

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
Тверской государственный технический университет
(ТвГТУ)

УТВЕРЖДАЮ

Проректор по учебно-воспитательной работе
_____ А. В. Твардовский
«____» _____ 20__ г.

УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

дисциплины регионального компонента «Объектно-ориентированное
программирование экономических информационных систем» для студентов,
обучающихся по специальности 230105 — «Программное обеспечение
вычислительной техники и автоматизированных систем»

Факультет информационных технологий

Кафедра «Программное обеспечение»

Курс 5, семестр 9 (17 недель)

Лекции — 51 час.

Лабораторный практикум — 34 часа.

Самостоятельная работа — 17 часов.

Экзамен — 9 семестр.

Всего — 102 часа.

Тверь, 2012.

Лабораторная работа №1

Учёт движения денежных средств в кассах предприятия (использование регистров накопления в прикладных решениях)

Цель работы

Изучение использования регистров накопления. Применение регистров остатков и регистров оборотов. Создание запросов и отчётов на основе данных регистра.

Выполнение работы

1. Создайте пустой каталог «Lab_1». Создайте пустую информационную базу «Лабораторная работа №1» в этом каталоге.
2. Создайте справочник «Валюты»: строковый код — 3 символа, наименование — 50 символов.
3. Создайте справочник «Кассы»: числовой код — 5 знаков, наименование — 50 символов.
4. Создайте регистр накопления «Остатки денежных средств»:
 - вид регистра — «Остатки»;
 - измерение «Касса» (из справочника);
 - измерение «Валюта» (из справочника);
 - ресурс «Сумма» типа **Число** (длина — 15, точность — 2).
5. Создайте регистр накопления «Обороты денежных средств»:
 - вид регистра — «Обороты»;
 - измерение «Валюта» (из справочника);
 - ресурс «Сумма» типа **Число** (длина — 15, точность — 2).
6. Создайте документ «Поступление денежных средств»:
 - реквизит «Касса» (из справочника);
 - табличная часть «Состав»:
 - реквизит табличной части «Валюта» (из справочника);
 - реквизит табличной части «Сумма» (тип — **Число**, длина — 15, точность — 2, неотрицательный);
 - документ делает движения по регистру «Остатки денежных средств»: с помощью «Конструктора движений» создать обработчик проведения документа: данные для полей регистра берутся из соответствующих полей табличной части «Состав», вид движения — «Приход»;
 - документ делает движения по регистру «Обороты денежных средств»: с помощью «Конструктора движений» дополнить ранее созданный обработчик проведения документа: данные для полей регистра берутся из соответствующих полей

табличной части «Состав»;

- вручную измените созданный обработчик (самостоятельно): документ не должен проводиться, если табличная часть пуста или содержит строки с незаполненными реквизитами «Валюта» и «Сумма», а также если не указана касса;
 - с помощью «Конструктора печати» создайте возможность вывода печатной формы документа;
 - создайте страницу справочной информации для документа. На ней разместите краткое описание документа в произвольной форме.
7. Создайте документ «Расходование денежных средств», полностью аналогичный документу «Поступление денежных средств» (вид движений по регистру «Остатки денежных средств» — «Расход»).
 8. С помощью схемы компоновки данных создайте отчёт «Денежные средства в кассах» на основе данных регистра остатков, отражающий в разрезе касс и валют следующие суммовые показатели:
 - начальный остаток;
 - приход;
 - расход;
 - конечный остаток.
 9. С помощью схемы компоновки данных создайте отчёт «Оборот денежных средств» на основе данных регистра оборотов, отражающий суммы оборота в разрезе валют.
 10. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Что такое регистр накопления? Что такое измерение, ресурс, реквизит?
2. За счёт чего регистры накопления обеспечивают повышение производительности алгоритмов учёта?
3. Каковы отличия регистра накопления от регистра сведений?
4. Какие виртуальные таблицы могут формировать регистры накопления?
5. Можно ли использовать регистр накопления для учёта цен товаров (цена — ресурс)?

Лабораторная работа №2

Учёт товаров по методу средней стоимости

Цель работы

Изучение использования регистров накопления для практической задачи учёта. Использование запросов во встроенном языке.

Выполнение работы

1. Создайте пустой каталог «Lab_2». Создайте пустую информационную базу «Лабораторная работа №2» в этом каталоге.
2. Создайте следующие объекты конфигурации:
 - справочник «Номенклатура»;
 - справочник «Склады»;
 - регистр накопления «Остатки номенклатуры»:
 - вид регистра — «Остатки»;
 - измерения «Склад», «Номенклатура»;
 - ресурсы «Количество», «Сумма»;
 - документ «Приходная накладная»:
 - реквизит «Склад»;
 - табличная часть «Товары»:
 - реквизиты табличной части «Номенклатура», «Количество», «Сумма»;
 - документ делает движения по регистру в приход (сделать с помощью конструктора движений);
 - документ «Расходная накладная»:
 - реквизит «Склад»;
 - табличная часть «Товары»:
 - реквизиты табличной части «Номенклатура», «Количество»;
 - документ делает движения по регистру в расход (сделать с помощью конструктора движений);
 - отчёт «Остатки товаров на складах»:
 - использует схему компоновки данных;
 - в качестве источника данных использует запрос по виртуальной таблице регистра «Остатки»;
 - отображает в разрезе складов и номенклатуры значения двух ресурсов:
 - остаток по количеству;
 - остаток по сумме.
3. Проверить работу созданной конфигурации. Проведения документов прихода должны

работать корректно. Проведения документов расхода должны корректно работать по количеству номенклатуры, но не влиять на сумму.

4. Открыть модуль документа «Расходная накладная», удалить из него тело процедуры **ОбработкаПроведения()**.

5. В теле процедуры **ОбработкаПроведения()**:

- создать объект — запрос **ЗапросОстатков**;
- назначить запросу текст, создав его конструктором запроса (или скопировать из приложения А),:
 - выбраны таблицы:
 - виртуальная таблица остатков регистра;
 - вложенный запрос (создать его и назвать «ТаблицаСписания»);
 - настроить вложенный запрос «ТаблицаСписания»:
 - выбрать таблицу «Товары» у расходных накладных;
 - из таблицы выбрать поля «Номенклатура» и «Количество»;
 - настроить группировку по полю «Номенклатура», а суммируемую функцию — по полю «Количество»;
 - указать условие — «**РасходнаяНакладнаяТовары.Ссылка = &ЭтотДокумент**» (без кавычек);
 - закрыть вложенный запрос;
 - назначить виртуальной таблице остатков регистра следующие параметры (без кавычек):
 - период — «**&МоментВремени**» (параметр);
 - условие — «**Склад = &Склад И Номенклатура В (&Временное)**»;
 - связь между таблицами — по равенству значений полей «Номенклатура», для вложенного запроса «ТаблицаСписания» поставить флажок «Все»;
 - из таблиц выбрать поля:
 - из таблицы регистра — «Номенклатура», «Склад», «КоличествоОстаток», «СуммаОстаток»;
 - из вложенного запроса «ТаблицаСписания» — «Количество»;
 - снабдить поля «КоличествоОстаток» и «СуммаОстаток» функцией **ЕСТЬNULL** для обработки нулевых остатков (см. приложение А);
 - назначить псевдонимы полей:
 - «Количество» — «КоличествоСписывается»;
 - «КоличествоОстаток» — «КоличествоОстаток»;
 - «СуммаОстаток» — «СуммаОстатка»;
 - закрыть конструктор запроса кнопкой «ОК» (основная часть запроса готова);
 - в любом месте модуля поставить две кавычки;
 - поставить курсор между кавычками, из контекстного меню выбрать

- конструктор запроса;
- в конструкторе запроса:
 - выбрать таблицу «Товары» у расходных накладных;
 - из таблицы выбрать поле «Номенклатура»;
 - указать условие — «**РасходнаяНакладнаяТовары.Ссылка = &ЭтотДокумент**» (без кавычек);
 - на закладке «Дополнительно» поставить флажок «Без повторяющихся»;
 - нажать на кнопку «Запрос» в левом нижнем углу, скопировать появившийся текст запроса в буфер обмена;
 - закрыть конструктор запроса кнопкой «Отмена», удалить поставленные кавычки из модуля (дополнительный запрос скопирован в буфер обмена);
 - поставить курсор в любое место ранее созданного запроса, вызвать конструктор запроса;
 - в параметрах виртуальной таблицы регистра (условие) найти параметр **&Временное**, удалить его, вставить на его место запрос из буфера обмена;
 - закрыть конструктор запросов кнопкой «ОК» (запрос полностью готов);
 - после текста запроса написать в модуле код, обрабатывающий данные запроса (или скопировать из приложения В):
 - установка параметров запроса (**&ЭтотДокумент, &МоментВремени, &Склад**);
 - выполнение и выборка результата запроса;
 - обход результата запроса:
 - если списывается нулевое количество товара, то вывести сообщение и не проводить документ;
 - если списывается не больше товара, чем имеется на складе, то сформировать движение документа по регистру;
 - если списывается больше товара, чем имеется на складе, то вывести сообщение и не проводить документ.
6. При копировании текста запроса и кода из приложений постарайтесь понять назначение каждого шага в их создании.
7. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Какие бывают методы учёта стоимости?
2. В чём различие методов ЛИФО и ФИФО?
3. Каковы сравнительные достоинства и недостатки методов списания по средней стоимости и ФИФО?
4. С какой целью в данной работе используется вложенный запрос?
5. С какой целью в данной работе используется дополнительный запрос в параметрах

виртуальной таблицы?

Лабораторная работа №3

Партионный учёт. Методы ФИФО, ЛИФО

Цель работы

Изучение использования регистров накопления для решения задачи партионного учёта. Использование запросов во встроенном языке.

Выполнение работы

В качестве исходной конфигурации к работе №3 используется конфигурация, полученная при выполнении работы №2.

1. Откройте конфигурацию, полученную при выполнении работы №2. Выгрузите конфигурацию в файл.
2. Создайте пустой каталог «Lab_3». Создайте пустую информационную базу «Лабораторная работа №3» в этом каталоге. Загрузите в неё конфигурацию из файла. Убедитесь в работоспособности загруженной конфигурации.
3. В регистр накопления «Остатки номенклатуры» добавьте измерение «Партия». Тип значения измерения — ссылка на документ «Приходная накладная».
4. Измените процедуру проведения документа «Приходная накладная» так, чтобы в его движениях в поле «Партия» попадала ссылка на данный документ. Для этой цели можно либо использовать конструктор движений, либо добавить в уже созданную процедуру проведения (в тело цикла) следующую строку:

Движение.Партия = Ссылка ;

5. Измените схему компоновки данных отчёта:
 - в запросе добавьте к выбранным полям измерение «Партия»;
 - на закладке «Ресурсы» обеспечьте расчёт итогов по партиям для всех ресурсов;
 - на закладки «Настройки» добавьте группировку «Партия», подчинённую группировке «Номенклатура».
6. Проверьте работу созданной конфигурации. Проведения документов прихода должны работать корректно:
 - проведение приходных накладных должно обеспечивать заполнение измерения «Партия» в регистре;
 - в отчёте должны быть отражены остатки товаров на складах по каждой партии (документу поступления).
7. Измените процедуру проведения документа «Расходная накладная»:
 - изменить текст запроса с помощью конструктора запроса (или скопировать из приложения С):
 - из таблицы регистра добавить измерение «Партия»;
 - на закладке «Порядок» указать упорядочивание по полю «Партия» (по возрастанию);

- на закладке «Итоги» указать группировку по полю «Номенклатура», поля «КоличествоОстатка» и «СуммаОстатка» сделать итоговыми с функцией «СУММА», поле «КоличествоСписывается» сделать итоговым с функцией «МАКСИМУМ», «МИНИМУМ» или «СРЕДНЕЕ».
- после текста запроса написать в модуле код, обрабатывающий данные запроса (или скопировать из приложения D):
 - при получении выборки результата запроса указать, что обход результата будет осуществляться по группировкам;
 - для случая, если требуемого количества товара на складе достаточно для списания:
 - запомнить в переменной количество товара, подлежащего списыванию;
 - из выборки остатков осуществить дочернюю выборку;
 - создать цикл, выполняющийся, пока количество товара, подлежащего списыванию, больше нуля;
 - в теле цикла:
 - получить следующий элемент выборки;
 - запомнить в переменной, сколько товара следует списать из данной партии (определяется как меньшее из двух: количество товара, подлежащего списыванию, и количество товара в данной партии);
 - создать движение по регистрам, заполнить его поля, в частности:
 - количество — определённое в предыдущем пункте;
 - сумма — пропорционально списываемому количеству, исходя из стоимости остатка по данной партии;
 - уменьшить значение переменной, хранящей количество товара, подлежащего списыванию, на количество товара, списанной из данной партии.
- 8. При копировании текста запроса и кода из приложений постарайтесь понять назначение каждого шага в их создании.
- 9. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Какой метод списания стоимости реализован в данной конфигурации? Что нужно изменить в процедуре проведения расходных накладных для применения другого метода списания?
2. С какой целью в запросе используются итоги?
3. С какой целью в запросе для поля «КоличествоСписывается» используется произвольная из трёх функций? Какое значение имеет выбор той или иной функции из указанных?

Лабораторная работа №4

Элементарный бухгалтерский учёт (использование планов счетов и регистров накопления в прикладных решениях)

Цель работы

Изучение использования планов счетов и регистров накопления. Применение объектов конфигурации для синтетического учёта на счетах. Создание запросов и отчётов на основе данных регистра.

Выполнение работы

1. Создайте пустой каталог «Lab_4». Создайте пустую информационную базу «Лабораторная работа №4» в этом каталоге.
2. Создайте план счетов «Основной»:
 - длина кода — 5;
 - длина наименования — 50;
 - маска кода — «@@@.@» (без кавычек);
 - автопорядок по коду;
 - длина порядка — 5;
 - основное представление — в виде наименования;
 - добавьте признак учёта — «Количественный»;
 - откройте список predetermined счетов (вкладка «Прочее» в окне плана счетов) и добавьте следующие predetermined счета:
 - «Вспомогательный», код «001.», порядок «001», активный/пассивный;
 - «Товары», код «41 .», порядок «41», активный, с учётом по признаку «Количественный»;
 - «Касса», код «50 .», порядок «50», активный;
 - убедитесь, что все счета созданы на одном уровне (ни один из созданных счетов не подчинён другому); если это не так, то исправьте расположение счетов с помощью кнопки «Переместить элемент в другую группу» на панели инструментов окна.
3. Создайте регистр бухгалтерии «Основной»:
 - связан с планом счетов «Основной»;
 - включена корреспонденция;
 - имеет ресурс «Сумма», балансовый;
 - имеет ресурс «Количество», небалансовый, связан с признаком учёта «Количественный», не связан с признаками учёта по субконто.
4. Создайте документ «Ввод остатка по кассе»:

- имеет реквизит «Сумма»;
 - документ делает движения по регистру бухгалтерии;
 - с помощью конструктора движений создайте движения со следующими параметрами:
 - счёт дебета — 50;
 - счёт кредита — 001;
 - поле «Сумма» соответствует реквизиту «Сумма».
5. Создайте справочник «Номенклатура».
6. Создайте документ «Покупка товаров»:
- имеет табличную часть «Состав»:
 - реквизит табличной части «Номенклатура»;
 - реквизит табличной части «Количество»;
 - реквизит табличной части «Сумма»;
 - документ делает движения по регистру бухгалтерии;
 - с помощью конструктора движений создайте движения со следующими параметрами:
 - счёт дебета — 41;
 - счёт кредита — 50;
 - табличная часть — «Состав»;
 - поля «Количество» и «Сумма» соответствуют одноимённым реквизитам документа.
7. Создайте отчёт «Сальдовая ведомость», использующий схему компоновки данных:
- добавьте набор данных — запрос;
 - в конструкторе запроса:
 - выберите виртуальную таблицу регистра «Остатки»;
 - из таблицы выберите следующие поля:
 - Счёт;
 - СуммаРазвёрнутыйОстатокДт (назначьте псевдоним СуммаДт);
 - КоличествоРазвёрнутыйОстатокДт (назначьте псевдоним КоличествоДт);
 - СуммаРазвёрнутыйОстатокКт (назначьте псевдоним СуммаКт);
 - КоличествоРазвёрнутыйОстатокКт (назначьте псевдоним КоличествоКт);
 - на вкладке «Ресурсы» выбрать все поля, кроме поля «Счёт», и назначить им расчёт общих итогов и итогов по счёту;
 - на вкладке «Настройки»:
 - создать группировку «Счёт»;
 - внизу на вкладке «Выбранные поля» выбрать поля:

- СуммаДт;
 - КоличествоДт;
 - СуммаКт;
 - КоличествоКт.
8. Создайте отчёт «Оборотная ведомость», использующий схему компоновки данных:
- добавьте набор данных — запрос;
 - в конструкторе запроса:
 - выберите виртуальную таблицу регистра «Обороты»;
 - из таблицы выберите следующие поля:
 - Счёт;
 - КорСчёт;
 - СуммаОборотДт (назначьте псевдоним СуммаДт);
 - КоличествоОборотДт (назначьте псевдоним КоличествоДт);
 - СуммаОборотКт (назначьте псевдоним СуммаКт);
 - КоличествоОборотКт (назначьте псевдоним КоличествоКт);
 - на вкладке «Ресурсы» выбрать все поля, кроме полей «Счёт» и «КорСчёт», и назначить им расчёт общих итогов и итогов по счёту;
 - на вкладке «Настройки»:
 - создать группировку «Счёт»;
 - создать группировку «КорСчёт», подчинённую группировке «Счёт»;
 - внизу на вкладке «Выбранные поля» выбрать поля:
 - СуммаДт;
 - КоличествоДт;
 - СуммаКт;
 - КоличествоКт.
9. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Что такое бухгалтерский учёт?
2. Что такое бухгалтерский баланс? Из каких частей состоит баланс?
3. Что такое счёт? В чём заключается метод двойной записи?
4. Что такое сальдо, проводка?
5. Какие бывают виды счётов по отношению к балансу?
6. Что такое план счетов бухгалтерского учёта и каковы его свойства?
7. Что такое синтетический и аналитический учёт?

Лабораторная работа №5

Аналитический учёт по субконто (использование планов видов характеристик для бухгалтерского учёта)

Цель работы

Изучение использования планов видов характеристик для целей бухгалтерского учёта. Применение объектов конфигурации для аналитического учёта на счетах. Реализация признаков учёта и признаков учёта по субконто.

Выполнение работы

1. Создайте пустой каталог «Lab_5». Создайте пустую информационную базу «Лабораторная работа №5» в этом каталоге.
2. Создайте план счетов «Основной» со следующими predetermined счетами:
 - 001 «Вспомогательный», активный/пассивный;
 - 10 «Материалы», активный;
 - 51 «Расчётные счета», активный;
 - 60 «Расчёты с поставщиками и подрядчиками», пассивный.
3. Добавьте признак учёта — «Валютный». Установите его для всех созданных счетов.
4. Создайте справочники «Номенклатура», «Склады», «Банковские счета», «Контрагенты», «Договоры». Справочник «Договоры» должен быть подчинён справочнику «Контрагенты» (подчинение элементам).
5. Создайте план видов характеристик «Виды субконто»:
 - тип значения характеристик — ссылки на все созданные справочники;
 - создайте predetermined характеристики «Номенклатура», «Склады», «Банковские счета», «Контрагенты», «Договоры»;
 - тип значения каждой характеристики — ссылка на соответствующий справочник.
6. В плане счетов выберите созданный план видов характеристик в качестве источника субконто. Установите максимальное количество субконто — 2.
7. Добавьте признак учёта субконто — «Количественный».
8. Установите виды субконто для счетов в соответствии с таблицей:

Код	Субконто1	Субконто2
001		
10	Номенклатура	Склады
51	Банковские счета	
60	Контрагенты	Договоры

9. Для счёта 10 для субконто «Номенклатура» установите признак «Количественный»,

для всех остальных счетов и субконто снимите этот признак.

10. Создайте регистр бухгалтерии «Основной»:

- связан с планом счетов «Основной»;
- поддерживает корреспонденцию;
- имеет ресурсы:
 - «Сумма», балансовый;
 - «Количество», небалансовый, признак учёта по субконто «Количественный»;
 - «Сумма в валюте», балансовый, признак учёта «Валютный».

11. Создайте документ «Ввод остатка по банку»:

- реквизиты «Банковский счёт», «Сумма» и «Сумма в валюте»;
- документ делает движения по регистру бухгалтерии со следующими параметрами:
 - счёт дебета — 51;
 - счёт кредита — 001;
 - поле «Сумма» соответствует реквизиту «Сумма»;
 - поле «Сумма в валюте» соответствует реквизиту «Сумма в валюте»;
 - субконто счёта дебета («Банковские счета») соответствует реквизиту «Банковский счёт»;
 - других соответствий нет.

12. Создайте документ «Поступление товаров»:

- реквизит «Поставщик» (из справочника «Контрагенты»);
- реквизит «Договор»;
- реквизит «Склад»;
- табличная часть «Состав»:
 - реквизиты табличной части «Номенклатура», «Количество», «Сумма», «Сумма в валюте»;
- документ делает движения по регистру бухгалтерии со следующими параметрами:
 - счёт дебета — 10;
 - счёт кредита — 60;
 - используется табличная часть «Состав»;
 - поле «Сумма» соответствует реквизиту табличной части «Сумма»;
 - поле «Количество» (дебет) соответствует реквизиту табличной части «Количество»;
 - поле «Сумма в валюте» соответствует реквизиту табличной части «Сумма в валюте»;
 - субконто счёта дебета («Номенклатура») соответствует реквизиту табличной части «Номенклатура»;
 - субконто счёта дебета («Склад») соответствует реквизиту «Склад»;

- субконто счёта кредита («Контрагенты») соответствует реквизиту «Поставщик»;
- субконто счёта кредита («Договоры») соответствует реквизиту «Договор»;
- других соответствий нет.

13. Создайте документ «Оплата поставщику»:

- реквизиты «Поставщик», «Договор», «Банковский счёт», «Сумма», «Сумма в валюте»;
- документ делает движения по регистру бухгалтерии со следующими параметрами:
 - счёт дебета — 60;
 - счёт кредита — 51;
 - поле «Сумма» соответствует реквизиту «Сумма»;
 - поле «Сумма в валюте» соответствует реквизиту «Сумма в валюте»;
 - субконто счёта дебета («Контрагенты») соответствует реквизиту «Поставщик»;
 - субконто счёта дебета («Договоры») соответствует реквизиту «Договор»;
 - субконто счёта кредита («Банковские счета») соответствует реквизиту «Банковский счёт»;
 - других соответствий нет.

14. Создайте отчёт «Сальдовая ведомость», использующий схему компоновки данных:

- добавьте набор данных — запрос;
- в конструкторе запроса:
 - выберите виртуальную таблицу регистра «Остатки»;
 - из таблицы выберите следующие поля:
 - Счёт;
 - Субконто1;
 - Субконто2;
 - СуммаРазвёрнутыйОстатокДт (назначьте псевдоним СуммаДт);
 - КоличествоРазвёрнутыйОстатокДт (назначьте псевдоним КоличествоДт);
 - СуммаРазвёрнутыйОстатокКт (назначьте псевдоним СуммаКт);
 - КоличествоРазвёрнутыйОстатокКт (назначьте псевдоним КоличествоКт);
 - на вкладке «Ресурсы» выбрать поля «СуммаДт», «КоличествоДт», «СуммаКт», «КоличествоКт» и назначить им расчёт общих итогов по всем разрезам;
 - на вкладке «Настройки»:
 - создать группировку «Счёт»;
 - создайте группировку «Субконто1», подчинённую группировке «Счёт»;
 - создайте группировку «Субконто2», подчинённую группировке «Субконто1»;
 - внизу на вкладке «Выбранные поля» выбрать поля:

- СуммаДт;
- КоличествоДт;
- СуммаКт;
- КоличествоКт.

15. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Что такое план счетов как объект конфигурации?
2. Для чего используется маска кода?
3. Для чего используется порядок кода?
4. Что такое субконто?
5. Для чего используются признаки учёта?
6. Для чего используются признаки учёта по субконто?

Лабораторная работа №6

Партионный учёт в бухгалтерском учёте

Цель работы

Изучение использования партионного учёта в бухгалтерском учёте.

Выполнение работы

1. Создайте пустой каталог «Lab_6». Создайте пустую информационную базу «Лабораторная работа №6» в этом каталоге.
2. Создайте следующие объекты конфигурации:
 - справочник «Контрагенты»;
 - справочник «Договоры», подчинённый справочнику «Контрагенты»;
 - справочник «Банковские счета»;
 - документ «Ввод остатка по банку» с реквизитами «Банковский счёт», «Сумма»;
 - документ «Сделка с контрагентом» с реквизитами «Контрагент», «Договор», «Сумма»;
 - документ «Погашение задолженности» с реквизитами «Контрагент», «Договор», «Банковский счёт», «Сумма».
3. Создайте план видов характеристик «Виды субконто регламентированные». Типы значения — ссылки на справочники «Контрагенты», «Договоры», «Банковские счета» и на документ «Сделка с контрагентом». Создайте predetermined виды характеристик «Контрагенты», «Договоры», «Банковские счета», «Сделки» с соответствующими типами.
4. Создайте план счетов «Регламентированный». Выберите план видов характеристик в качестве источника видов субконто. Максимальное количество субконто — 3. Создайте predetermined счета согласно таблице.

Номер	Вид	Наименование	Субконто1	Субконто2	Субконто3
000	АП	Вспомогательный			
51	А	Расчётные счета	БанковскиеСчета		
60	П	Расчёты с поставщиками и подрядчиками	Контрагенты	Договоры	Сделки

5. Создайте регистр бухгалтерии «Регламентированный», связанный с созданными планом счетов и поддерживающий корреспонденцию. В регистре создайте балансовый ресурс «Сумма».
6. Для документа «Ввод остатка по банку» создайте процедуру движения по регистру бухгалтерии со следующими параметрами:
 - счёт дебета — 51;
 - счёт кредита — 000;

- поле «Сумма» соответствует реквизиту «Сумма»;
 - субконто счёта дебета («Банковские счета») соответствует реквизиту «Банковский счёт»;
 - других соответствий нет.
7. Для документа «Сделка с контрагентом» создайте процедуру движения по регистру бухгалтерии со следующими параметрами:
- счёт дебета — 000;
 - счёт кредита — 60;
 - поле «Сумма» соответствует реквизиту «Сумма»;
 - субконто счёта кредита («Контрагенты») соответствует реквизиту «Контрагент»;
 - субконто счёта кредита («Договоры») соответствует реквизиту «Договор»;
 - субконто счёта кредита («Сделки») соответствует реквизиту «Ссылка» (в конструкторе движений такое поле не выбирается, но можно вписать непосредственно);
 - других соответствий нет.
8. Для документа «Погашение задолженности» создайте пустую процедуру движения по регистру бухгалтерии (можно открыть конструктор движений и сразу же в нём нажать «ОК»). Откройте модуль документа «Погашение задолженности».
9. В документе «Погашение задолженности» в теле процедуры **ОбработкаПроведения()**:
- создать объект — запрос **ЗапросПоСделкам**;
 - назначить запросу текст, создав его конструктором запроса (или скопировать из приложения Е):
 - выбрана виртуальная таблица остатков регистра;
 - заданы параметры виртуальной таблицы:
 - период — **&МоментВремени** (параметр);
 - условие счёта — **«Счет = &Дебиторка»**;
 - условие счёта — **«Субконто1 = &Контрагент И Субконто2 = &Договор»**;
 - из таблицы выбраны поля «Субконто1» и «СуммаОстатокКт», им назначены псевдонимы соответственно «Сделка» и «Сумма»;
 - задано упорядочивание по возрастанию значения поля «Сделка»;
 - отмечено, что у таблицы будет итоговое поле «Сумма» (с функцией суммы), а также будут рассчитаны общие итоги.
 - после текста запроса написать в модуле код, обрабатывающий данные запроса (или скопировать из приложения F):
 - установка параметров запроса:
 - **&Контрагент** — значение реквизита «Контрагент»;
 - **&Договор** — значение реквизита «Договор»;
 - **&МоментВремени** — значение метода **«МоментВремени()»**;

- **&Дебиторка** — ссылка на предопределённый счёт «Расчёты с поставщиками и подрядчиками»;
 - выполнение и выборка результата запроса (указать, что обход результата будет осуществляться по группировкам);
 - для случая, если погашаемая сумма меньше или равна сумме задолженности:
 - запомнить в переменной сумму, подлежащую погашению;
 - из выборки остатков осуществить дочернюю выборку;
 - создать цикл, выполняющийся, пока сумма, подлежащая погашению, больше нуля;
 - в теле цикла:
 - получить следующий элемент выборки;
 - запомнить в переменной, какую сумму следует погасить по данной сделке (определяется как меньшее из двух: сумма, подлежащая погашению по текущей сделке, и оставшаяся для погашения сумма);
 - создать движение по регистрам, заполнить его поля, в частности:
 - счёт дебета — ссылка на счёт «Расчёты с поставщиками и подрядчиками»;
 - счёт кредита — ссылка на счёт «Расчётные счета»;
 - сумма — сумму, которую следует погасить по текущей сделке;
 - субконто счёта дебета («Контрагенты») — значение реквизита «Контрагент»;
 - субконто счёта дебета («Договоры») — значение реквизита «Договор»;
 - субконто счёта дебета («Сделки») — поле «Сделка» из дочерней выборки;
 - субконто счёта кредита («Банковские счета») — значение реквизита «Банковский счёт»;
 - уменьшить значение переменной, хранящей сумму, подлежащую погашению, на сумму, погашенную по текущей сделке;
 - для случая, если погашаемая сумма больше суммы задолженности, вывести диагностическое сообщение и отказать в проведении документа.
10. Создайте отчёт «Сальдовая ведомость» аналогично тому, что используется в работах №4 и №5. В отчёте должны быть группировки «Счёт», «Субконто1», «Субконто2», «Субконто3», последовательно подчинённые друг другу. Выбранные поля — суммы остатка по дебету и по кредиту.
11. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации.

Вопросы для подготовки к защите

1. Что такое регистр бухгалтерии?

2. Каковы основные сходства и различия регистра накопления и регистра бухгалтерии?
3. Какой из методов применяется для закрытия сделок в данной работе?
4. Что нужно изменить в работе, чтобы применялся иной метод закрытия сделок?
5. Какие реальные и виртуальные таблицы предоставляет регистр бухгалтерии?

Лабораторная работа №7

Элементарные периодические расчёты (использование планов видов расчёта и регистров расчёта в прикладных решениях)

Цель работы

Изучение использования планов видов расчёта и регистров расчёта. Применение объектов конфигурации для реализации механизма периодических расчётов. Создание отчёта без использования схемы компоновки данных. Работа с диаграммой Ганта.

Выполнение работы

1. Создайте пустой каталог «Lab_7». Создайте пустую информационную базу «Лабораторная работа №7» в этом каталоге.
2. Создайте следующие объекты конфигурации:
 - справочник «Сотрудники»;
 - план видов расчёта «Основные начисления»:
 - использует период действия;
 - зависит от базы по периоду действия;
 - базовый план видов расчёта — «Основные начисления»;
 - содержит predetermined виды расчёта:
 - «Невыход»;
 - «Оклад» (вытесняющий — «Невыход»);
 - «Премия» (базовый — «Оклад», ведущие — «Невыход» и «Оклад», период действия является базовым периодом);
 - регистр сведений «Графики работы» (непериодический, независимый):
 - измерение «Дата» (**Дата**);
 - ресурс «Значение» (**Число**);
 - регистр расчёта «Основные начисления»:
 - связан с планом видов расчёта «Основные начисления»;
 - поддерживает период действия;
 - связан с графиком «Графики работы»;
 - значение графика — ресурс «Значение»;
 - дата графика — измерение «Дата»;
 - поддержка базового периода;
 - периодичность — месяц;
 - измерение «Сотрудник»;
 - ресурс «Сумма»;

- документ «Начисления сотрудникам»:
 - табличная часть «Состав»:
 - реквизит табличной части «Сотрудник»;
 - реквизит табличной части «Вид расчёта» (из плана видов расчёта);
 - реквизит табличной части «Начало периода» (**Дата**);
 - реквизит табличной части «Конец периода» (**Дата**);
 - реквизит табличной части «Результат» (**Число**, длина — 15, точность — 2);
 - документ делает движения по регистру расчёта, соответствия полей:

Поле	Выражение
ВидРасчета	ТекСтрокаСостав.ВидРасчёта
ПериодДействияНачало	ТекСтрокаСостав.НачалоПериода
ПериодДействияКонец	КонецДня(ТекСтрокаСостав.КонецПериода)
ПериодРегистрации	Дата
БазовыйПериодНачало	ТекСтрокаСостав.НачалоПериода
БазовыйПериодКонец	КонецДня(ТекСтрокаСостав.КонецПериода)
Сотрудник	ТекСтрокаСостав.Сотрудник
Сумма	ТекСтрокаСостав.Результат

3. Создайте отчёт «Диаграмма Ганта»:

- создайте и откройте основную форму отчёта;
- добавьте на форму диаграмму Ганта (меню «Форма» — «Вставить элемент управления...») и растяните её на всю форму, задайте ей имя **ДиаграммаГанта**;
- В модуле формы в процедуре — обработчике нажатия на кнопку «Сформировать» запишите программный код, выполняющий следующие действия:
 - создание запроса **ЗапросНачислений**;
 - задание текста запроса (текст запроса приведён в приложении G):
 - выбор виртуальной таблицы «Фактический период действия» регистра расчёта;
 - выбор из таблицы полей «Сотрудник», «Вид расчёта», «Период действия (начало)», «Период действия (конец)», «Сумма», «Регистратор»;
 - выполнение и обход результата запроса с помещением данных в диаграмму Ганта (код приведён в приложении H):
 - запрет обновления диаграммы;
 - очистка диаграммы;

- в цикле обхода выборки:
 - установка серии как вида расчёта;
 - установка точки как сотрудника;
 - добавление интервала к текущему значению (начало периода, конец периода, текст, расшифровка на документ-регистратор);
 - раскрашивание серий в зависимости от вида расчёта в свой цвет;
 - разрешение обновления диаграммы.
4. Определите период, в котором будет демонстрироваться работа конфигурации (например, предыдущий месяц). Текущий месяц брать не нужно.
5. Заполните регистр сведений «Графики работы» в соответствии с календарём на выбранный период — рабочим дням соответствует значение 1, нерабочим — 0. Например, для октября 2012 г. содержимое регистра будет выглядеть так:

Дата	Значение
01.10.2012	1
02.10.2012	1
03.10.2012	1
04.10.2012	1
05.10.2012	1
06.10.2012	0
07.10.2012	0
08.10.2012	1
...	...
30.10.2012	1
31.10.2012	1

6. Занесите в информационную базу данные, демонстрирующие работоспособность конфигурации в пределах выбранного периода:
- произведите начисления оклада, премии и невыхода нескольким сотрудникам:
 - период действия оклада — весь месяц, результат — размер оклада;
 - период действия премии — весь месяц, результат не указывается (т. к. его предполагается рассчитывать в следующей работе);
 - период действия невыхода — произвольный интервал дней, результат не указывается;
 - сформируйте отчёт «Диаграмма Ганта» и поясните выведенные интервалы фактического периода действия расчётов;
 - самостоятельно создайте предопределённый вид расчёта «Отпуск», полностью аналогичный виду расчёта «Невыход», и внесите соответствующие изменения в объекты конфигурации;

- дополните регистр «Графики работы» ещё одним периодом и создайте ситуацию, при которой необходимо сторнирование, поясните вывод диаграммы Ганта в этом случае.

Вопросы для подготовки к защите

1. Что такое вид расчёта? Что такое период расчёта?
2. От каких параметров зависит результат расчёта?
3. Какие существуют типы взаимовлияния видов расчёта?
4. Что обозначает период действия расчёта? Что обозначает фактический период действия расчёта?
5. Что такое база расчёта, базовый период?
6. Что такое базовые, вытесняющие, ведущие виды расчёта?
7. В каком случае используется сторнирование? Что такое сторно-запись?
8. Каково основное отличие диаграммы Ганта от прочих графиков и диаграмм?

Лабораторная работа №8

Периодические расчёты. Перерасчёты

Цель работы

Изучение использования регистров накопления для решения задачи партионного учёта. Использование запросов во встроенном языке.

Выполнение работы

В качестве исходной конфигурации к работе №8 используется конфигурация, полученная при выполнении работы №7.

1. Откройте конфигурацию, полученную при выполнении работы №7. Выгрузите конфигурацию в файл.
2. Создайте пустой каталог «Lab_8». Создайте пустую информационную базу «Лабораторная работа №8» в этом каталоге. Загрузите в неё конфигурацию из файла. Убедитесь в работоспособности загруженной конфигурации.
3. В регистре расчёта создайте перерасчёт с наименованием «Перерасчёт». Для него создайте измерение «Сотрудник», связанное с измерением регистра и с данными ведущих регистров (также измерение регистра).
4. Создайте отчёт «Перерасчёты», использующий схему компоновки данных:
 - добавьте набор данных — запрос;
 - в конструкторе запроса:
 - выберите таблицу перерасчётов;
 - из таблицы выберите следующие поля:
 - ОбъектПерерасчета;
 - ВидРасчета;
 - Сотрудник;
 - на вкладке «Настройки»:
 - добавить группировку детальных записей (т. е. не выбирать поле);
 - внизу на вкладке «Выбранные поля» выбрать поля:
 - ОбъектПерерасчета;
 - ВидРасчета;
 - Сотрудник.
5. Определите период, в котором будет демонстрироваться работа конфигурации (например, предыдущий месяц). Убедитесь, что регистр сведений «Графики работы» содержит данные по выбранному периоду.
6. В выбранном периоде создайте и проведите документ с начислениями нескольким сотрудникам, затем — документ с начислением им премии (размер премии указывать не нужно). Сформируйте отчёт по перерасчётам. Объясните полученный результат.

7. В первом документе измените оклад некоторым сотрудникам, перепроведите его и сформируйте отчёт по перерасчётам. Объясните полученный результат.
8. Перепроведите документ с начислением премий, ничего в нём не меняя. Сформируйте отчёт по перерасчётам. Объясните полученный результат.
9. Запретите оперативное проведение документов «Начисления сотрудникам».
10. Переименуйте реквизит табличной части документа «Результат» в «Исходные данные».
11. Переименуйте ресурс регистра расчёта «Сумма» в «Исходные данные».
12. Добавьте в регистр расчёта ресурс «Результат».
13. Измените процедуру проведения документа так, чтобы реквизит табличной части документа «Исходные данные» соответствовал одноимённому ресурсу регистра.
14. Добавьте в процедуру проведения документа код, реализующий следующие действия (можно скопировать из приложения I):
 - запрос на выборку сотрудников из данного документа;
 - вызов процедуры расчёта начислений для вида расчёта «Оклад», массива сотрудников и данного набора движений документа, а также запись начислений;
 - вызов процедуры расчёта начислений для вида расчёта «Премия», массива сотрудников и данного набора движений документа, а также запись начислений.
15. Добавьте в модуль документа процедуру расчёта начислений (можно скопировать из приложения J):
 - для вида расчёта «Оклад»:
 - получение запросом периодов действия и фактических периодов действия записей об окладе сотрудников (источник запроса — виртуальная таблица «Данные графика»);
 - обход коллекции движений документа и расчёт результата по значению исходных данных пропорционально соотношению периода действия и фактического периода действия оклада;
 - при обнаружении нулевого фактического периода действия выдавать диагностическое сообщение;
 - для вида расчёта «Премия»:
 - получение запросом базовых записей для премии сотрудников (источник запроса — виртуальная таблица «База: Основные начисления»);
 - обход коллекции движений документа и расчёт результата как 10% от значения базы.
16. Создайте отчёт «Начисления», использующий схему компоновки данных:
 - добавьте набор данных — запрос;
 - в конструкторе запроса:
 - выберите реальную таблицу регистра расчёта;
 - из таблицы выберите следующие поля:
 - Сотрудник;

- ВидРасчета;
- ПериодДействияНачало;
- ПериодДействияКонец;
- Регистратор;
- Результат;
- на вкладке «Настройки»:
 - добавить группировку детальных записей;
 - внизу на вкладке «Выбранные поля» выбрать все доступные поля (за исключением системных).

17. Осуществите начисления, демонстрирующие влияние невыхода на результат расчёта оклада, а оклада — на результат расчёта премии.

Вопросы для подготовки к защите

1. Что такое план видов расчёта?
2. Какие настройки, относящиеся к взаимовлиянию видов расчёта, имеет план видов расчёта?
3. Какие предопределённые табличные части существуют в виде расчёта? Для чего они предназначены?
4. Что такое регистр расчёта?
5. Для чего используется график? Каким требованиям он должен удовлетворять?
6. Какие настройки, относящиеся к взаимовлиянию видов расчёта, имеет регистр расчёта?
7. Для чего используются перерасчёты?

Лабораторная работа №9

Механизм бизнес-процессов

Цель работы

Изучение использования бизнес-процессов и задач. Создание конфигурации, использующий механизм бизнес-процессов с адресацией задач.

Выполнение работы

1. Создайте пустой каталог «Lab_9». Создайте пустую информационную базу «Лабораторная работа №9» в этом каталоге.
2. Создайте справочники «Сотрудники», «Должности», «Подразделения».
3. В справочнике «Должности» создайте предопределённые элементы «Бухгалтер», «Кладовщик», «Менеджер отдела продаж».
4. В справочнике «Подразделения» создайте предопределённые элементы «Бухгалтерия», «Отдел продаж», «Склад».
5. Создайте документы «Счёт», «ПКО», «Отгрузка».
6. В ветви «Общие» дерева конфигурации создайте параметр сеанса «Текущий исполнитель» (тип — ссылка на справочник «Сотрудники»).
7. Создайте роль «Полные права», в которой установлены права на все действия в конфигурации, с настройкой «Устанавливать права для новых объектов».
8. Обновите конфигурацию базы данных. Создайте трёх пользователей информационной базы, установите каждому из них роль «Полные права».
9. Создайте регистр сведений «Адресация»:
 - неперiodический, независимый;
 - имеет измерения «Сотрудник», «Должность», «Подразделение» (типы — ссылки на соответствующие справочники), измерение «Сотрудник» является ведущим.
10. Создайте задачу «Задача»:
 - адресация — по регистру сведений «Адресация»;
 - имеет реквизиты адресации «Сотрудник», «Подразделение» и «Должность» (типы — ссылки на соответствующие справочники), связанные с соответствующими измерениями адресации;
 - основной реквизит адресации — «Сотрудник»;
 - текущий исполнитель — параметр сеанса «Текущий исполнитель»;
 - реквизиты — «Дата выполнения» (тип — **Дата**), «Исполнитель» (тип — ссылка на справочник «Сотрудники»).
11. Создайте основную форму задачи с настройками по умолчанию.
12. Создайте бизнес-процесс «Продажа»:
 - связан с задачами «Задача»;

- имеет реквизиты «Счёт», «Поступление денежных средств» и «Отгрузка товаров» (типы — ссылки на соответствующие документы);
- создайте карту маршрута бизнес-процесса:

- откройте карту маршрута из контекстного меню бизнес-процесса в дереве конфигурации либо с помощью кнопки «Действия» в окне объекта конфигурации;
- в открывшемся редакторе изобразите карту маршрута, состоящую из точки старта, точки завершения и трёх точек действия;
- соедините точки маршрута линиями и задайте им имена (см. рис.);
- для точек действия определите следующие значения адресации:
 - «Выписка счёта» — подразделение «Отдел продаж»;
 - «Получение оплаты» — должность «Бухгалтер»;
 - «Отгрузка товара со склада» — подразделение «Склад», должность «Кладовщик».

13. Для каждого документа создайте реквизит «Бизнес-процесс» (тип — ссылка на бизнес-процесс «Продажа»).

14. В режиме «Предприятие»:

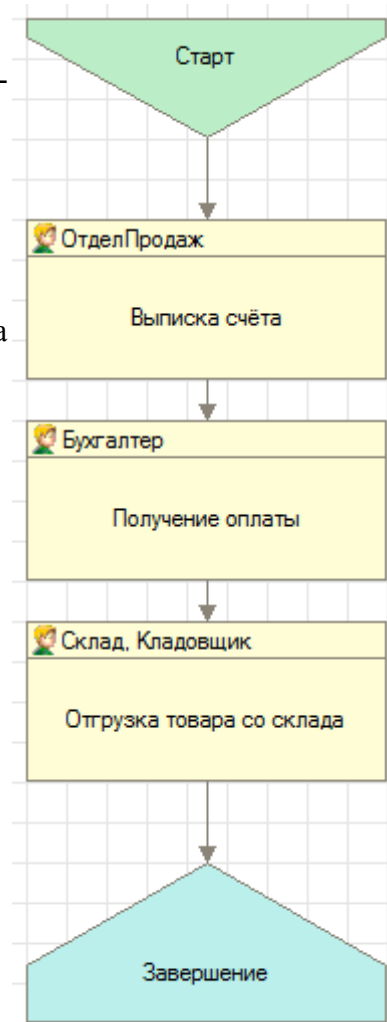
- создайте три элемента справочника «Сотрудники», наименования которых совпадают с именами созданных пользователей конфигурации;
- в регистре сведений добавьте по одной записи для каждого сотрудника (они занимают разные должности в соответствующих подразделениях).

Вернитесь в режим «Конфигуратор».

15. Откройте модуль приложения из контекстного меню в корне дерева конфигурации и добавьте в модуль приложения следующие процедуры (можно скопировать из приложения К):

- процедуру **ПриНачалеРаботыСистемы**, которая автоматически вызывается при запуске режима «Предприятие», подключает обработчик оповещения и записывает в параметр сеанса того сотрудника, от имени которого произведён запуск;
- экспортную процедуру **ОбработчикОповещения**, которая:
 - в случае, если происходит событие с данным именем ("ЗаписьСчёта"):
 - с помощью запроса определяет, к каким бизнес-процессу и задаче относится записанный документ;
 - записывает документ в реквизит бизнес-процесса;
 - помечает связанную с документом задачу как выполненную.

16. в модуль документа «Счёт» добавьте процедуру, которая при записи документа посылает оповещение "ЗаписьСчёта" (есть в приложении L).



17. Добавьте процедуру — обработчик открытия формы задачи, в модуле формы для него напишите программный код, устанавливающий значение текущего исполнителя как исполнителя задачи и сотрудника (есть в приложении L).
18. В карте маршрута бизнес-процесса:
 - для точки «Старт» создайте обработчик события «Перед стартом», сообщающий о старте бизнес-процесса (есть в приложении L);
 - для точки «Выписка счёта» создайте обработчик события «Обработка интерактивной активации», который в случае не привязанного к бизнес-процессу документа «Счёт» создаёт новый документ и открывает его форму (есть в приложении L).
19. Проверьте работу созданной конфигурации в режиме. Сценарий проверки:
 - 1) Создать экземпляр нового бизнес-процесса. Реквизиты бизнес-процесса не заполнять. При его записи будет создана новая задача «Выписка счёта», а сам бизнес-процесс будет отмечен как стартованный.
 - 2) Открыть форму задачи «Выписка счёта». При открытии формы также откроется форма нового документа «Счёт».
 - 3) Записать задачу «Выписка счёта», не закрывая форму документа.
 - 4) Записать документ «Счёт». При записи документа задача «Выписка счёта» отметится как завершённая, а также будет создана следующая задача — «Получение оплаты».
 - 5) Открыть записанный ранее экземпляр бизнес-процесса. Убедиться, что реквизит «Счёт» теперь содержит ссылку на записанный документ.
20. Внесите изменения в обработчики событий и модули, позволяющие реализовать аналогичное продолжение сценария:
 - 1) Открытие задачи «Получение оплаты» должно приводить к открытию формы документа «ПКО».
 - 2) Запись документа «ПКО» должны приводить к завершению задачи «Получение оплаты» и созданию задачи «Отгрузка товара со склада». При этом реквизит бизнес-процесса «Поступление денежных средств» содержит ссылку на этот документ.
 - 3) Открытие задачи «Отгрузка товара со склада» должно приводить к открытию формы документа «Отгрузка».
 - 4) Запись документа «Отгрузка» должна приводить к завершению задачи «Получение оплаты». При этом реквизит бизнес-процесса «Отгрузка товаров» содержит ссылку на этот документ.
 - 5) При завершении третьей задачи бизнес-процесс автоматически отметится как завершённый.

Вопросы для подготовки к защите

1. Что такое бизнес-процесс как объект конфигурации? Каковы его свойства?
2. Что такое задачи как объект конфигурации? Каковы их свойства?
3. Какие бывают типы маршрутов бизнес-процесса?
4. Что такое адресация, каковы её типы? Что такое реквизиты адресации?
5. Что такое параметр сеанса?

Приложение А

Текст запроса для лабораторной работы №2

```
ВЫБРАТЬ
    ОстаткиНоменклатурыОстатки.Номенклатура,
    ОстаткиНоменклатурыОстатки.Склад,
    ЕСТЬNULL (ОстаткиНоменклатурыОстатки.КоличествоОстаток, 0)
        КАК КоличествоОстатка,
    ЕСТЬNULL (ОстаткиНоменклатурыОстатки.СуммаОстаток, 0) КАК СуммаОстатка,
    ТаблицаСписания.Количество КАК КоличествоСписывается
ИЗ
    (ВЫБРАТЬ
        РасходнаяНакладнаяТовары.Номенклатура КАК Номенклатура,
        СУММА (РасходнаяНакладнаяТовары.Количество) КАК Количество
    ИЗ
        Документ.РасходнаяНакладная.Товары КАК РасходнаяНакладнаяТовары
    ГДЕ
        РасходнаяНакладнаяТовары.Ссылка = &ЭтотДокумент

    СГРУППИРОВАТЬ ПО
        РасходнаяНакладнаяТовары.Номенклатура) КАК ТаблицаСписания
ЛЕВОЕ СОЕДИНЕНИЕ
    РегистрНакопления.ОстаткиНоменклатуры.Остатки (
        &МоментВремени,
        Склад = &Склад И Номенклатура В
        (ВЫБРАТЬ РАЗЛИЧНЫЕ
            РасходнаяНакладнаяТовары.Номенклатура
        ИЗ
            Документ.РасходнаяНакладная.Товары
            КАК РасходнаяНакладнаяТовары
        ГДЕ
            РасходнаяНакладнаяТовары.Ссылка = &ЭтотДокумент))
КАК ОстаткиНоменклатурыОстатки
ПО ТаблицаСписания.Номенклатура =
    ОстаткиНоменклатурыОстатки.Номенклатура
```

Приложение В

Фрагмент кода модуля для лабораторной работы №2

```
ЗапросОстатков.УстановитьПараметр("МоментВремени", Ссылка.МоментВремени());
ЗапросОстатков.УстановитьПараметр("Склад", Склад);
ЗапросОстатков.УстановитьПараметр("ЭтотДокумент", Ссылка);
ВыборкаОстатков = ЗапросОстатков.Выполнить().Выбрать();
Пока ВыборкаОстатков.Следующий() Цикл
    Если ВыборкаОстатков.КоличествоСписывается = 0 Тогда
        Сообщить("Списывается нулевое количество товара "" +
            ВыборкаОстатков.Номенклатура + "".",
            СтатусСообщения.Важное);
        Отказ = Истина;
        Возврат;
    КонецЕсли;
    Если ВыборкаОстатков.КоличествоСписывается <=
        ВыборкаОстатков.КоличествоОстатка Тогда
        Движение = Движения.ОстаткиНоменклатуры.Добавить();
        Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Номенклатура = ВыборкаОстатков.Номенклатура;
        Движение.Склад = Склад;
        Движение.Количество = ВыборкаОстатков.КоличествоСписывается;
        Движение.Сумма = ВыборкаОстатков.КоличествоСписывается /
            ВыборкаОстатков.КоличествоОстатка *
            ВыборкаОстатков.СуммаОстатка;
    Иначе
        Сообщить("Товар "" + ВыборкаОстатков.Номенклатура +
            "" на складе "" + ВыборкаОстатков.Склад +
            "": списывается " +
            Формат(ВыборкаОстатков.КоличествоСписывается,
            "ЧЦ=15; ЧДЦ=3; ЧН=") + ", но осталось только " +
            Формат(ВыборкаОстатков.КоличествоОстатка,
            "ЧЦ=15; ЧДЦ=3; ЧН=") + ".", СтатусСообщения.Важное);
        Отказ = Истина;
        Возврат;
    КонецЕсли;
КонецЦикла
```


Приложение С

Текст запроса для лабораторной работы №3

```
ВЫБРАТЬ
    ОстаткиНоменклатурыОстатки.Номенклатура КАК Номенклатура,
    ОстаткиНоменклатурыОстатки.Склад,
    ОстаткиНоменклатурыОстатки.Партия КАК Партия,
    ЕСТЬNULL (ОстаткиНоменклатурыОстатки.КоличествоОстаток, 0)
        КАК КоличествоОстатка,
    ЕСТЬNULL (ОстаткиНоменклатурыОстатки.СуммаОстаток, 0) КАК СуммаОстатка,
    ТаблицаСписания.Количество КАК КоличествоСписывается
ИЗ
    (ВЫБРАТЬ
        РасходнаяНакладнаяТовары.Номенклатура КАК Номенклатура,
        СУММА (РасходнаяНакладнаяТовары.Количество) КАК Количество
    ИЗ
        Документ.РасходнаяНакладная.Товары КАК РасходнаяНакладнаяТовары
    ГДЕ
        РасходнаяНакладнаяТовары.Ссылка = &ЭтотДокумент
    СГРУППИРОВАТЬ ПО
        РасходнаяНакладнаяТовары.Номенклатура) КАК ТаблицаСписания
ЛЕВОЕ СОЕДИНЕНИЕ
    РегистрНакопления.ОстаткиНоменклатуры.Остатки (
        &МоментВремени,
        Склад = &Склад
        И Номенклатура В
            (ВЫБРАТЬ РАЗЛИЧНЫЕ
                РасходнаяНакладнаяТовары.Номенклатура
            ИЗ
                Документ.РасходнаяНакладная.Товары
                КАК РасходнаяНакладнаяТовары
            ГДЕ
                РасходнаяНакладнаяТовары.Ссылка =
                    &ЭтотДокумент) )
        КАК ОстаткиНоменклатурыОстатки
    ПО ТаблицаСписания.Номенклатура =
        ОстаткиНоменклатурыОстатки.Номенклатура
УПОРЯДОЧИТЬ ПО
    Партия
ИТОГИ
    СУММА (КоличествоОстатка) ,
    СУММА (СуммаОстатка) ,
    МАКСИМУМ (КоличествоСписывается)
ПО
    Номенклатура
```

Приложение D

Фрагмент кода модуля для лабораторной работы №3

```
ЗапросОстатков.УстановитьПараметр("МоментВремени", МоментВремени());
ЗапросОстатков.УстановитьПараметр("Склад", Склад);
ЗапросОстатков.УстановитьПараметр("ЭтотДокумент", Ссылка);

ВыборкаОстатков = ЗапросОстатков.Выполнить().Выбрать (
    ОбходРезультатаЗапроса.ПоГруппировкам);

Пока ВыборкаОстатков.Следующий() Цикл
    Если ВыборкаОстатков.КоличествоСписывается = 0 Тогда
        Сообщить("Списывается нулевое количество товара "" +
            ВыборкаОстатков.Номенклатура + "".",
            СтатусСообщения.Важное);
        Отказ = Истина;
        Возврат;
    КонецЕсли;
    Если ВыборкаОстатков.КоличествоСписывается <=
        ВыборкаОстатков.КоличествоОстатка Тогда
        ОсталосьСписать = ВыборкаОстатков.КоличествоСписывается;
        ВыборкаПартий = ВыборкаОстатков.Выбрать();
        Пока ОсталосьСписать > 0 Цикл
            ВыборкаПартий.Следующий();
            Списывается = Мин(ОсталосьСписать,
                ВыборкаПартий.КоличествоОстатка);
            Движение = Движения.ОстаткиНоменклатуры.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = ВыборкаОстатков.Номенклатура;
            Движение.Партия = ВыборкаПартий.Партия;
            Движение.Склад = Склад;
            Движение.Количество = Списывается;
            Движение.Сумма = Списывается /
                ВыборкаПартий.КоличествоОстатка *
                ВыборкаПартий.СуммаОстатка;
            ОсталосьСписать = ОсталосьСписать - Списывается;
        КонецЦикла;
    Иначе
        Сообщить("Товар "" + ВыборкаОстатков.Номенклатура +
            "" на складе "" + ВыборкаОстатков.Склад +
            """: списывается " +
            Формат(ВыборкаОстатков.КоличествоСписывается,
            "ЧЦ=15; ЧДЦ=3; ЧН=") + ", но осталось только " +
            Формат(ВыборкаОстатков.КоличествоОстатка,
            "ЧЦ=15; ЧДЦ=3; ЧН=") + ".", СтатусСообщения.Важное);
        Отказ = Истина;
        Возврат;
    КонецЕсли;
КонецЦикла
```

Приложение Е

Текст запроса для лабораторной работы №6

```
ВЫБРАТЬ
    РегламентированныйОстатки.Субконто3 КАК Сделка,
    РегламентированныйОстатки.СуммаОстатокКт КАК Сумма
ИЗ
    РегистрБухгалтерии.Регламентированный.Остатки (
        &МоментВремени,
        Счет = &Дебиторка,
        ,
        Субконто1 = &Контрагент
        И Субконто2 = &Договор) КАК РегламентированныйОстатки

УПОРЯДОЧИТЬ ПО
    Сделка
ИТОГИ
    СУММА (Сумма)
ПО
    ОБЩИЕ
```

Приложение F

Фрагмент кода модуля для лабораторной работы №6

```
ЗапросПоСделкам.УстановитьПараметр("МоментВремени", МоментВремени());
ЗапросПоСделкам.УстановитьПараметр("Контрагент", Контрагент);
ЗапросПоСделкам.УстановитьПараметр("Договор", Договор);
ЗапросПоСделкам.УстановитьПараметр("Дебиторка",
    ПланыСчетов.Регламентированный.РасчётыСПоставщикамиИПодрядчиками);

ВыборкаИтог = ЗапросПоСделкам.Выполнить()
    .Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);

ОсталосьЗаккрыть = Сумма;

Если Не ВыборкаИтог.Следующий() Тогда
    Сообщить("Задолженность перед контрагентом "" + Контрагент +
        "" по договору "" + Договор +
        "" отсутствует.");
    Отказ = Истина;
    Возврат;
КонецЕсли;

Если ОсталосьЗаккрыть > ВыборкаИтог.Сумма Тогда
    Сообщить("Задолженность перед контрагентом "" + Контрагент +
        "" по договору "" + Договор +
        "" меньше погашаемой.");
    Отказ = Истина;
    Возврат;
КонецЕсли;

ВыборкаПоСделкам = ВыборкаИтог.Выбрать();
Пока ОсталосьЗаккрыть > 0 Цикл
    ВыборкаПоСделкам.Следующий();
    НадоЗаккрыть = Мин(ОсталосьЗаккрыть, ВыборкаПоСделкам.Сумма);
    Движение = Движения.Регламентированный.Добавить();
    Движение.СчетДт =
        ПланыСчетов.Регламентированный.РасчётыСПоставщикамиИПодрядчиками;
    Движение.СчетКт = ПланыСчетов.Регламентированный.РасчётныеСчета;
    Движение.Период = Дата;
    Движение.Сумма = НадоЗаккрыть;
    Движение.СубконтоДт[ПланыВидовХарактеристик
        .ВидыСубконтоРегламентированные.Контрагенты] =
        Контрагент;
    Движение.СубконтоДт[ПланыВидовХарактеристик
        .ВидыСубконтоРегламентированные.Договоры] =
        Договор;
    Движение.СубконтоДт[ПланыВидовХарактеристик
        .ВидыСубконтоРегламентированные.Сделки] =
        ВыборкаПоСделкам.Сделка;
    Движение.СубконтоКт[ПланыВидовХарактеристик
        .ВидыСубконтоРегламентированные.БанковскиеСчета] =
        БанковскийСчёт;
    Сообщить("Закрыта сделка "" + ВыборкаПоСделкам.Сделка +
        "" + ?(НадоЗаккрыть = ВыборкаПоСделкам.Сумма, "",
        " (частично)");
    ОсталосьЗаккрыть = ОсталосьЗаккрыть - НадоЗаккрыть;
КонецЦикла
```

Приложение G

Текст запроса для лабораторной работы №7

ВЫБРАТЬ

ОсновныеНачисленияФактическийПериодДействия.Сотрудник,
ОсновныеНачисленияФактическийПериодДействия.ВидРасчета,
ОсновныеНачисленияФактическийПериодДействия.ПериодДействияНачало,
ОсновныеНачисленияФактическийПериодДействия.ПериодДействияКонец,
ОсновныеНачисленияФактическийПериодДействия.Сумма,
ОсновныеНачисленияФактическийПериодДействия.Регистратор

ИЗ

РегистрРасчета.ОсновныеНачисления.ФактическийПериодДействия
КАК ОсновныеНачисленияФактическийПериодДействия

Приложение Н

Фрагмент кода модуля для лабораторной работы №7

```
ВыборкаРезультата = ЗапросНачислений.Выполнить().Выбрать();

Диаграмма = ЭлементыФормы.ДиаграммаГанта;
Диаграмма.Обновление = Ложь;
Диаграмма.Очистить();
Диаграмма.ОтображатьЗаголовок = Ложь;

Пока ВыборкаРезультата.Следующий() Цикл
    ТекущаяСерия = Диаграмма.УстановитьСерию(ВыборкаРезультата.ВидРасчета,
                                              ВыборкаРезультата.ВидРасчета);
    ТекущаяТочка = Диаграмма.УстановитьТочку(ВыборкаРезультата.Сотрудник,
                                              ВыборкаРезультата.Сотрудник);
    ТекущееЗначение = Диаграмма.ПолучитьЗначение(ТекущаяТочка, ТекущаяСерия);
    ТекущийИнтервал = ТекущееЗначение.Добавить();
    ТекущийИнтервал.Начало = ВыборкаРезультата.ПериодДействияНачало;
    ТекущийИнтервал.Конец = ВыборкаРезультата.ПериодДействияКонец;
    ТекущийИнтервал.Текст = ВыборкаРезультата.Регистратор;
    ТекущийИнтервал.Расшифровка = ВыборкаРезультата.Регистратор;
КонецЦикла;

Для Каждого Серия Из Диаграмма.Серии Цикл
    Если Серия.Значение = ПланыВидовРасчета.ОсновныеНачисления.Оклад Тогда
        Серия.Цвет = WEBЦвета.ГолубойСоСтальнымОттенком;
    ИначеЕсли Серия.Значение =
        ПланыВидовРасчета.ОсновныеНачисления.Премия Тогда
        Серия.Цвет = WEBЦвета.Зеленый;
    ИначеЕсли Серия.Значение =
        ПланыВидовРасчета.ОсновныеНачисления.Невыход Тогда
        Серия.Цвет = WEBЦвета.Красный;
    КонецЕсли;
КонецЦикла;

Диаграмма.Обновление = Истина;
```

Приложение I

Фрагмент процедуры проведения документа для лабораторной работы №8

Движения.ОсновныеНачисления.Записать ();

ЗапросПоСотрудникам = Новый Запрос;

ЗапросПоСотрудникам.Текст =

```
"ВЫБРАТЬ РАЗЛИЧНЫЕ
|      НачисленияСотрудникамСостав.Сотрудник
| ИЗ
|      Документ.НачисленияСотрудникам.Состав
|      КАК НачисленияСотрудникамСостав
| ГДЕ
|      НачисленияСотрудникамСостав.Ссылка = &ТекущийДокумент";
```

ЗапросПоСотрудникам.УстановитьПараметр ("ТекущийДокумент", Ссылка);

МассивСотрудников = ЗапросПоСотрудникам.Выполнить ();

Выгрузить ().ВыгрузитьКолонку ("Сотрудник");

РассчитатьНачисления (ПланыВидовРасчета.ОсновныеНачисления.

Оклад, МассивСотрудников);

Движения.ОсновныеНачисления.Записать (, Истина);

РассчитатьНачисления (ПланыВидовРасчета.ОсновныеНачисления.

Премия, МассивСотрудников);

Движения.ОсновныеНачисления.Записать (, Истина);

Приложение J

Процедура расчёта начислений для лабораторной работы №8

Процедура РассчитатьНачисления(ВидРасчёта, МассивСотрудников)

Регистратор = Движения.ОсновныеНачисления.Отбор.Регистратор.Значение;

Если ВидРасчёта = ПланыВидовРасчета.ОсновныеНачисления.Оклад Тогда

ЗапросНормаФакт = Новый Запрос;

ЗапросНормаФакт.Текст = "ВЫБРАТЬ

| ОсновныеНачисленияДанныеГрафика.ЗначениеПериодДействия КАК Норма,
| ОсновныеНачисленияДанныеГрафика.ЗначениеФактическийПериодДействия
| КАК Факт,

| ОсновныеНачисленияДанныеГрафика.НомерСтроки,
| ОсновныеНачисленияДанныеГрафика.Сотрудник

| ИЗ

| РегистрРасчета.ОсновныеНачисления.ДанныеГрафика (
| Регистратор = &Регистратор
| И ВидРасчета = &ВидРасчета
| И Сотрудник В (&МассивСотрудников))
| КАК ОсновныеНачисленияДанныеГрафика";

ЗапросНормаФакт.УстановитьПараметр("Регистратор", Регистратор);

ЗапросНормаФакт.УстановитьПараметр("ВидРасчета", ВидРасчёта);

ЗапросНормаФакт.УстановитьПараметр("МассивСотрудников", МассивСотрудников);

ВыборкаРезультата = ЗапросНормаФакт.Выполнить().Выбрать();

Для Каждого ЗаписьРегистра Из Движения.ОсновныеНачисления Цикл

СтруктураНомер = Новый Структура("НомерСтроки");

СтруктураНомер.НомерСтроки = ЗаписьРегистра.НомерСтроки;

ВыборкаРезультата.Сбросить();

Если ВыборкаРезультата.НайтиСледующий(СтруктураНомер) Тогда

Если ВыборкаРезультата.Норма = 0 Тогда

Сообщить("По сотруднику "" + ВыборкаРезультата.Сотрудник +
"" нет рабочих дней в заданном периоде.");

ЗаписьРегистра.Результат = 0;

Иначе

ЗаписьРегистра.Результат = ЗаписьРегистра.

ИсходныеДанные * ВыборкаРезультата.Факт /

ВыборкаРезультата.Норма;

Сообщить("По сотруднику "" + ВыборкаРезультата.Сотрудник +
"" рассчитан оклад.");

КонецЕсли;

КонецЕсли;

КонецЦикла;

ИначеЕсли ВидРасчёта = ПланыВидовРасчета.ОсновныеНачисления.Премия Тогда

ЗапросБаза = Новый Запрос;

ЗапросБаза.Текст = "ВЫБРАТЬ

```
|      ОсновныеНачисленияБазаОсновныеНачисления.РезультатБаза КАК База,  
|      ОсновныеНачисленияБазаОсновныеНачисления.НомерСтроки,  
|      ОсновныеНачисленияБазаОсновныеНачисления.Сотрудник  
|ИЗ  
|      РегистрРасчета.ОсновныеНачисления.БазаОсновныеНачисления (  
|          &Измерения,  
|          &Измерения,  
|          ,  
|          Регистратор = &Регистратор  
|          И ВидРасчета = &ВидРасчета  
|          И Сотрудник В (&МассивСотрудников)) КАК  
|      ОсновныеНачисленияБазаОсновныеНачисления";
```

Измерения = Новый Массив(1);

Измерения[0] = "Сотрудник";

ЗапросБаза.УстановитьПараметр("Измерения", Измерения);

ЗапросБаза.УстановитьПараметр("Регистратор", Регистратор);

ЗапросБаза.УстановитьПараметр("ВидРасчета", ВидРасчёта);

ЗапросБаза.УстановитьПараметр("МассивСотрудников", МассивСотрудников);

ВыборкаРезультата = ЗапросБаза.Выполнить().Выбрать();

Для Каждого ЗаписьРегистра Из Движения.ОсновныеНачисления Цикл

СтруктураНомер = Новый Структура("НомерСтроки");

СтруктураНомер.НомерСтроки = ЗаписьРегистра.НомерСтроки;

ВыборкаРезультата.Сбросить();

Если ВыборкаРезультата.НайтиСледующий(СтруктураНомер) Тогда

ЗаписьРегистра.Результат = ВыборкаРезультата.База * 10 / 100;

Сообщить("По сотруднику "" + ВыборкаРезультата.Сотрудник +
"" рассчитана премия.");

КонецЕсли;

КонецЦикла;

КонецЕсли

КонецПроцедуры

Приложение К

Модуль приложения для лабораторной работы №9

Процедура ПриНачалеРаботыСистемы()

```
    ПараметрыСеанса.ТекущийИсполнитель =  
        Справочники.Сотрудники.НайтиПоНаименованию(ИмяПользователя());  
    ПодключитьОбработчикОповещения("ОбработчикОповещения");  
    Если ПараметрыСеанса.ТекущийИсполнитель = Справочники.Сотрудники.ПустаяСсылка() Тогда  
        Сообщить("Сотрудник не опознан");  
    Иначе  
        Сообщить("Зарегистрирован вход сотрудника " + "" +  
            ПараметрыСеанса.ТекущийИсполнитель + "");  
    КонецЕсли
```

КонецПроцедуры

Процедура ОбработчикОповещения(ИмяСобытия, Параметр, Источник) Экспорт

```
    Если ИмяСобытия = "ЗаписьСчёта" Тогда  
        ЗапросЗадачи = Новый Запрос;  
        ЗапросЗадачи.Текст =  
            "ВЫБРАТЬ  
            |      Задача.Ссылка,  
            |      Продажа.Ссылка КАК БП  
            |ИЗ  
            |      БизнесПроцесс.Продажа КАК Продажа  
            |      ЛЕВОЕ СОЕДИНЕНИЕ Задача.Задача.ЗадачиПоИсполнителю КАК Задача  
            |      ПО Продажа.Ссылка = Задача.БизнесПроцесс  
            |ГДЕ  
            |      (НЕ Задача.Выполнена)  
            |      И Продажа.Ссылка = &ЭтотБП";  
        ЗапросЗадачи.УстановитьПараметр("ЭтотБП", Источник.БизнесПроцесс);  
        Выборка = ЗапросЗадачи.Выполнить().Выбрать();  
        Пока Выборка.Следующий() Цикл  
            ТекущаяЗадача = Выборка.Ссылка.ПолучитьОбъект();  
            БП = Выборка.БП.Ссылка.ПолучитьОбъект();  
            БП.Счёт = Источник.Ссылка;  
            БП.Записать();  
            Если ТекущаяЗадача.ТочкаМаршрута =  
                БизнесПроцессы.Продажа.ТочкиМаршрута.ВыпискиСчёта Тогда  
                ТекущаяЗадача.ДатаВыполнения = ТекущаяДата();  
                ТекущаяЗадача.ВыполнитьЗадачу();  
            КонецЕсли;  
        КонецЦикла;  
    КонецЕсли;
```

КонецПроцедуры

Приложение L

Процедуры из модулей для лабораторной работы №9

Процедура из модуля документа «Счёт»:

Процедура ПриЗаписи (Отказ)

Оповестить ("Запись Счёта", , ЭтотОбъект);

КонецПроцедуры

Процедура из модуля формы задачи «Задача»:

Процедура ПриОткрытии ()

Если Сотрудник = Справочники.Сотрудники.ПустаяСсылка () Тогда

Сотрудник = ПараметрыСеанса.ТекущийИсполнитель;

КонецЕсли;

Если Исполнитель = Справочники.Сотрудники.ПустаяСсылка () Тогда

Исполнитель = ПараметрыСеанса.ТекущийИсполнитель;

КонецЕсли;

КонецПроцедуры

Процедуры из модуля бизнес-процесса «Продажа»:

Процедура СтартПередСтартом (ТочкаМаршрутаБизнесПроцесса, Отказ)

Сообщить ("Старт бизнес-процесса.");

КонецПроцедуры

Процедура ВыпискаСчётаОбработкаИнтерактивнойАктивации (ТочкаМаршрутаБизнесПроцесса,

Задача, СтандартнаяОбработка)

Если Счёт.Ссылка <> Документы.Счёт.ПустаяСсылка () Тогда

Возврат;

КонецЕсли;

НовыйДокумент = Документы.Счёт.СоздатьДокумент ();

НовыйДокумент.БизнесПроцесс = Ссылка;

НовыйДокумент.ПолучитьФорму ().Открыть ();

КонецПроцедуры