

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ  
ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

**Б1.В.ДВ.05.01 Основы программирования и конфигурирования в корпоративных  
информационных системах (II часть)**

**Направление подготовки (специальность) 09.03.01 Информатика и вычислительная  
техника**

**Профиль образовательной программы “Автоматизированные системы обработки  
информации и управления”**

**Форма обучения очная**

## **СОДЕРЖАНИЕ**

<b>1. Конспект лекций .....</b>
<b>1.1 Лекция № 1 Введение в Регистры .....</b>
<b>1.2 Лекция № 2 Соединения источников в запросе .....</b>
<b>1.3 Лекция № 3 Формы и редактор форм .....</b>
<b>1.4 Лекция № 4 Объектная модель .....</b>
<b>1.5 Лекция № 5 Сервисные функции .....</b>
<b>1.6 Лекция № 6 Справочники .....</b>
<b>1.7 Лекция № 7 Создание документов. Доступ к данным документа</b>
<b>1.8 Лекция № 8 Источники данных. Структура запроса (описание запроса).....</b>
<b>1.9 Лекция № 9 Отчеты. Рабочий стол. Критерии отбора</b>
<b>2. Методические материалы по выполнению лабораторных работ .....</b>
<b>2.1 Лабораторная работа № 1 ЛР-1 Введение в Регистры</b>
<b>2.2 Лабораторная работа № 2 ЛР-2 Соединения источников в запросе</b>
<b>2.3 Лабораторная работа № 3 ЛР-3 Формы и редактор форм</b>
<b>2.4 Лабораторная работа № 4 ЛР-4 Объектная модель</b>
<b>2.5 Лабораторная работа № 5 ЛР-5 Сервисные функции</b>
<b>2.6 Лабораторная работа № 6 ЛР-6 Обработчики событий формы</b>
<b>2.7 Лабораторная работа № 7 ЛР-7, 8, 9 Справочники</b>
<b>2.8 Лабораторная работа № 8 ЛР-10, 11, 12 Создание документов. Доступ к данным документа</b>
<b>2.9 Лабораторная работа № 9 ЛР-13, 14, 15 Источники данных. Структура запроса (описание запроса)</b>
<b>2.10 Лабораторная работа № 1 ЛР-16, 17, 18 Отчеты. Рабочий стол. Критерии отбора</b>

# 1. КОНСПЕКТ ЛЕКЦИЙ

## 1.1. Лекция № 1 (2 часа)

Тема: «Введение в Регистры»

### 1.1.1. Вопросы лекции:

- 1 Виды регистров.
- 2 Регистр сведений "Артикулы".
- 3 Связи наборов данных в компоновке.

### 1.1.2. Краткое содержание вопросов:

- 1 Виды регистров.

Регистры - это таблицы для накопления оперативных данных и получения сводной информации. Данные в регистры добавляются только при проведении документов. Сведения из регистров используются для формирования отчетов. Классическая схема использования регистров в 1С:Предприятие выглядит следующим образом:

Документы => Регистры => Отчеты

В них хранится информация об изменении состояний объектов.

- Регистры сведений
- Регистры бухгалтерии
- Регистры накоплений
- Регистры расчета

Регистр сведений позволяет хранить произвольные данные в разрезе нескольких изменений. Например, в регистре сведений можно хранить курсы валют в разрезе валют, или цены предприятия в разрезе номенклатуры и типа цен. Регистр сведений фактически представляет многомерный массив данных, который может выдать функцию по определенному набору аргументов.

Регистры бухгалтерии - это прикладные объекты конфигурации. Они используются в механизме бухгалтерского учета и позволяют вести многоуровневый и многомерный аналитический учет, учет по нескольким планам счетов, опциональное ведение количественного, суммового и валютного учета по отдельным разрезам аналитики и т.д. По своему виду регистр бухгалтерии напоминает регистр накопления: он может иметь набор изменений, в разрезе которых накапливаются числовые значения ресурсов. Также он может содержать набор реквизитов для хранения дополнительной информации, связанной с записями регистр.

Регистры накопления - это прикладные объекты конфигурации. Они составляют основу механизма учета движения средств (финансов, товаров, материалов и т.д.), который позволяет автоматизировать такие направления, как складской учет, взаиморасчеты, планирование. Регистр накопления образует многомерную систему измерений и позволяет "накапливать" числовые данные в разрезе нескольких измерений. Например, в таком регистре можно накапливать информацию об остатках товаров в разрезе номенклатуры и склада, или информацию об объемах продаж в разрезе номенклатуры и подразделения компании.

Регистры расчета - это прикладные объекты конфигурации. Они используются в механизме сложных периодических расчетов и служат для хранения записей о тех или иных видах расчета, которые необходимо выполнить, а также для хранения промежуточных данных и самих результатов выполненных расчетов. Информация в регистре расчета хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов. Измерения регистра описывают разрезы, в которых хранится информация, а ресурсы регистра непосредственно содержат хранимую информацию.

2 Регистр сведений "Артикулы".

Регистр сведений – объект конфигурации 1С:Предприятие 8 предназначенный для описания структуры хранения данных в разрезе нескольких измерений.

На основе объекта конфигурации Регистр сведений платформа создает в базе данных таблицу, в которой может храниться произвольная информация, «привязанная» к набору измерений.

Информация в регистре сведений хранится в виде набора строк (записей). Каждая строка содержит набор измерений и соответствующие им значения ресурсов. Комбинация измерений однозначно идентифицирует запись.

### **Периодичность**

Регистры сведений информация в которых развернута во времени, называются периодическими. При проектировании регистра сведений можно указать минимальную периодичность, с которой записи будут заноситься в регистр. Допустимые значения периодичности:

- Непериодический
- По позиции регистратора
- В пределах секунды
- В пределах дня
- В пределах месяца
- В пределах квартала
- В пределах года
- Если регистр не периодический, то поле "Период" для него не создается.

## **3 Связи наборов данных в компоновке.**

Схема компоновки данных (1С СКД) – удобный конструктор для создания в программных продуктах 1С:Предприятие сложных отчетов, способствующих развитию и отслеживанию автоматизации производства, позволяющих сделать их максимально гибкими и красивыми за минимум времени. Дополнительный плюс Схемы компоновки данных (1С СКД) - автоматическое формирование управляемой формы отчета, а при дальнейшем развитии данного направления, является важным фактором при выборе метода разработки отчета. Но из-за сложности структуры Схемы компоновки данных (1С СКД) и огромного количества настроек зачастую приводит к более длительной разработке отчета, чем через «конструктор выходной формы». Поэтому программисту 1С необходимо разбираться во всех тонкостях работы Схемы компоновки данных (1С СКД), чтобы в дальнейшем ускорить время разработки формирования отчетов.

Разберем первые три вкладки Схемы компоновки данных (1С СКД) – набор данных, связи набора данных и вычисляемые поля.

### **Набор данных в 1С СКД**

Набор данных включает в себя возможность создания трёх объектов – запрос, объект и объединение, остановимся подробнее на каждом из них:

#### **Запрос**

Представляет собой обычный запрос, который формируется по кнопке Конструктор запроса. Если установлен флаг Автозаполнение, то все выбранные реквизиты автоматически попадут в поля набора данных. Так же есть возможность самим настроить заполнение полей в запросе на вкладке Компоновка данных, где находятся три вкладки:

- таблицы, здесь выбираются таблицы, которые будут участвовать при формировании отчета, обычно выбираются данные по умолчанию, так как на вкладке Таблицы и поля мы уже выбрали нужные нам документы, справочники, регистры...

- поля, здесь мы выбираем те объекты, которые должны попасть в отчет, флаг дочерние указывает, будут ли доступные дочерние элементы по объекту или нет, логично, что у строковых, числовых и подобных данных нельзя будет установить флаг в положение Истина.

- условия, здесь мы выбираем те объекты, которые можно будет использовать в условиях в СКД.

## 1.2. Лекция № 2 (2 часа)

**Тема:** «Соединения источников в запросе»

### 1.2.1. Вопросы лекции:

1 Пакетный запрос

2 Временные таблицы

### 1.2.2. Краткое содержание вопросов:

1 Пакетный запрос

Пакетный запрос это нескольких запросов, описанных как один большой запрос. Эти запросы выполняются последовательно один за другим. Результат любого из этих запросов можно поместить во временную таблицу и обратиться к ней из другого запроса.

В чём плюсы этого метода?

Когда результаты такого запроса помещаются во временную таблицу, они могут использоваться несколько раз. При этом не нужно несколько раз выполнять один и тот же вложенный запрос, а выбирать данные из готовой таблицы.

В чём минусы этого метода?

При использовании временных таблиц в пакетных запросах система формирует в базе данных реальные таблицы. Чем это плохо? Тратится время на создание таблицы, а после исполнения запроса, на её уничтожение.

2 Временные таблицы

В 8.1. появились временные таблицы. Они хранятся на сервере и позволяют писать более простые запросы.

Временные таблицы хранятся в объекте типа МенеджерВременныхТаблиц. Когда этот объект уничтожается, уничтожаются и временные таблицы.

Чтобы создать временную таблицу, используется ключевое слово "ПОМЕСТИТЬ В", например:

ВЫБРАТЬ \* ИЗ Справочник.Номенклатура ПОМЕСТИТЬ В ВТТовары

Временная таблица создается при выполнении запроса, если повторно выполнить запрос, то выдастся ошибка, что таблица уже существует.

Описания временных таблиц хранятся в свойстве запроса МенеджерВременныхТаблиц. К сожалению, в духе 1С, нельзя получить список временных таблиц, которые хранятся в запросе.

Пример, как можно выгрузить временную таблицу в таблицу значений, а заодно и как использовать менеджер временных таблиц:

Запрос=Новый Запрос("ВЫБРАТЬ \* ИЗ Справочник.Номенклатура ПОМЕСТИТЬ В ВТТовары");

Запрос.Выполнить(); //Создалась таблица ВТТовары

ЗапросТМП=Новый Запрос("ВЫБРАТЬ \* ИЗ ВТТовары");

ЗапросТМП.МенеджерВременныхТаблиц=Запрос.МенеджерВременныхТаблиц;  
//Копируем ссылку на временные таблицы

ТЗ=ЗапросТМП.Выполнить().Выгрузить(); //Получаем временную таблицу в таблице значений

## 1.3. Лекция № 3 (2 часа)

**Тема:** «Формы и редактор форм»

### 1.3.1. Вопросы лекции:

1 Виды форм: констант, документов, списков.

2 Периодические регистры сведений.

3 Курсы валют.

### **1.3.2. Краткое содержание вопросов:**

1 Виды форм: констант, документов, списков.

Если форма констант не создана программистом, то она генерируется системой автоматически. Когда констант много, то появляется прокрутка, и форма может быть не очень удобной. Кроме того, возникает желание сгруппировать константы по смыслу, например, с помощью закладок (панели).

В качестве формы констант обычно используется общая форма, которую можно создать конструктором форм:

- a) Щелкните правой кнопкой на ветке "Константы" в дереве конфигурации
- b) Выберите "Создать форму констант" или "Создать основную форму констант".

в) Разместите на форме элементы управления (поля ввода, флажки), связанные с константами.

Основная форма констант прописывается в свойствах всей конфигурации.

Форму констант можно получать из программных модулей с помощью конструкции:

Форма = Константы.ПолучитьФорму(<Имя>); //если имя не указано, то основная форма констант

Форма.Открыть();

Форму констант также можно вызывать по собственному пункту в меню, например, Сервис => Настройка параметров учета или Операции => Константы.

У формы констант создается реквизит "КонстантыНабор". Далее начинается самое интересное. По умолчанию в набор констант входят только имеющиеся на данный момент константы (свойство "Состав"). При создании новой константы ее не получится разместить на форме: ее просто нет в списке констант, который появляется по кнопке "Размещение данных"). Можно попытаться разместить элемент управления и связать его с константой, но в режиме Предприятие такой элемент будет недоступным. Это происходит, потому что новая константа не входит в набор констант для этой формы.

Чтобы новую константу разместить на форме, нужно сделать одно из двух:

1. Или включить ее в набор (свойство Состав)
2. Или у формы установить флаг "Использовать все константы", тогда свойство Набор становится недоступным.

Отмечу, что константы типа "Хранилище значения" нельзя разместить на форме, даже если они входят в набор или установлен флаг "Использовать все константы". Их редактирование нужно запрограммировать самостоятельно.

### **2 Периодические регистры сведений.**

Регистры сведений предназначены для хранения информации, развернутой по комбинации измерений. Главное отличие периодического регистра сведений от обычного заключается в том, что в нем присутствует дополнительное системное измерение "Период", имеющее тип "дата". Это позволяет получать не только текущие сведения об объекте, но также на любой момент времени.

Для периодического реквизита при создании указывается свойство "Периодичность" из следующих возможных значений:

- в пределах секунды,
- в пределах дня,
- в пределах месяца,
- в пределах квартала,
- в пределах года,
- по позиции регистратора (доступен, если режим записи у регистра — «Подчинение регистратору»).

Обратите внимание, что тип ресурса сведений может быть как примитивный (число, строка, дата, булево), так и ссылочный (СправочникСсылка, ДокументСсылка, ПеречислениеСсылка и т.д.). Более того, в регистре сведений можно хранить даже картинки

и другие неструктурированные сведения, поскольку можно создать ресурс типа "ХранилищеЗначения".

### 3 Курсы валют.

Обратиться к данному справочнику можно при помощи меню Панель разделов/Банк и касса/Панель навигации/Справочники и настройки/Валюты. При первом запуске заполнено всего три позиции валюта (Евро, Российский рубль, Доллар США). Если необходимо загрузить другие валюты, следует на панели инструментов обратиться к командной клавише Подбор из классификатора. Откроется форма подбора валют, из которой можно выбрать нужную валюту необходимую для ведения учета.

Для добавления курса валют вручную необходимо кликнуть по строке Валюты (например открыть карточку Доллар США), обратиться к гиперссылке Курсы валют. Нажать Создать, указать дату, с которой вводится новый курс, а в колонке Курс — значение курса, его кратность, после чего данные будут сохранены в регистре сведений «Курсы валют».

Курсы валют можно загрузить при помощи интернета. После всех настроек следует форму записать и закрыть.

## 1.4. Лекция № 4 (2 часа)

Тема: «Объектная модель»

### 1.4.1. Вопросы лекции:

1 Понятие модуля. Конструкции и ключевые слова языка. Директивы компиляции модуля.

2 Сервисные функции. Синтакс-помощник. Шаблоны текста. Контекстная подсказка. Синтаксический контроль. Форматирование модуля и другие полезные свойства.

### 1.4.2. Краткое содержание вопросов:

1 Понятие модуля. Конструкции и ключевые слова языка. Директивы компиляции модуля.

#### Директивы компиляции

Каждая процедура и функция модуля формы, модуля команды и общего модуля управляемого приложения предваряется директивой компиляции, определяющей среду исполнения данной процедуры. Директива предваряется символом "&". Допустимые директивы:

&НаКлиенте (&AtClient) — определяет клиентскую процедуру (функцию);

&НаСервере (&AtServer) — определяет серверную процедуру (функцию);

&НаСервереБезКонтекста (&AtServerNoContext) — определяет серверную процедуру (функцию), исполняемую на сервере вне контекста формы. Переменные не могут быть внеконтекстными. В таких методах недоступен контекст формы (включая данные формы). Допустимыми являются вызовы только других внеконтекстных методов. При вызове этих методов не выполняется передача данных формы на сервер и обратно. Применение внеконтекстных методов позволяет существенно уменьшить объем передаваемых данных при вызове серверной процедуры из среды клиентского приложения;

&НаКлиентеНаСервереБезКонтекста (&AtClientAtServerNoContext) — определяет процедуру (функцию), исполняемую в модуле формы на клиенте и на сервере, не имеющую

доступа к контексту формы, данным формы, переменным, но имеющую доступ к процедурам и функциям общих модулей – серверных, не глобальных и серверных и клиентских одновременно. Сама процедура (функция) доступна для клиентский, серверных контекстных и неконтекстных процедур и функций модуля формы. Из серверных внеоконтекстных методов формы допускается вызов серверных методов общих модулей;

**&НаКлиентеНаСервере (&AtClientAtServer)** — определяет процедуру (функцию), исполняемую в модуле команды, выполняемую на клиенте и на сервере, имеющую доступ к процедурам и функциям общих модулей – серверных, не глобальных и серверных и клиентских одновременно, не имеющую доступ к переменным. Сама процедура (функция) доступна для клиентских серверных процедур и функций модуля команды.

Клиентская процедура (функция) исполняется в среде клиентского приложения. В такой процедуре доступен клиентский контекст формы и вызовы любых процедур модуля.

Серверная процедура (функция) исполняется в среде серверного приложения. В такой процедуре доступны данные формы, доступен серверный контекст формы и вызовы серверных и серверных внеоконтекстных процедур модуля. При вызове такой процедуры данные формы будут передаваться с клиента на сервер и обратно (по окончанию вызова).

Серверная процедура (функция), исполняемая вне контекста формы, (внеконтекстная) исполняется в среде серверного приложения. В такой процедуре (функции) недоступен контекст формы (включая данные формы). Допустимыми являются вызовы только других внеоконтекстных процедур (функций). При вызове этих процедур (функций) не выполняется передача данных формы на сервер и обратно. Применение внеоконтекстных процедур (функций) позволяет существенно уменьшить объем передаваемых данных при вызове серверной процедуры (функции) из среды клиентского приложения.

Отсутствие директивы компиляции перед процедурой (функцией) означает использование директивы по умолчанию. Директивой по умолчанию является **&НаСервере**. Не допускается использование нескольких директив компиляции перед одной процедурой (функцией). Не допускается наличие одноименных процедур (функций), отличающихся только директивами компиляции.

В модуле управляемой формы допускается использование инструкций препроцессора. Рекомендуется использовать инструкции препроцессора только внутри процедур (функций). Для понимания результата при "пересечении" инструкциями препроцессора границ процедур (функций) следует учитывать, что обработка инструкций препроцессора выполняется до обработки директив компиляции.

В модуле команды предопределенная процедура-обработчик ОбработатьКоманду должна предваряться директивой &НаКлиенте, так как выполнение команды происходит в клиентском приложении.

### Модуль формы

В модуле формы доступны директивы компиляции – &НаКлиенте, &НаСервере, &НаСерверБезКонтекста, &НаКлиентеНаСерверБезКонтекста.

### Модуль команды

В модуле команды доступны директивы компиляции – &НаКлиенте, &НаСервере, &НаКлиентеНаСервере.

### Общий модуль

В общем модуле доступны директивы компиляции – &НаКлиенте, &НаСервере.

2 Сервисные функции. Синтакс-помощник. Шаблоны текста. Контекстная подсказка. Синтаксический контроль. Форматирование модуля и другие полезные свойства.

Начиная осваивать 1С обращаешь внимание в первую очередь на: локальные и глобальные переменные, циклы и условия, функции. Все это очень важно, но упускаем из виду такие вещи как : Синтакс-помощник, контекстная подсказка, отладчик

Игнорируя данные моменты мы сами себя тормозим в освоении 1С: Предприятие. Сейчас я коротко расскажу о них.

#### 1с\_отладка

#### Синтакс-помощник

Синтакс-помощник 1с — это главный инструмент, которым должен овладеть начинающий разработчик. Синтаксис-помощник это часть встроенной справки, предназначена для получения информации об объектах встроенного языка, их свойствах, методах и связанных с ними событиях.

Синтакс-помощник позволяет:

находить описания объектов встроенного языка выбором из дерева объектов;

находить описания объектов встроенного языка поиском по подстроке;

открыв в текстовом редакторе программный модуль, установить курсор на интересующую конструкцию встроенного языка и по контекстному меню или горячей клавише (Ctrl+F1) сразу перейти к описанию этой конструкции в синтакс-помощнике.

#### Контекстная подсказка

Контекстная подсказка 1с позволяет в процессе ввода текста или при нажатии (Ctrl+Пробел) выводится контекстный список, позволяющий выбирать нужное свойство, метод, функцию и т.д., что позволяет быстро и правильно набирать тексты модулей.

Цветовые выделения в контекстной подсказке свойство, метод, функции ....

Красный – Текущий модуль.

Зеленый – Локальный контекст.

Темный — Глобальный контекст.

Не забываем настроить контекстную подсказку в конфигураторе 1с Сервис – Параметры – Модули – Контекстная подсказка.

#### Отладчик

Третий наш друг это Отладчик 1с. Отладчик 1с является встроенным в конфигуратор инструментом. Он помогает проверять работу алгоритмов создаваемых в процессе разработки, производить анализ алгоритмов, выявлять ошибки, отслеживать последовательность выполнения операторов встроенного языка и просматривать значения переменных.

Отладчик можно запускать для разных клиентов, также и подключаться к предмету отладки.

Отладчик позволяет установить на конкретную строку модуля специальный маркер — точку останова, при достижении которой исполнение программного модуля останавливается и управление передается отладчику. Точки останова могут быть безусловными или с условием. При достижении безусловной точки останова исполнение программного модуля останавливается в любом случае.

## **1.5. Лекция № 5 (2 часа)**

**Тема:** «Сервисные функции»

### **1.5.1. Вопросы лекции:**

1 Программное выполнение запроса. Команды формы. Показатели производительности и сценарий "клиент-сервер".

2 Экспортируемые процедуры и общие модули. Параметризируемая команда объекта. Поддержка других языков при создании интерфейса. Механизм объектных блокировок.

### **1.5.2. Краткое содержание вопросов:**

1 Программное выполнение запроса. Команды формы. Показатели производительности и сценарий "клиент-сервер".

Запросы в системе "1С:Предприятие 8" предназначены для выборки информации из базы данных. По сути, запрос — это обращение к системе с просьбой выбрать определенную информацию из базы данных, а часто не только выбрать, но и произвести некоторую обработку: сгруппировать, отсортировать, вычислить. Например, с помощью запроса можно легко узнать объем продаж каждого товара в течение года с детализацией до месяца или выбрать элементы справочников по определенному условию. Заметьте, что изменять данные с помощью запросов в 1С:Предприятии не разрешается.

Язык запросов 1С:Предприятия 8 основан на стандартном SQL, но при этом содержит значительное количество расширений, ориентированных на финансово-экономические задачи, и значительно облегчает разработку бизнес-приложений.

Для работы с запросами в системе "1С:Предприятие 8" предусмотрено несколько специальных объектов, а именно:

1. Запрос
2. Результат Запроса
3. Выборка Из Результата Запроса.

Общая схема выполнения запроса такова:

Создание объекта "Запрос" с нужным текстом запроса на специальном языке запросов.

Установка параметров запроса с помощью метода Установить Параметр.

Выполнение запроса, получение результата.

Получение выборки из результата запроса или выгрузка результата в таблицу значений / дерево значений. Также есть возможность использовать результат запроса для формирования сводной таблицы.

Обработка выборки или таблицы значений (например, перебор строк) и выполнение действий, для которых был нужен запрос, например, вывод области при формировании отчета

2 Экспортируемые процедуры и общие модули. Параметризируемая команда объекта. Поддержка других языков при создании интерфейса. Механизм объектных блокировок.

Общие модули принадлежат всей конфигурации в целом, но, в отличие от модуля приложения (см. ниже), их может быть несколько, например, модули Расчет Налогов, Обработка Документов. Общие модули не могут содержать объявлений переменных, и в

них нет раздела основной программы, таким образом, они состоят только из процедур и функций

Команда - это объект конфигурации, с помощью которого разработчик может описывать действия, предназначенные для выполнения пользователем.

Существуют общие команды - команды, которые не имеют объектной специфики или служат для выполнения действий с объектами, которые не используют стандартные команды.

При работе с объектными данными (справочники, документы, счета и пр.) система «1С:Предприятие» обеспечивает два вида объектных блокировок — пессимистическую и оптимистическую. Они позволяют выполнять целостные изменения объектов при одновременной работе нескольких пользователей.

#### Пессимистическая блокировка

Механизм пессимистической блокировки объектов базы данных предназначен для того, чтобы запретить изменение данных объекта другими сессиями или данным сеансом до тех пор, пока блокировка не будет снята (автоматически или с помощью методов встроенного языка К

В основном механизм пессимистической блокировки используется системой «Предприятие» для блокировки объектов, редактируемых в форме. В то же время разработчик имеет возможность задействовать механизм, используя средства встроенного языка.

Система «1С: Предприятие» использует механизм пессимистической блокировки с помощью расширений форм прикладных объектов. В тот момент, когда пользователь начинает модификацию объекта в форме, расширение формы устанавливает пессимистическую блокировку. Если после этого другой пользователь, например, попытается выполнить редактирование того же объекта, ему будет выдано сообщение о том, что не удалось заблокировать объект. Когда пользователь, редактировавший объект, закроет форму объекта, расширение формы снимет пессимистическую блокировку.

### 1.6. Лекция № 6 (2 часа)

**Тема:** «Справочники»

#### 1.6.1. Вопросы лекции:

1 Перечисления. Иерархия групп. Подчиненные справочники. Табличные части.

2 Расширение функциональности формы. Работа с данными справочника. Реквизиты формы, объекты базы.

3 Создание печатных форм.

#### 1.6.2. Краткое содержание вопросов:

1 Перечисления. Иерархия групп. Подчиненные справочники. Табличные части.

#### Типы справочников

Условно справочники можно разделить на:

простые. Плоский список (нет возможности создавать группы), может иметь дополнительные реквизиты.

иерархические. Простые справочники + имеется возможность создавать группы, указывать подчиненность одних записей другим.

Ранее мы создавали оба типа справочников. В этом уроке рассмотрим работу со справочниками подробнее.

#### Иерархические справочники

##### Виды иерархии:

Иерархия групп и элементов: Элементы могут подчиняться только группам. Пример: Мыло, Шампунь — Моющие средства, Торты, Пирожные — Кондитерские изделия.

Иерархия элементов: Элементы могут быть подчинены другим элементам. Например, можно использовать для создания штатного расписания: Должность «Начальник

отдела», ему подчиняются: «Старший инженер-программист», ему подчинены сотрудники с должностью «Программист».

На первый взгляд в чем разница? зачем 2 вида иерархии? Ответ кроется в принципе самодостаточности. Группа это лишь объединяющий фактор, упрощающий поиск и отображение данных, элемент же самостоятельная единица. При осуществлении деятельности имеет значение что именно мы купили, кто является нашим начальником, т.е. важны элементы. По умолчанию, в полях ввода можно выбрать только элементы, для выбора групп необходимо вносить изменения.

Следует отметить, имеется возможность указания разного состава реквизитов для элементов и групп. Назначение реквизита указывается в свойстве «Использование», может принимать значения: Для элемента, Для группы, Для группы и элемента.

Пример: Справочник.Номенклатура, реквизит «Ответственный» — Использование «Для группы», реквизит «ШтрихКод» — «Для элемента». Иначе говоря, для группы товаров можно указать ответственного, который будет следить за правильностью заполнения товаров. ШтрихКод — уникален для каждого товара, поэтому необходим только для элемента.

### **Подчиненные справочники**

Отдельно стоит отметить возможность создания подчиненных справочников. Общий принцип можно понять на основе примера: Справочник.Страны (владелец), Справочник.Города (подчиненный). Платформа предоставляет возможность указания нескольких справочников-владельцев у подчиненного справочника. Это удобно для хранения однотипных сущностей подчиненных разным функциональностям, например: Справочник.БанковскиеСчета может быть подчинен справочникам: Контрагенты (Поставщики, Покупатели), Фирмы (свои компании).

2 Расширение функциональности формы. Работа с данными справочника. Реквизиты формы, объекты базы.

У большинства форм существует такое понятие, как «основной реквизит формы». Он выделен полужирным на закладке «Реквизиты» формы, или его можно просмотреть в свойствах самой формы.

Основной реквизит формы определяет основную функциональность формы (так сказать ее основное предназначение). Функциональность формы можно неограниченно расширять (ограничением является здравый смысл). Расширять функциональность формы можно за счет добавления реквизитов (расширяя этим понятие «данные формы»). Вернемся к форме элемента справочника «Физические лица». На закладке «Реквизиты» добавьте новый реквизит и определите у него тип <СправочникСписок.Номенклатура>. Имя реквизита определите как «СписокНоменклатуры».

оздайте в элементе управления «Панель» (в диалоге формы) новую страницу. Добавьте в диалог элемент управления «табличное поле» и через свойство «Данные» свяжите с реквизитом «СписокНоменклатуры».

Остается добавить элемент управления «командная панель» и настроить свойства, как показано на рисунке.

В результате на одной из страниц формы элемента справочника «Физические лица» мы можем работать со списком элементов справочника «Номенклатура».

### **3 Создание печатных форм.**

Для формирования печатных форм в системе используются два объекта: «Макет» и «Табличный документ». Следует отметить, что «Макет» является объектом метаданных, чаще всего содержащим внутри себя «Табличный документ».

Табличный документ, как и любая другая таблица, представляет собой совокупность ячеек прямоугольной формы, организованных в строки и колонки. Каждая колонка и каждая строка имеет свой уникальный номер. Столбцы в строках могут иметь индивидуальную ширину (разную в различных строках).

Табличный документ представляет «готовую» печатную форму и используется самостоятельно. Элемент управления типа «Поле табличного документа» может располагаться в форме. В этом режиме в табличный документ можно вставить другие элементы управления.

Макет предназначен для определения «кирпичиков», на основании которых формируется готовая выходная форма. Проектирование макета заключается в «рисовании» этих составных частей: именованных областей. Макет располагается внутри конфигурации.

## **1.7. Лекция № 7 (2 часа)**

**Тема:** «Создание документов. Доступ к данным документа»

### **1.7.1. Вопросы лекции:**

1 Доступ к данным документа. Модуль объекта. Создание объектов копированием. Журналы документов.

2 Регистры сведений. Создание регистра сведений. Работа с данными регистра. Форма списка регистра.

3 Режим записи "Подчинение регистратору". Планы видов характеристик. Функциональные опции. Учетные объекты.

### **1.7.2. Краткое содержание вопросов:**

1 Доступ к данным документа. Модуль объекта. Создание объектов копированием. Журналы документов.

Модули – это те объекты, где содержится программный код. В Платформе существует достаточно большое количество видов модулей, каждый из которых имеет свое предназначение и особенности.

Любая строка кода должна находиться в каком-либо модуле. Различают модули общего назначения и модули объекта. Некоторые модули могут быть скомпилированы как на Клиенте, так и на Сервере, а некоторые только на Сервере. Модуль может состоять из нескольких разделов.

В разделе описания переменных описываются локальные переменные данного модуля, которые впоследствии могут быть использованы в любой процедуре. Внутри каждой процедуры можно обращаться к переменной модуля. Кроме того, внутри самой процедуры может быть еще одно объявление переменной с таким же именем. Это будет локальная переменная данной процедуры.

Несмотря на одинаковое название, это две разные переменные: одна используется внутри конкретной процедуры, а другая – вне ее.

Данные модули характерны для справочников, документов, планов видов расчетов, планов счетов и многих других объектов. Модуль объекта предназначен для обработки стандартных событий.

Например, событие на ввод элемента справочника, событие на запись элемента, удаление, проведение документа и т.д. В принципе, событие записи существует и в Модуле формы. Но событие записи в Модуле формы возникает в процессе интерактивной записи, при работе с конкретной формой. Событие записи в Модуле объекта будет выполняться при любой записи из любой формы данного объекта.

Кроме того, если объект записывается программно, в этом случае будет срабатывать событие модуля объекта. В событии записи Модуля объекта можно встраивать все проверки на корректность записываемых данных, так как эта процедура будет отрабатывать в момент абсолютно любой записи.

2 Регистры сведений. Создание регистра сведений. Работа с данными регистра. Форма списка регистра.

Объект конфигурации "Регистр сведений" является прикладным объектом и предназначен для описания структуры хранения данных в разрезе нескольких измерений. На основе этого объекта платформа создает в базе данных информационную структуру, в которой может храниться произвольная информация "привязанная" к набору измерений.

Стоимость услуг имеет особенность меняться со временем, и может сложиться такая ситуация, что когда нам потребуется внести изменения или уточнения в один из ранее проведенных документов "ОказаниеУслуги", мы не сможем получить правильную стоимость услуги, поскольку в реквизите справочника будет храниться последнее введенное значение. Поэтому для хранения стоимости услуг лучше использовать регистр сведений.

В конфигураторе создадим новый объект конфигурации "Регистр сведений". Назовем его "Цены". Установим периодичность этого регистра в пределах секунды. Перейдем на закладку "Данные" и создадим измерение регистра "Номенклатура" с типом СправочникСсылка.Номенклатура. Укажем, что это измерение будет ведущим. После этого создадим новый ресурс "Цена", тип Число, длина 15, точность 2, неотрицательное.

3 Режим записи "Подчинение регистратору". Планы видов характеристик. Функциональные опции. Учетные объекты.

#### Регистр сведений (Information Register)

Регистры сведений - это прикладные объекты конфигурации. Они позволяют хранить в прикладном решении произвольные данные в разрезе нескольких измерений. Например, в регистре сведений можно хранить курсы валют в разрезе валют, или цены предприятия в разрезе номенклатуры и типа цен.

#### Структура

Информация в регистре сведений хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.

Измерения регистра описывают разрезы, в которых хранится информация, а ресурсы регистра непосредственно содержат хранимую информацию.

#### Периодичность

Одной из возможностей регистра сведений является хранение данных не только в разрезе указанных измерений, но и в разрезе времени.

В этом случае к каждой записи регистра будет добавляться поле Период, хранящее дату, которой были внесены записи в регистр. Использование периодичности регистра сведений позволяет не просто хранить статические данные, но и отслеживать их изменение во времени.

#### Подчинение регистратору

Внесение изменений в регистр сведений может выполняться как вручную, так и при помощи документов. В случае, когда изменения в регистр сведений вносятся с помощью документов, к каждой записи регистра добавляется специальное поле, в котором хранится информация о регистраторе - документе, с которым связана эта запись.

### 1.8. Лекция № 8 (2 часа)

Тема: «Источники данных. Структура запроса (описание запроса) »

#### 1.8.1. Вопросы лекции:

1 Структура запроса (описание запроса). Использование конструктора запросов. Особенности работы с виртуальными таблицами.

2 Построение запросов по нескольким таблицам. Работа с временными таблицами. Использование предопределенных данных. Пакетные запросы.

#### 1.8.2. Краткое содержание вопросов:

1 Структура запроса (описание запроса). Использование конструктора запросов. Особенности работы с виртуальными таблицами.

Виртуальная таблица "РегистрРасчета.<ИмяРегистра>.ДанныеГрафика" определена для тех регистров расчета, которые поддерживают период действия. При конфигурировании с таким регистром расчета необходимо связать непериодический регистр сведений который и будет поставлять информацию о графике.

Для более удобного получения данных графика в системе определена виртуальная таблица, которая помимо прочих полей содержит виртуальные поля:

<Имя ресурса графика>БазовыйПериод;  
<Имя ресурса графика>ПериодДействия;  
<Имя ресурса графика>ПериодРегистрации;  
<Имя ресурса графика>ФактическийПериодДействия.

То есть для каждого числового ресурса регистра сведений, назначенного регистру расчета в качестве графика, можно получить для строк регистра расчета его сумму с учетом базового периода строки регистра расчета, периода действия, периода регистрации и фактического периода действия.

При построении виртуальной таблицы данных графика происходит соединение таблиц регистра расчета, регистра сведений, а в случае получения поля "<Имя ресурса графика>ФактическийПериодДействия" - еще и таблицы фактического периода действия регистра расчета. Так как данные для всех четырех перечисленных полей получаются путем соединения с таблицей регистра сведений по разным условиям, это значит, что в общем случае будет выполнено четыре соединения с таблицей регистра сведений.

При получении виртуальных таблиц система старается действовать оптимально, в частности, выполняется только то количество соединений, которое необходимо для получения полей виртуальной таблицы перечисленных в разделе "ВЫБРАТЬ".

Следствием этого является то, что при составлении текста запроса не следует выбирать поля данной виртуальной таблицы "про запас". Эта рекомендация может показаться общей, но следует помнить, что, как правило, получение "ненужного" поля в запросе к реальным таблицам или другим виртуальным таблицам приводит к некоторому увеличению сетевого трафика и, в общем случае, весьма незначительному снижению производительности. Ясно, что получение в запросе пяти полей регистра мало чем отличается от получения в запросе еще одного, шестого поля, если речь не идет о "больших" полях. Но в случае виртуальной таблицы данных графика появление еще одного виртуального поля из ряда перечисленных выше изменяет ситуацию принципиально. Время выполнения запроса может измениться в разы. То есть рекомендация составлять запросы оптимально по части числа выбираемых полей принимает особый смысл для данной виртуальной таблицы.

Следует также заметить, что принципиальное влияние на производительность запроса к виртуальной таблице данных графика оказывает не столько число виртуальных полей, сколько число их различных "видов".

2 Построение запросов по нескольким таблицам. Работа с временными таблицами. Использование предопределенных данных. Пакетные запросы.

Имея некоторый опыт разработки, не раз приходилось видеть, насколько нерационально и громоздко порой строятся запросы начинающими разработчиками к информационным базам 1С. Существует достаточно простой и удобный механизм временных таблиц, при использовании которого можно разбить большой и плохо читаемый запрос на несколько небольших. Можно так же по некоторому алгоритму сформировать таблицу значений, поместить ее во временную таблицу и выполнить по ней запрос.

В данной публикации на примере процедуры проведения документа Реализация, не вникая в детали алгоритма проведения, описываются некоторые приемы использования временных таблиц в запросах к информационной базе на платформе 8.2. Автор публикации не претендует на полноту освещения вопроса или методическую безупречность.

Рассмотрим пример работы с временными таблицами в конкретном запросе.

1. Создается таблица значений с указанием типов колонок. Типы колонок для таблицы значений, которая будет использоваться в запросе, обязательно должны быть определены (в тексте программного модуля номера и краткое описание разделов указаны в комментарии).

2. Таблица значений заполняется по некоторому произвольному алгоритму. В данном примере она дублирует движения документа.

3. Копируются отдельные таблицы значений для использования в условиях запроса.

4. Создается объект менеджер временных таблиц. Менеджер временных таблиц предназначен для хранения информации о временных таблицах, которые участвуют в запросе. При удалении объекта временные таблицы, находящиеся в нем, удаляются. Для временных таблиц, участвующих в одном запросе, должен использоваться единый менеджер временных таблиц.

5. Свойствам МенеджерВременныхТаблиц всех запросов, где участвуют временные таблицы, присваивается значение созданного ранее менеджера временных таблиц.

6. Формируются временные таблицы. При необходимости можно сформировать требуемое количество временных таблиц путем выполнения запросов к заполненной ранее таблице значений. Важной особенностью является то, что таблица значений непосредственно не может участвовать в запросах, где используются соединения с другими таблицами или агрегатные функции (Сумма, Среднее, Максимум, Минимум и др.). Для этого предварительно следует простым запросом, где в качестве параметра используется ссылка на таблицу значений, поместить ее во временную таблицу.

7. Временные таблицы используются в запросе. В приведенном запросе они используются в условии виртуальной таблицы регистра накопления в виде небольших вложенных запросов, а также временная таблица связана внутренним соединением с виртуальной таблицей регистра.

## **1.9. Лекция № 9 (2 часа)**

**Тема:** «Отчеты. Рабочий стол. Критерии отбора»

### **1.9.1. Вопросы лекции:**

1 Критерии отбора. Обработка заполнения данных и установка значений по умолчанию.

2 Хранилище значений (работа с изображениями). Механизм полнотекстового поиска. Регламентные задания. Бизнес-процессы и задачи.

### **1.9.2. Краткое содержание вопросов:**

1 Критерии отбора. Обработка заполнения данных и установка значений по умолчанию.

В составе метаданных 1С:Предприятия 8 существует объект Критерий отбора.

Этот объект, в отличие от большинства прикладных объектов конфигурации, не описывает каких-либо самостоятельных данных, которые будут храниться в информационной базе 1С:Предприятия. Критерий отбора предназначен для того, чтобы описать условия отбора по некоторым объектам. Например, можно указать, что критерий отбора строится по типу данных СправочникСсылка.Номенклатура, и в этот критерий входят документы "ПоступлениеТоваровУслуг" и "РеализацияТоваровУслуг". Конечно, все отборы мы можем делать с помощью запроса.

Особенностью (преимуществом) критерия отбора является то, что он может строиться в том числе по реквизитам табличных частей. Таким образом мы можем очень просто настроить отбор так, чтобы в его результат попадали лишь те документы, в табличной части которых есть интересующий нас элемент справочника "Номенклатура".

Критерий отбора является прикладным объектом, для него, в частности, можно создать форму, которая будет отображаться в пользовательском режиме. Однако находится он в ветке "Общие" из-за того, что имеет узкоспециализированное назначение. Особенностью Критерия отбора является то, что он не входит в глобальный командный интерфейс.

2 Хранилище значений (работа с изображениями). Механизм полнотекстового поиска. Регламентные задания. Бизнес-процессы и задачи.

Механизм полнотекстового поиска в 1С позволяет быстро находить необходимую для пользователя информацию. Данный вид поиска особенно эффективен, если информационная база располагает большим объемом информации, а также точно не известно, где находятся интересующие пользователя данные или как часто бывает, их

точное название не известно. Для того чтобы открыть окно управления полнотекстовым поиском необходимо выполнить следующее: пункт меню Операции Управление полнотекстовым поиском.

Обновить индекс – Создание индекса/Обновление индекса; Очистить индекс – обнуление индекса(рекомендуется после обновления всех данных); пункт Разрешить слияние индексов – отвечает за слияние основного и дополнительного индекса.

Полнотекстовый поиск осуществляется при помощи полнотекстового индекса. При отсутствии индекса полнотекстовый поиск как таковой не возможен. Для того чтобы поиск имел результат, все необходимые данные должны быть включены в полнотекстовый индекс. Если пользователем введены в базу новые данные, их следует включить в рассматриваемый индекс, иначе они не будут участвовать в поиске. Чтобы этого избежать, необходимо обновлять полнотекстовый индекс. При обновлении система анализирует только определенные типы данных: Стока, Данные ссылочного типа (ссылки на документы, справочники), Число, Дата, ХранилищеЗначения. Если пользователь не имеет прав доступа к определенной информации, то он не сможет увидеть ее в результатах поиска. Следует также помнить и о том, что в свойствах объектов, по которым будет происходить поиск должно быть установлено значение Полнотекстовый Поиск – Использовать, которое задано по умолчанию.

## 2. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

### **2.1. Лабораторная работа №1 (2 часа).**

**Тема:** «Введение в Регистры».

#### **2.1.1. Вопросы к занятию:**

1. Информационная база
2. Виды регистров.

#### **2.1.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Метаданные...
  1. термин, который в буквальном смысле означает «данные о данных» или «информация об информации»
  2. является объектами бухгалтерской программы, предназначенными для ввода и сохранения в базе данных информации о фактах хозяйственной деятельности, происходящих на предприятии
  3. предназначены для регистрации информации о введенных документах
  4. объект программы, обеспечивающий формирование некоторой выходной (печатной) формы документа на основе информации, содержащейся в базе учетных данных
2. Вся информация, с которой приходится оперировать в процесс конфигурирования является...
  1. метаинформацией
  2. пиктограмма состояния счета
3. Конфигурация это...
  1. оперативно доступные для выбора функций
  2. набор описаний информационных объектов бухгалтерской программы
4. Из скольких частей состоит информационная база, с которой работает программа «1С: Бухгалтерия»
  1. 1
  2. 2
  3. 3
  4. 4

### **2.2. Лабораторная работа №2 (2 часа).**

**Тема:** «Соединение источников в запросе».

#### **2.2.1. Вопросы к занятию:**

1. База метаданных
2. Пакетный запрос

#### **2.2.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. База метаданных...
  1. создается и ведется в режиме конфигурирования. Она содержит информацию с описанием текущей конфигурации программы
  2. работа с ней осуществляется в режиме «1С: Предприятие», однако, организация этой базы данных определяется текущим состоянием базы метаданных, в результате чего работа с ней невозможна в отрыве от конкретной базы метаданных
2. База данных бухгалтерского отчета...
  1. создается и ведется в режиме конфигурирования. Она содержит информацию с описанием текущей конфигурации программы

2. работа с ней осуществляется в режиме «1С: Предприятие», однако, организация этой базы данных определяется текущим состоянием базы метаданных, в результате чего работа с ней невозможна в отрыве от конкретной базы метаданных
3. Информационная база «1С: Бухгалтерия (демо)»
  1. содержит учетные данные, представляющие собой контрольный (демонстрационный) пример ведения учета на условном предприятии
  2. представляет собой пустую базу данных, которую можно использовать для ведения бухгалтерского учета на предприятии
  3. используется, когда на предприятии ведется достаточно простой учет и не требуется привлечения сложных средств автоматизации учета, заложенных в типовую конфигурацию
4. Информационная база «1С: Бухгалтерия. Типовая конфигурация»...
  1. содержит учетные данные, представляющие собой контрольный (демонстрационный) пример ведения учета на условном предприятии
  2. представляет собой пустую базу данных, которую можно использовать для ведения бухгалтерского учета на предприятии
  3. используется, когда на предприятии ведется достаточно простой учет и не требуется привлечения сложных средств автоматизации учета, заложенных в типовую конфигурацию

### **2.3. Лабораторная работа №3 (2 часа).**

**Тема:** «Формы и редактор форм».

#### **2.3.1. Вопросы к занятию:**

1. виды форм
2. форма констант
3. форма документа
4. формы списков

#### **2.3.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Объект конфигурации Константа:

- 1) предназначен для хранения предположительно не изменяющейся информации
- 2) для хранения значений во времени необходимо установить признак периодичности
- 3) в платформе не поддерживает признак периодичности, необходимо использовать периодический регистр сведений
- 4) верны варианты предложенных ответов 1 и 2
- 5) верны варианты предложенных ответов 1 и 3

2. Какие виды иерархии существуют в системе «1С:Предприятие 8.2»?

- 1) иерархия групп
- 2) иерархия элементов
- 3) иерархия групп и элементов
- 4) верны ответы 1 и 3
- 5) верны ответы 2 и 3
- 6) верны все указанные ответы

3. Какие варианты подчинения существуют в системе «1С: Предприятие 8.2»?

- 1) группам
- 2) элементам
- 3) группам и элементам
- 4) верны ответы 1 и 3
- 5) верны ответы 2 и 3
- 6) верны все указанные ответы

4. В какое значение можно установить свойство «Серии кодов» объекта конфигурации «Справочник»?
- 1) во всем справочнике
  - 2) в пределах подчинения
  - 3) в пределах подчинения владельцу
  - 4) верны все указанные ответы
  - 5) верны ответы 1 и 2

## **2.4. Лабораторная работа №4 (2 часа).**

**Тема:** «Объектная модель».

### **2.4.1. Вопросы к занятию:**

1. Программные средства
2. Понятие модуля
3. Конструкции и ключевые слова языка
4. Директивы компиляции модуля

### **2.4.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. На закладке "Связи" конструктора запросов можно определить:
  - 1) Соединение таблиц-источников данных и связи между ними
  - 2) Объединение таблиц-источников данных и связи между ними
  - 3) Связи между полями таблицы, получаемой в результате выполнения запроса
  - 4) Связи между полями таблицы-источника данных и таблицы, получаемой в результате выполнения запроса
2. При соединении таблиц-источников данных в конструкторе запросов можно назначить:
  - 1) соединение без указания условия связи при соединении таблиц-источников
  - 2) соединение с указанием условия связи, причем это условие может быть одно
  - 3) соединение с указанием условия связи, причем это условие может быть простое
  - 4) количество соединений с указанием условий связи (условия могут простыми или произвольными)
3. Можно ли в процессе отладки увидеть в окне "Выражение" содержимое таблицы значений целиком?
  - 1) нельзя, окно дает возможность увидеть состав и название колонок таблицы значений
  - 2) нельзя, но можно увидеть данные построчно в цикле перебора строк таблицы значений
  - 3) можно, поскольку содержимое таблицы значений будет сразу видно в окне "Результат"
  - 4) можно, но только посредством команды "Показать в отдельном окне"
4. Для чего предназначен построитель отчета?
  - 1) Для динамического создания отчета как программными, так и интерактивными средствами
  - 2) Для динамического создания отчета программными средствами
  - 3) Для динамического создания отчета интерактивными средствами
  - 4) Нет верного ответа среди предложенных вариантов задания

## **2.5. Лабораторная работа №5 (2 часа).**

**Тема:** «Сервисные функции».

### **2.5.1. Вопросы к занятию:**

1. Синтаксис-помощник
2. Шаблоны текста
3. Контекстная подсказка

### **2.5.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Что необходимо выполнить в процедуре "ОбработкаПроведения" в модуле документа для отмены проведения?

- 1) СтатусВозврата(О)
- 2) Отказ = Истина
- 3) Отказ = Ложь
- 4) Режим = Ложь

2. Какая процедура выполняет обработку события ввода документа "на основании"?

1) "ОбработкаЗаполнения" с формальным параметром "Основание" произвольного типа

- 2) "ВводНаОсновании" с формальным параметром "Основание" типа "Документ"
- 3) "ОбработкаЗаполнения" с формальным параметром "Основание" типа "Документ"

4) Ввода документа не требует дополнительных методов обработки, т.к. это процесс автоматический

3. Какой метод документа обеспечивает вызов события проведения документа?

1) Записать(РежимЗаписиДокумента.Проведение), где Записать - метод объекта документа

- 2) Провести(), где Провести - метод объекта документа

3) Записать(РежимЗаписиДокумента.Проведение), где Записать - метод формы документа

- 4) Провести(), где Провести - метод формы документа

4. Какой метод документа обеспечивает вызов события отмены проведения документа?

1) Записать(РежимЗаписиДокумента.ОтменаПроведения), где Записать - метод объекта документа

- 2) ОтменитьПроведение(), где ОтменитьПроведение - метод объекта документа

- 3) ОтменитьПроведение(), где ОтменитьПроведение - метод формы документа

4) Записать(РежимЗаписиДокумента.ОтменаПроведения), где Записать - метод формы документа

## **2.6. Лабораторная работа №6 (2 часа).**

*Тема: «Обработчики событий формы».*

### **2.6.1. Вопросы к занятию:**

1. Программное выполнение запроса

2. Команды формы

### **2.6.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Какие типы панелей инструментов использует программа 1С:Бухгалтерия?

- 1. жестко связанные с окном
- 2. плавающие
- 3. многомерные
- 4. пиктораммные

2. Панели, жестко привязанные к окну...

- 1. содержат пиктограммы, предназначенные для работы с информацией, расположенной только в этом окне
- 2. содержат пиктограммы для вызова команд общего применения
- 3. представляют собой объект программы, предназначенной для хранения условно-постоянной информации

4. представляют собой поименованные списки, содержащие некоторой набор значений
3. Плавающие панели инструментов...
  1. содержат пиктограммы, предназначенные для работы с информацией, расположенной только в этом окне
  2. содержат пиктограммы для вызова команд общего применения
  3. представляют собой объект программы, предназначенной для хранения условно-постоянной информации
  4. представляют собой поименованные списки, содержащие некоторой набор значений
4. Что можно делать с плавающими панелями инструментов?
  1. изменять их форму
  2. перемещать
  3. располагать в любом удобном месте

## **2.7. Лабораторная работа №7, 8, 9 (6 часов).**

**Тема:** «Справочники».

### **2.7.1. Вопросы к занятию:**

1. линейные справочники
2. иерархические справочники

### **2.7.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Для каких компонент существуют сетевые ключи защиты?
  - 1) клиентское приложение
  - 2) типовое тиражное решение
  - 3) сервер 1С:Предприятие
  - 4) правильны ответы 1 и 3
  - 5) правильны ответы 1 и 2
  - 6) верны все варианты
2. Сколько конфигураций обязательно должно существовать в прикладном решении
  - 1) одна
  - 2) две
  - 3) три
  - 4) сколько угодно
3. Из чего состоит конфигурация?
  - 1) объекты конфигурации
  - 2) объекты встроенного языка
  - 3) объекты информационной базы
  - 4) верны ответы 1 и 3
  - 5) верны все варианты
4. Какие объекты используются в алгоритмах?
  - 1) объекты конфигурации
  - 2) объекты встроенного языка
  - 3) объекты информационной базы
  - 4) верны ответы 1 и 3
  - 5) верны все варианты

## **2.8. Лабораторная работа №10, 11, 12 (6 часов).**

**Тема:** «Создание документов. Доступ к данным документа».

### **2.8.1. Вопросы к занятию:**

1. Доступ к данным документа

2. Модуль объекта
3. Создание объектов копированием
4. Журналы документов

### **2.8.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Когда в Журнале документ считается выбранным?
  1. Когда курсор стоит на строке с записью о документе в любой графе журнала
  2. Когда Добавление, удаление и копирование строк в табличной части документа
  3. Когда получена краткая справка по элементам формы документа
  4. Когда изменены вводимые данные документа
2. Что определяет Интервал видимости?
  1. Документы, за какой период времени будут видны при просмотре журнала.
  2. Изменение вводимых данных документа
  3. Добавление, удаление и копирование строк в табличной части документа
  4. Сохранение изменений
  5. Переход к описанию документа
  6. Получение краткой справки по элементам формы документа
  7. Пользователю управлять вводимыми данными, изменения параметры обработки данных алгоритмом документа
3. Как устанавливается интервал видимости?
  1. Автоматически
  2. Вручную
4. Как изменить интервал видимости записей журнала?
  1. Меню Действия – команда Интервал
  2. Выбрать в меню Журналы соответствующий вид
  3. В меню Операции главного меню программы команду Журналы, указав имя журнала в списке
  4. Выбрать в меню Действия главного меню программы команду Ввести
  5. Выбрать в журнале документ, затем выбрать в меню Действия главного меню программы команду Копировать
  6. Выбрать в журнале документ, затем выбрать в меню Действия главного меню программы команду Удалить

## **2.9. Лабораторная работа №13, 14, 15 (6 часов).**

**Тема:** «Источники данных. Структура запроса (описание запроса)».

### **2.9.1. Вопросы к занятию:**

1. Структура запроса
2. Построение запросов по нескольким таблицам

### **2.9.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Что такое код локализации?
  - 1) Стока, состоящая из кода языка и кода страны и идентифицирующая регион мира
  - 2) Код программного продукта (указывается на регистрационной анкете, документации из комплекта поставки)
  - 3) Параметр форматирования строки для преобразования
  - 4) Нет верного ответа среди предложенных вариантов задания

2. При настройке ограничения доступа к данным в качестве значений, по которым производится ограничения доступа к данным, могут использоваться:
  - 1) Только значения параметров сеанса
  - 2) Только данные из таблиц
  - 3) Значения параметров сеанса и данные из таблиц (запросов)
  - 4) Только значения имеющие типы: Число, Стока, Булево, Дата
 3. Для повышения скорости выполнения запроса необходимо:
  - 1) Задавать параметры большинства реальных таблиц
  - 2) Задавать параметры большинства виртуальных таблиц
  - 3) Вместо параметров у реальной (виртуальной) таблицы использовать отбор, заданный конструкцией "ГДЕ"
    4. На закладке "Условия" конструктора запросов отдельная строка списка условий может быть сформирована:
      - 1) С помощью двойного клика ЛКМ на нужном поле в списке доступных полей
      - 2) Путем переноса нужного поля в список, используя технологию Drag & Drop
      - 3) Нажать кнопку "Добавить". Если условие произвольное, то текст условия ввести "вручную"
    - 4) Вызвав контекстное меню, выбрать "Добавить". Возможно использование произвольного выражения
  - 5) Верны все вышеперечисленные ответы задания

## **2.10. Лабораторная работа №16, 17, 18 (6 часов).**

**Тема:** «Отчеты. Рабочий стол. Критерии отбора».

### **2.10.1. Вопросы к занятию:**

1. Критерии отбора

2. Обработка заполнения данных и установка значений по умолчанию

### **2.10.2. Краткое описание проводимого занятия:**

*Задания для проведения текущего контроля успеваемости*

1. Может ли пользователь изменить состав отображаемых полей отчета?

1) Только в модуле формы отчета, описав их языком запросов

2) Может, используя настройки построителя отчета, выбирая их из числа полей

3) Нет, данная функция не предусмотрена в платформе

4) Нет верного ответа среди предложенных вариантов задания

2. Настройки построителя отчетов включают:

1) Группировки, ВыбранныеПоля, ДоступныеПоля, ИзмеренияКолонки, ИзмеренияСтроки, Отбор, Порядок, Фильтр

2) ВыбранныеПоля, ИзмеренияКолонки, ИзмеренияСтроки, Показатели, УсловноеОформление, Дополнительные поля, Фильтр

3) ВыбранныеПоля, ДоступныеПоля, ИзмеренияКолонки, ИзмеренияСтроки, Отбор, Порядок, УсловноеОформление

4) Нет верного ответа среди предложенных вариантов задания

3. Для формирования табличного документа построитель отчета использует...

1) макет, созданный разработчиком

2) макет, генерируемый автоматически

3) верно первое и второе

4) Нет верного ответа среди предложенных вариантов