если.

Зная особенности таблицы остатков, рассмотрим варианты возникновения прибыли или убытка. Предположим, на активном и *пассивном счете* имеется *по* $10, оцененные *по* курсу 30 рублей за доллар. На момент переоценки курс вырос, составив уже 31 рубль. В запросе мы вычисляем следующие данные:

Для *активного счета*: 300-31\*10=-10. Показатель отрицательный, счет – активный – значит – мы имеем *прибыль* (**первый** блок Если)

Для *пассивного счета* -300-31\*(-10)=-300+310=10. Показатель положительный, счет пассивный – имеем убыток (**четвертый** блок Если)

При падении курса валюты, например, до 29 рублей имеем:

Для *активного счета*: 300-29\*10=10. Показатель положительный, счет активный – имеем убыток (**второй** блок Если)

Для *пассивного счета*: -300-29\*(-10)=-300+290=-10. Показатель отрицательный, счет пассивный – имеем *прибыль* (**третий** блок Если)

После того, как стало понятно – что мы будем отражать – *прибыль* или убыток, мы формируем ту часть бухгалтерской записи, которая специфична для этих операций. А именно, при отражении прибыли делается *запись* *по* дебету активного или *пассивного счета* и кредиту счета учета капитала, при отражении убытка – наоборот. Если отражение прибыли для счетов разного вида, в сущности, одинаковыми проводками, может показаться непонятным – вспомните о том, что показатель *активного счета* увеличивается *по* дебету. Например, благодаря такой проводке окажется, что на счете кассы увеличилась рублевая оценка валюты. У *пассивного счета* показатель *по* дебету, наоборот, уменьшает *остаток* счета (кредитовое сальдо). То есть, например, уменьшится рублевая оценка нашего долга перед поставщиками.

Аналогичные рассуждения касаются отражения убытков – оценка активов падает, а оценка задолженностей – растет.

При формировании проводок мы используем конструкцию следующего вида:

Для Каждого Запись Из ВыборкаДЗ.Счет.ВидыСубконто Цикл

Движение.СубконтоКт[Запись.ВидСубконто]=ВыборкаДЗ["Субконто"+Запись.НомерСтроки];

КонецЦикла;

Здесь мы перебираем виды *субконто*, относящиеся к счету, с которым мы в данный момент работаем, и "переписываем" в соответствующие *субконто* проводки те данные, которые были получены в запросе. В нашей конфигурации предусмотрено три *субконто* – именно столько мы получаем из запроса, а обход коллекции видов *субконто* для счета зависит от их количества на конкретном счете.

Завершается формирование проводки следующей операцией:

Движение.Сумма=?(ВыборкаДЗ.Дельта>0,ВыборкаДЗ.Дельта,-ВыборкаДЗ.Дельта);

Эту *запись* следует читать так: **Если** *показатель* **Дельта** *больше* 0, *присвоить его* **Сумме**, *если меньше – тоже присвоить, но поменять знак*. Свойство **Сумма** должно содержать положительные значения, а наш показатель может быть как положительным, так и отрицательным.

**Выводы**

Данная работа посвящена вопросам проведения *по* подсистеме бухгалтерского учета документов, фиксирующих списание материальных ценностей, а так же – разработке документа, предназначенного для отражения в учете результатов переоценки валютных остатков на счетах.

В нашей же следующей работе мы займемся решением задачи расчета заработной платы на основе расчетных механизмов 1С:Предприятие 8.2.