

Лабораторные работы по курсу «Хранилища данных» для студентов ИВМиИТ

Разработали:
проф. Ш.Т.Ишмухаметов,
ст.преп. Р.Г.Рубцова

Содержание:

Лабораторная работа № 1. Работа в Oracle Database Express Edition.....	1
Лабораторная работа № 3. Работа с базой данных с использованием утилиты SQLplus.	8
Лабораторная работа № 4. Экспорт баз данных.....	10
Лабораторная работа 5. Выполнение расчетов с использованием программирования в среде Visual Basic for Applications (VBA).....	12
Лабораторная работа № 6. Формирование хранилища данных из разных источников.....	14
Лабораторная работа № 7.....	22
Оценка эффективности работы филиалов	22

Лабораторная работа № 1. Работа в Oracle Database Express Edition

Введение.

Пакет Oracle Database 10g Express Edition (Oracle Database XE) является свободно распространяемой версией СУБД Oracle. Работа с СУБД выполняется с помощью интуитивно понятного WEB-интерфейса браузера. С помощью этого интерфейса можно выполнять все основные операции по созданию таблиц баз данных, установлению связей между таблицами, вводу данных, созданию представлений, отчетов, администрированию пользователей.

Рассмотрим основные правила работы с этим пакетом.

1. Установка Oracle Database Express Edition. Для этого запускаем дистрибутив пакета - файл **OracleXE.exe** (размер 216 Мб). Указываем место для установки и соглашаемся с остальными опциями. На одной из страниц установки потребуется указать пароль, с которым будем осуществляться доступ к серверу СУБД. Укажем в качестве пароля слово **oracle**.

2. После установки в меню **Пуск\Программы** появится раздел **Oracle Database 10g Express Edition**, содержащий подразделы

Get Help – помощь.

Backup Database – резервирование БД.

Get Started – вызов справки по Oracle Express.

Go To Database Home Page – домашняя страница.

Run SQL Command Line – работа с БД с помощью команд SQL.

Start Database – запуск сервера.

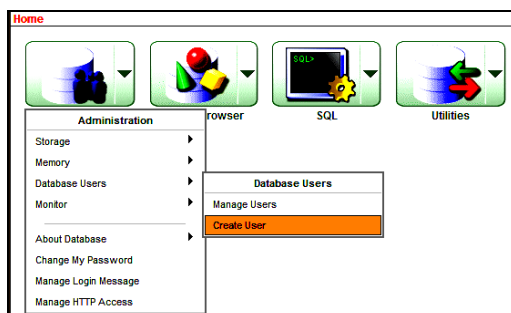
Stop Database – остановка сервера.

3. Идем к домашней странице «**Go To Database Home Page**». При этом открывается окно *Database Login*:

4. Первоначально на сервере создан один пользователь **system** с паролем, указанный при установке (**oracle**), поэтому заходим с этими логином и паролем. Для проверки *Имени пользователя* кликните по ссылке: «**Click here to learn how to get started**».

В пункте 2 будет указан **Username**, который надо ввести.

5. Далее нажмем на кнопку **Administration**, выберем раздел **Database Users** → **Create Users**:



6. Создадим нового пользователя с вашим именем и паролем, например:

7. После заполнения формы нажимаем на кнопку **Create** (Создать). Далее заканчиваем сеанс, нажав ссылку **Logout** в правом верхнем углу экрана.



8. Выбираем **Object Browser**. Далее выполняем команду: **Create** → **Table**.

9. Открывается меню для создания таблицы. В нашем примере имя таблицы: *Student_Dana*. Заполним пол **я:Column Name** (Имя поля), **Type** (Тип данных), **Scale** (Размерность) таблицы (см.рис):

Home > Object Browser

Table

Create Table Cancel Next >

Table Name: Student_Data Preserve Case

Column Name	Type	Precision	Scale	Not Null	Move
Student_ID	NUMBER			<input checked="" type="checkbox"/>	▼ ▲
Name	VARCHAR2		20	<input checked="" type="checkbox"/>	▼ ▲
Address	VARCHAR2		30	<input checked="" type="checkbox"/>	▼ ▲
Telephone	VARCHAR2		10	<input type="checkbox"/>	▼ ▲
	- Select Datatype -				▼ ▲

10. Далее нажимаем **Next**. Открывается форма для создания Ключа (**Primary Key**): Выбираем **Populated from a new sequence**, задаем ключевое поле (в нашем случае **Student_ID (Number)**):

Home > Object Browser

Table

Primary Key Cancel < Previous Next >

Table name: STUDENT_DATA

Primary Key: No Primary Key
 Populated from a new sequence
 Populated from an existing sequence
 Not populated

Primary Key Constraint Name: STUDENT_DATA_PK

Sequence Name: STUDENT_DATA_SEQ

Primary Key: STUDENT_ID(NUMBER)

11. Нажимаем кнопку **Next**. Открывается форма для задания внешнего ключа (**Foreign Key**). Если внешний ключ не задается, нажимаем **Далее**.

12. Открывается форма для создания *Ограничений* (**Constraints**). При отсутствии ограничений нажимаем **Finish**.

13. Следующая форма сообщает о том, что пользователем создана таблица. Нажимаем кнопку **Create**.

14. Открывается диалоговое окно построителя таблицы: **STUDENT_DATA**. В левой части приводится список всех созданных таблиц. В правой части: мастер создания и изменения таблицы, который содержит следующие возможности:

- Добавить столбец,
- Изменить столбец,
- Переименовать столбец,
- Удалить столбец,
- Переименовать таблицу,
- Копировать таблицу,
- Удалить таблицу.

Чтобы заполнить таблицу, выбираем вкладку **Data**, кнопку **Insert Row**.

В появившуюся форму заносим данные:

STUDENT_DATA	
Create Row <input type="button" value="Cancel"/> <input type="button" value="Create"/> <input type="button" value="Create and Create Another"/>	
Table:	STUDENT_DATA
* Student Id	1
* Name	Sergey
* Address	Kazan, Polevaya, 7
Telephone	2456789

Далее нажимаем **Create**, затем кнопку **Insert Row**. Заполняем данные на следующего студента. В результате заполнения полей таблицы появляется список всех студентов:

STUDENT_DATA				
				Row created.
Table	Data	Indexes	Model	Constraints
Grants	Statistics	UI Defaults	Triggers	Dep
Query	Count Rows	Insert Row		
EDIT	STUDENT_ID	NAME	ADDRESS	TELEPHONE
	1	Sergey	Kazan, Polevaya, 7	2456789
	2	Ivan	Moscow, Lenina, 12	1356780
	3	Petr	Kazan, Baumana, 145	2929240
	4	Elena	Kazan, Baumana, 40	2924878
row(s) 1 - 4 of 4				

15. Самостоятельно создайте таблицу **SUBJECT (Предметы)**, содержащую следующие поля:

Column Name	Data Type	Nullable	Default	Primary Key
SUBJECT_ID	NUMBER	No	-	-
SUBJECT_NAME	VARCHAR2(25)	Yes	-	-
				1 - 2

Внесите в таблицу следующие данные:

EDIT	SUBJECT_ID	SUBJECT_NAME
	1	BD
	2	Math
	3	Inform
row(s) 1 - 3 of 3		

16. Создайте таблицу **EXAMS (Экзамены)**:

Column Name	Data Type	Nullable	Default	Primary Key
EXAM_ID	NUMBER	No	-	1
STUD_NAME	VARCHAR2(20)	No	-	-
EXAM_NAME	VARCHAR2(20)	No	-	-
MARK	NUMBER	Yes	-	-

Занесите в таблицу следующие данные:

EDIT	EXAM_ID	STUD_NAME	EXAM_NAME	MARK
	1	Sergey	BD	5
	2	Sergey	Math	3
	3	Sergey	Inform	-
	4	Ivan	BD	3
	5	Ivan	Math	4

Лабораторная работа № 2. Создание пользовательских приложений.

Пользовательские приложения в Oracle представляют собой отчеты и формы, выполненные как Web-страницы и позволяющие получать информацию из Базы данных.

Рассмотрим пример создания приложения формы отчета по таблицам, входящим в учебную базу данных пользователя с идентификатором HR.

2.1. Разблокировка пробной учетной записи пользователя.

Чтобы создать свое приложение, необходимо войти как пользователь базы данных. Oracle Database XE поставляется с экспериментальным пользователем базы данных, именуемым **HR**. Этот пользователь владеет несколькими таблицами базы данных в пробной схеме, которая может быть использована при создании приложений для вымышленного подразделения Human Resources. Однако, из соображений безопасности, учетная запись этого пользователя заблокирована. Вы должны разблокировать эту учетную запись, прежде чем сможете создать свое пробное приложение.

Чтобы разблокировать пробную учетную запись пользователя:

1. Убедитесь, что Вы все еще подключены как администратор базы данных, как это описано в предыдущем разделе.
2. Щелкните на иконке **Administration**, а затем щелкните **Database Users**.
3. Щелкните на иконку схемы **HR**, чтобы отобразить пользовательскую информацию для **HR**.



4. В *Manage Database User* введите следующие настройки:
 - **Password** и **Confirm Password**: введите *hr* в качестве пароля.
 - **Account Status**: Выберите *Unlocked*.
 - **Roles**: Убедитесь, что активированы как **CONNECT**, так и **RESOURCE**.
5. Щелкните **Alter User**.

Теперь все готово для создания приложения.

2.2. Подключение к пробной учетной записи.

Для подключения к пробной учетной записи:

1. Закончите работу с учетной записью администратора базы данных, нажав **Logout** в верхнем правом углу домашней страницы базы данных.
 2. Нажмите **Login**.
 3. В окне подключения введите **hr** в качестве имени пользователя и пароля.
 4. Нажмите **Login**.
- Появится домашняя страница базы данных.

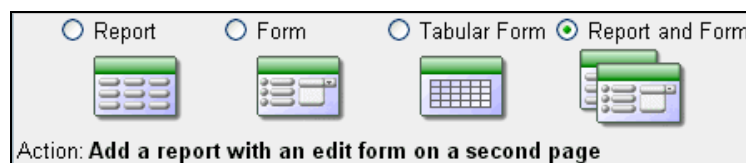
2.3. Создание простого приложения.

Создание приложения - это самый простой способ просматривать и редактировать данные в Вашей базе данных. Создайте это приложение на основе таблицы **EMPLOYEES**, являющейся частью схемы **HR**, для этого:

1. На домашней странице базы данных щелкните иконку **Application Builder**.
2. Нажмите кнопку **Create**.
3. На открывшейся странице выберите **Create Application** и нажмите **Next**.
4. В поле *Create Application* введите следующие параметры:
 - a. В поле *Name* введите **MyApp**.
 - b. Остальные поля оставьте без изменений.
 - c. Нажмите **Next**.

Далее добавьте страницы к вашему приложению.

5. Под разделом *Add Page*:
 - a. В опции *Select Page Type* выберите **Report and Form**.



Обратите внимание, что в поле **Action** отображается тип страницы, которую вы добавляете.

- b. В поле *Table or View* выберите **EMPLOYEES**.
- c. Нажмите кнопку **Add Page**.

Две новые страницы отобразятся вверху страницы в разделе *Create Application*.

Create Application					Cancel	< Previous	Next >
Page	Page Name	Page Type	Source Type	Source			
1	EMPLOYEES	Report	Table	EMPLOYEES			✘
2	EMPLOYEES	Form	Table	EMPLOYEES			✘

- d. Click **Next**.
 6. Опцию *Tabs* оставьте без изменений (**One Level of Tabs**) и нажмите **Next**.
 7. Опцию *Shared Components* оставьте без изменений и нажмите **Next**.

Эта опция позволит вам импортировать общие компоненты из других приложений. Общие компоненты - это стандартные элементы, которые могут быть отображены или применены на любой странице приложения.

8. Параметры полей: *Authentication Scheme*, *Language* и *User Language Preference Derived From* оставьте без изменений и нажмите **Next**.

9. В опциях **User Interface** выберите **Theme 2** и нажмите **Next**.

Темы - это наборы шаблонов, которые можно использовать для задания расположения элементов и определения внешнего вида всего приложения.

10. Подтвердите сделанные изменения. Чтобы вернуться на предыдущую страницу мастера, нажмите **Previous**. Чтобы принять изменения, нажмите **Create**.

После того как вы нажмёте **Create**, вверху страницы появится следующее сообщение: «Application created successfully».

2.4. Запуск нового приложения.

Чтобы запустить Ваше приложение:

1. Щелкните иконку **Run Application**.



2. На странице авторизации, введите **hr** как в поле **User Name**, так и в поле **Password**.

Появится Ваше приложение, показывая таблицу **EMPLOYEES**.

3. Исследуйте Ваше приложение. При желании, можно делать запросы к таблице **EMPLOYEES**. Для управления приложением, используйте инструментальное меню разработчика в нижней части страницы:



Инструментальное меню разработчика дает возможность оперативно отредактировать текущую страницу, создать новую страницу, элемент управления или компонент, посмотреть состояние сессии, а также включить/отключить режим отладки или ссылки редактирования.

4. Для выхода из приложения и возврата в Application Builder, щелкните **Edit Page 1** в инструментальном меню разработчика.

5. Для возврата на домашнюю страницу базы данных выберите пункт **Home** в верхней части страницы:



Поздравляем! Вы только что создали своё первое приложение, используя Oracle Database XE.

Лабораторная работа № 3. Работа с базой данных с использованием утилиты SQLplus.

Утилита SQLplus является клиентским приложением, которое может осуществить доступ к базе данных Oracle Express через локальную или глобальную сеть. Ее можно установить на компьютер клиента без установки сервера Oracle Express. При установке сервера она устанавливается автоматически.

Для работы с этой утилитой, добавим нового пользователя с именем **Stud**. Для этого выполните команды:

1. Запустите домашнюю страницу **Oracle Express**, выполняя команду **Пуск→Все программы→Oracle Database 10g Express Edition→Go To Database Home Page**.

2. В появившемся приглашении введите логин *system* и пароль *oracle*. В первой вкладке администрирования выберите раздел **DateBase Users→Create Users** и добавьте нового пользователя с именем **test**, пароль – **test**.

Выполните подключение к базе данных пользователя **test** с помощью **SQLplus**, для этого:

3. Откройте меню **Пуск** системы **Windows** и нажмите **«Выполнить»**. В появившемся окне введите команду **cmd** для запуска командного интерпретатора **DOS**.

4. Наберите команду **SQLplus**, нажмите клавишу <Enter>.

5. Введите имя пользователя **test**, затем пароль **test**. Должно появиться приглашение **SQL>**

6. Создайте таблицу:

- **Orders (Заказы)** с полями: *№ заказа, ФИО покупателя, Дата*.

```
Create table Orders (
  ID Int not null primary key,
  FIO varchar2 (20) not null,
  Data date not null);
```

Нажмите клавишу <Enter>.

При работе в командной строке DOS перемещаться и выбирать предыдущие команды можно с помощью кнопок клавиатуры: стрелок вверх-вниз.

- Добавьте новый столбец **Название товара: (Name_Tovar)**

```
Alter table Orders add (Name_Tovar varchar2 (40) not null);
```

Нажмите клавишу <Enter>.

- Добавьте строку данных.

```
Insert into Orders values (
  1, 'Ivanov', '25.10.2008', 'Bred');
```

- Добавьте еще несколько строк данных.
- Выполните просмотр данных из таблицы **Orders**

```
SELECT * FROM Orders;
```

клавиша <Enter>.

- Выполните просмотр количества записей:

```
SELECT count(*) FROM Orders;
```

клавиша <Enter>.

7. Выполните команду отсоединения:

Disconnect

Нажмите клавишу <Enter>.

Снова откройте домашнюю страницу, войдя под именем **test**, пароль – **test** и посмотрите созданную таблицу через раздел **Object Brower**.

Лабораторная работа № 4. Экспорт баз данных.

Цель работы: Научиться создавать базы данных в разных средах и экспортировать данные в другие форматы.

Задание 1. Создать источник данных DSN для связи с сервером Oracle и экспортировать таблицы.

Задание 2. Создать базу данных в MS Access.

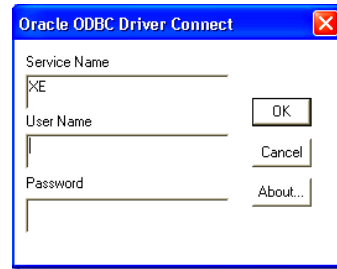
Порядок выполнения **задания 1.**

- 1) Открыть **Меню** и выполнить следующую команду: **Пуск→Настройки→Панель управления→Администрирование→Источники данных ODBC (Control Panel→Administrative Tools→Data Sources (ODBC)).**
- 2) Вкладка **User DSN (Пользовательский DSN).**
- 3) Кнопка **Add.**
- 4) Выбрать из списка **Oracle in XE.**
- 5) Кнопка **Finish.**
- 6) Открывается окно: **Oracle ODBC Driver Configuration.**
- 7) Ввести в окно «**Data Sources Name**» имя.
- 8) В окне **TNS Service Name** выбрать в открывающемся списке по стрелке **XE.**
- 9) **ОК.**
- 10) Откроется список всех имен пользователей.
- 11) **ОК.**

В результате будет установлено соединение с сервером Oracle.

Выполнить **экспорт таблиц**, для этого:

- 1) Выделить таблицу, которую необходимо будет экспортировать.
- 2) **Файл→Экспорт.**
- 3) Открывается окно «**Экспорт объекта: Таблица**»
- 4) Указать папку, в которой будет сохранена таблица.
- 5) В строке «**Тип файла**» из открывающегося списка выбрать **Базы данных ODBC (ODBC Databases).**
- 6) В окне **Экспорт** указать (или подтвердить) имя таблицы, под каким она будет перемещена в среду Oracle.
- 7) Нажать клавишу **<ОК>.**
- 8) В окне, где задано имя **test**, выделить его. Это же имя отобразится в окне **Имя DSN: test.**
- 9) Нажать клавишу **<ОК>.**
- 10) В следующем окне ввести данные: **Service Name: XE, User Name, Password** – которые были указаны при создании Пользователя в среде БД Oracle:



Задание 3.

- а) Создать базу данных в MS Access, содержащую следующие таблицы:

Таблица 1. Студенты:

- *ID студента* (номер студенческого билета)
- *ФИО студента*
- *Домашний телефон*
- *Номер группы*

Таблица 2. Сводная экзаменационная ведомость:

- *ID студента* (внешний ключ из табл.1)
- *Номер группы*
- *Оценка 1* (оценка по 1-ому экзамену) – пустое поле
- *Оценка 2* (оценка по 2-ому экзамену) – пустое поле

Таблица 3 Результаты сдачи экзамена по предмету БД гр. 10:

- *ID студента*
- *Оценка*

- b) Задать структуры трех таблиц (см.выше) в СУБД MS Access.
 c) Заполнить таблицы данными.
 d) Перенести таблицы в БД Oracle.
 e) Создать SQL-запрос и перенести данные из таблицы 3 в таблицу 2.

Лабораторная работа 5. Выполнение расчетов с использованием программирования в среде Visual Basic for Applications (VBA).

1. Создать таблицу **Employers**. Поля таблицы: **Months, Zarplata, Nadbavka**. Тип данных – числовой. Имя поля ввести латиницей.
2. Заполните поле **Months** цифрами от 1 до 12.
3. Создать пустую форму в режиме **Конструктора** (выбрать источник данных – пустая строка).
4. В меню **Вид→Панель элементов**, выбрать элементы: **Кнопка** и **Надпись** и перенести их на форму.
5. В открывшемся диалоговом окне нажать кнопку **<Cancel>**.
6. (В режиме Конструктора) Щелкнуть правой кнопкой мыши по созданной кнопке. Выполнить команду **Обработка событий→Программа**. Открывается диалоговое окно VBA.

Примечание: После ввода команды в среде VBA каждой строки нажать клавишу **Enter**.

```
Private Sub Кнопка0_Click()
Dim dbs As Database,r1 As Recordset           'определить объектные переменные
Dim n,s,z As Long
Set dbs=CurrentDb                             'текущая БД=currentDb
Set r1=dbs.OpenRecordset("Employers")        'присвоить переменной ссылку на таблицу "Employers"

r1.MoveFirst                                  'установить курсор на 1 запись
n=r1.RecordCount                              'подсчитать число записей в табл.
Do While Not r1.EOF                            'цикл по записям - пока не дойдет до конца записи
r1.Edit
r1!Zar=6000+1000*Int(Rnd()*10)                 'занести данные случайным образом
r1.Update
r1.MoveNext

Loop                                           'конец цикла
s=0
r1!MoveFirst
n=r1.RecordCount
Do While Not r1.EOF
z=r1!Zar
s=s+z
r1.Edit                                       'Редактировать таблицу
If z>6900 Then r1!Nad=500 Else r1!Nad=400    'Установить надбавку в зависимости от зарплаты
r1.Update                                     'занести исправления в таблицу
r1.MoveNext
Loop
s=s&"\n"
Надпись1.Caption=s                            'значение средней зарплаты вывести в надпись1
End Sub
```

7. Просмотреть поля таблицы **Employers**.
8. Какие данные отображаются в окне «**Надпись1**»?

Задание для самостоятельной работы.

1. Создать 3 таблицы:

– «**Товары**». Поля таблицы – *Наименование товара, Базовая цена* (тип поля – числовой). Ввести три записи.

– «**Продажи**». Поля таблицы – *Месяц, Товар 1, Товар 2, Товар 3, Базовая цена товара, Цена товара*.

Занести данные в столбец Месяц от 1 до 12.

– «**Курс Валюты**». Поля таблицы – *Месяц, КурсЕвро*.

Занести данные в поле Месяц от 1 до 12.

2. Написать программу на VBA, которая должна выполнить следующее:

1) Заполнить случайными числами поля таблицы «**Продажи**»: *Товар 1, Товар 2, Товар 3*.

2) Заполнить поле *КурсЕвро* таблицы «**Курс Валюты**» случайными числами от **40,5 до 43** с одним знаком после запятой (0,1).

Например, $r1!Zar = 6000 + 1000 * Int(Rnd() * 10)$

«6000» – минимальная зарплата

«1000» – шаг изменения зарплата

« $Int(Rnd() * 10)$ » – случайное число от 0 до 10.

3) Подсчитать стоимость проданного товара по базовой цене.

4) Подсчитать поле Цена товара, полученную путем умножения Базовой цены на коэффициент k , рассчитываемый по формуле:

$$k = 1 + (\text{курс евро} - 40) * 0,1$$

Лабораторная работа № 6. Формирование хранилища данных из разных источников

Задание: Используя в качестве источников табличные данные о результатах продаж, сформировать реляционное хранилище данных по схеме «звезда». (Каждый студент должен разработать свой вариант базы данных). Необходимо сформировать 5-6 таблиц измерений и 2 таблицы фактов в **MS Access** для дальнейшего переноса в Oracle.

Пример. Рассмотрим базовый пример «Фирма по оптовой продаже комплектующих для компьютера» для выполнения сквозного проекта по курсу «Хранилища данных», варианты которого были даны в лабораторной работе № 4.

Предположим, что наша фирма имеет два филиала, расположенные в разных городах, и есть центральный офис фирмы, в котором обрабатываются результаты продаж по этим регионам и формируется хранилище данных. Конечная цель работы – сформировать общее хранилище данных продаж, составить отчеты по результатам продаж в каждом из филиалов, сделать выводы об эффективности продаж по каждому из филиалов и составить рекомендации по дальнейшей деятельности филиалов.

I. Составить таблицы измерений:

1. Таблица «**Категории товаров (Category)**», поля: *ID категории, Наименование:*

Имя поля	Тип данных
ID_category	Числовой
Name_category	Текстовый

Ввести 4-5 категорий, например:

ID категории	Наименование категории
1	Материнская плата
2	Видеокарта
3	HDD
4	Карта памяти

2. Таблица «**Поставщики (Postavshiki)**», поля: *ID поставщика, Название, Категория товара, Юридический Адрес, Телефон:*

Имя поля	Тип данных
ID_postav	Числовой
Name_post	Текстовый
Adress	Текстовый
Phone	Текстовый

Ввести по 10-12 записей:

ID_postav	Name_post	Adress	Phone
1	ОАО "Сириус"	Казань	1234567
2	ООО "Сигма"	Москва	5678900
3	КФ "Компьютерия"	Москва	2467809
4	ЧП "Металком"	Уфа	245678
5	ООО "Кей"	Москва	1356992
6	ООО "Венс"	Ульяновск	156789

3. Таблица «**Товары (Tovary)**», поля: *ID товара, Код категории товара, Наименование Товара, Цена за ед., Гарантия:*

Имя поля	Тип данных
ID_tovara	Числовой
Kod_category	Числовой
Name_tovara	Текстовый
Cena_za_ed	Числовой
Garantiya	Текстовый

вести по 5 записей для каждой категории:

ID_tovara	Kod_category	Name_tovara	Cena_za_ed	Garantiya
1	1	ASUS MAXIMUS Extreme Socket-775	11 285,00р.	12
2	1	ASUS P5E3 Deluxe Socket-775	9 023,00р.	12
3	2	256Mb PCI-E HD2400 XT TV DVI	1 830,00р.	12
4	2	512Mb PCI-E HD2600 Pro DDR2 TV DVI	2 108,00р.	12
5	3	500 Gb SATA2 Seagate-Maxtor STM350	3 581,00р.	24
6	3	250 Gb IDE Hitachi HDP725025GLAT80	2 645,00р.	24
7	3	SATA 080 Gb Western Digital	2 274,00р.	24
8	4	DIMM DDR2 PC2-4300/ 4200 533MHz	511,00р.	12
9	4	DDR2 PC2-8500 1066MHz 2048	3 192,00р.	12
10	3	256Mb PCI-E HD2600 Pro DDR2 TV DVI	1 900,00р.	12

4. Таблица «**Покупатели (Pokupateli)**», поля: *ID Покупателя, Наименование организации, Контактное лицо, Город, Адрес:*

Имя поля	Тип данных
ID_pokupatel	Числовой
Name_pokupatel	Текстовый
FIO_contact_lisa	Текстовый
City	Текстовый
Adress	Текстовый

Заполнить таблицу данными (10-12 записей):

ID_pokupatel	Name_pokupatel	FIO_contact_lisa	City	Adress
1	КГУ	Иванов	Казань	ул.Кремлевская
2	Школа № 131	Петров	Казань	ул.Островского
3	МГУ	Федорова	Москва	ул.Ленинские Горы
4	ТЦ "Mera"	Сидорова	Казань	пр.Победы
5	ТЦ "Mera-2"	Калинина	Уфа	пр. Мира
6	Дет.садик 15	Никифорова	Альметьевс	ул.Садовая

5. Таблица «**Служащие филиала 1: (Prodavci_Filial_1)**»: поля: *ID продавца, ФИО, Телефон:*

Имя поля	Тип данных
ID_prodavca	Числовой
FIO_prodavca	Текстовый
Phone	Текстовый

Заполнить таблицу данными (не менее 5 записей):

ID_prodavca	FIO_prodavca	Phone
1	Садыков	157-23-09
2	Галныкин	512-12-12
3	Малкин	298-98-98
4	Рудаков	121-21-20
5	Пенкин	234-52-30

5.1. Таблица «**Служащие филиала 2: (Prodavci_Filial_2)**»: поля: *ID продавца, ФИО, Телефон.*

6. Таблица «**Таблица скидок (Table_Skidka)**», поля: *Количество товара, Скидка, Описание:*

Имя поля	Тип данных
kolich_tovara	Числовой
skidka_v_procent	Числовой
description	Поле MEMO

Заполнить таблицу данными:

Количество товара	Скидка, %	Описание
5	3	Скидка в размере 3% при покупке товара от 5 до 10 шт.
10	5	Скидка в размере 5% при покупке товара от 10 до 20 шт.
20	7	Скидка в размере 7% при покупке товара от 20 и более.

II. Составить таблицы фактов:

В нашем проекте будет создано две таблицы фактов.

Таблица «**Заказы**» будет содержать данные о заказах – номер заказа, ID продавца. ID покупателя и дату совершения заказа

Таблица «**Подробности заказов**» содержит данные о товарах, входящих в заказ (от 1 до 3 наименований - это число можно задать в программе) и их количестве. Количество генерируется случайным образом в пределах от 1 до 50 (этот параметр также можно поменять).

Примечание: Эти таблицы будут заполняться автоматически с помощью программ, представленных ниже.


1. Создать структуру таблицы «**Заказы (TABLE_ZAKAZ)**», поля таблицы: *Код подразделения, ID заказа, ID заказчика, Дата заказа, ID продавца:*

	Имя поля	Тип данных
▶	ID_filiala	Числовой
	ID_zakaza	Числовой
	ID_pokupatel	Числовой
	Data_zakaza	Дата/время
	ID_prodavca	Числовой

2. Создать структуру таблицы «**Подробности заказа (TABLE_Detali_zakazov)**», поля таблицы: *ID заказа, ID товара, Код филиала, ID Поставщика, Количество, Цена за единицу, Общая цена, Скидка, Цена с учетом скидки:*

	Имя поля	Тип данных
▶	ID_ZAKAZA	Числовой
	ID_tovara	Числовой
	ID_filiala	Числовой
	ID_Postav	Числовой
	Kolichestvo	Числовой
	Cena_za_ed	Числовой
	Vsego_cena	Числовой
	Skidka	Числовой
	Cena_so_skidkoi	Числовой

3. Для заполнения таблиц данными создать форму с кнопками, при нажатии которых будут генерироваться данные для заполнения таблиц «**TABLE_ZAKAZ**» и «**TABLE_Detali_zakazov**».

- Открыть вкладку **Формы**→**Создать**→**Конструктор**→**ОК**. Откроется **Мастер Формы**.
- На панели инструментов выбрать кнопку , щелкнув по ней и перетащив её в поле формы. Открывается **Мастер создание кнопок**.
- Кнопка <Отмена> закрывает мастера.
- Щелкнув правой кнопкой мыши по кнопке, выполнить: **Обработка событий**→**Программы**→**ОК**. При этом открывается окно редактора **Visual Basic**.

Листинг программы для первой кнопки (заполняем таблицы «**TABLE_ZAKAZ**» и «**TABLE_Detali_zakazov**»):

```
' Программа 1.
Private Sub Кнопка0_Click()
Dim nz As Integer
Dim d As String
Dim dbs As Database,r1 As Recordset,r2 As Recordset
Set dbs=CurrentDb
Set r1=dbs.OpenRecordset("table_zakaz")
Set r2=dbs.OpenRecordset("TABLE_Detali_zakazov")
Dim t As Integer
```



```

nz=InputBox("Необх.колич-о заказов (не менее 200)", , "200")
nt=InputBox("Число товаров", , "10")
np=InputBox("Число поставщиков", , "6")
nf=InputBox("Код филиала", , "1")
npr=InputBox("Число продавцов в филиале", , "5")
nzz=InputBox("Число покупателей", , "6")

' Заполнение данными таблицы Table_Zakazov "Таблица заказов"

For i=1 To nz
izz=Int(Rnd*nzz)+1 ' номер заказчика
ip=Int(Rnd*npr)+1 ' номер продавца
d=Int(Rnd*30)+1 ' Выбор дня
d=d+"." ' Формирование даты в формате dd.mm.yy
m=Str(Int(Rnd*12)+1)
d=d+m+"."
m=Str(Int(Rnd*3)+2010)
d=d+m
r1.AddNew
r1!ID_filiala=nf
r1!ID_zakaza=i 'Изменяемое поле. Для второго филиала
                должно=i+1000
r1!ID_pokupatel=izz
r1!Data_zakaza=d
r1!ID_prodavca=ip
r1.Update
Next i
' Заполнение данными таблицы Table_Detali_Zakaz "Детали заказов"
For i=1 To nz
izz=nt*(Rnd*nzz)+1 ' номер заказа
kz=Int(Rnd*3)+1 ' кол.позиций данного заказа (от 1 до 3)
ip=Int(Rnd*npr)+1 ' номер продавца
For k=1 To kz
r2.AddNew
r2!ID_filiala=nf
r2!ID_zakaza=i 'Изменяемое поле.Для второго филиала должно=i+1000
r2!ID_tovara=Int(Rnd*nt)+1
r2!ID_postav=Int(Rnd*np)+1
r2!Kolichestvo=Int(Rnd*49)+1
r2.Update
Next k
Next i
End Sub

```

-
- Закрывать редактор VBA.
 - Сохранить форму под названием «Forma_Zakazov».
 - Открыть форму с помощью **Конструктора** и переименовать кнопку на «Сформировать заказы». Закрывать форму.
 - Открыть форму и запустить процедуру «Сформировать заказы». В ходе работы программа запросит ввести необходимые данные: *Количество заказов* – это число определяет, сколько строчек будет внесено в таблицу фактов о заказах. Данные будут разбросаны по месяцам 2010-2012 г.г. Остальные значения должны соответствовать количеству товаров, поставщиков, продавцов, покупателей, внесенных в таблицы измерений.

– Задайте первоначально значения *Количества заказов* равным 10 записям для проверки работы программы.

В результате в таблице «**Детали заказов (TABLE_Detali_zakazov)**» будут заполнены следующие поля:

	id_ZAKAZA	ID_tovara	ID_filiala	ID_Postav	Kolichestvo	Cena_za_ed	Vsego_cena	Skidka	Cena_so_skidk
▶	1	10	1	4	49	0р.	0р.	0	0р.
	1	5	1	5	16	0р.	0р.	0	0р.

и т.д., в зависимости от количества введенных данных.

Все поля таблицы «**Заказы (TABLE_ZAKAZ)**» будут заполнены.


Примечание. При повторном запуске программы, *добавляются* новые данные к уже существующим данным в таблице заказов. Если необходимо заново сформировать таблицу заказов, то сначала надо удалить из нее все записи с помощью запроса на удаление (См. ниже).

4. Следующим действием необходимо в таблице «**Детали заказов**» заполнить поле *Цена за единицу (Cena_za_ed)*, перенеся данные из таблицы «**Товары (Tovary)**». Будем считать цены за единицу товара в таблице «**Товары**» оптовыми, а розничные цены установим на **20%** выше оптовых. Вычислим значение розничных цен в таблице «**Детали заказов (TABLE_Detali_zakazov)**».

Для этого необходимо выполнить следующий запрос.

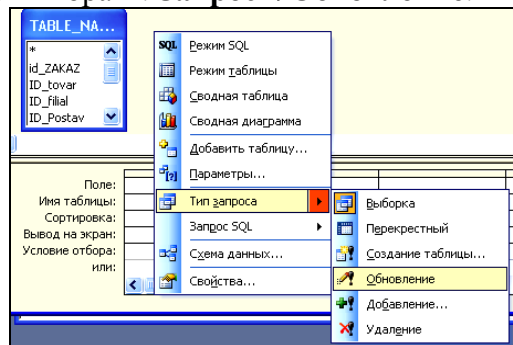
- Вкладка **Запросы**→**Создать**→**Конструктор**. Из списка выбрать таблицу «**Tovary**».
- Кнопка **<Закрыть>**.
- На панели инструментов выполнить команду: **Запрос**→**Обновление**.
- Затем выбрать **Вид**→**Режим SQL**.
- Записать запрос:

```
UPDATE TABLE_Detali_zakazov INNER JOIN Tovary ON TABLE_Detali_zakazov.ID_tovara=Tovary.ID_tovara SET
TABLE_Detali_zakazov.Cena_za_ed=Tovary!Cena_za_ed*1.2;
```

- Выполнить запрос, нажать кнопку на панели инструментов: **<Запуск>** .
- Сохранить запрос под названием «*Запрос на обновление-Цена за ед.*».

- Запрос на вычисление **Цены_Всего (Vsego_cena)**:

- 1) Выполнить команду: **Запросы**→**Создать**→**В режиме конструктора**.
- 2) добавить таблицу «**TABLE_Detali_zakazov**».
- 3) На панели инструментов выбрать: **Запрос**→**Обновление**.



- 4) В графу **Поле** вставить поле **Vsego_cena**, щелкнув два раза по имени поля в таблице.
- 5) В строке **Обновление**: щелкнув правой кнопкой мыши, выбрать **Построить**.

6) В открывшемся диалоговом окне **Построитель выражений** выбрать **Таблицы**→**TABLE_Detali_zakazov**→**Cena_za_ed**, выбрать операцию умножения - (*), далее выбрать **Kolichestvo** и нажать кнопку **<ОК>**.

В окне построителя выражение будет выглядеть следующим образом:

```
TABLE_Detali_zakazov!Cena_za_ed*TABLE_Detali_zakazov!Kolichestvo
```

7) Сохраните запрос «Цена всего».

8) В режиме **SQL** запрос будет выглядеть следующим образом:

```
UPDATE TABLE_Detali_zakazov SET
TABLE_Detali_zakazov.Vsego_cena=TABLE_Detali_zakazov!Cena_za_ed*TA
BLE_Detali_zakazov!Kolichestvo;
```

Примечание: Для переключения в режим **SQL** надо открыть запрос в конструкторе, затем на панели инструментов выбрать меню **Вид**→**SQL**.

В результате этого запроса будет вычислено и заполнено данными поле **Vsego_cena**.

5. Добавить кнопку для внесения данных в поле **Skidka** таблицы «**TABLE_Detali_zakazov**» из таблицы «**Table_Skidka**».

Для этого открыть форму в режиме Конструктора, добавить из панели элементов кнопку, она будет иметь имя **Кнопка1**, в окне редактора **Visual Basic** набрать следующую программу:

Листинг программы для кнопки *Расчет скидки*, для таблицы «**TABLE_Detali_zakazov**»:

' Программа 2.

```
Private Sub Кнопка1_Click()
Dim dbs As Database, r1 As Recordset, r2 As Recordset
Dim Skidki(2,10) As Integer
Set dbs=CurrentDb
Set r1=dbs.OpenRecordset("TABLE_Detali_Zakazov")
Set r2=dbs.OpenRecordset("Table_Skidka")
n=r2.RecordCount           'Количество строк в таблице Скидки
r2.MoveFirst
k=1
While k<=n
Skidki(1,k)=r2!kolich_tovara
Skidki(2,k)=r2!Skidka_v_procent
If k<n Then r2.MoveNext
k=k+1
Wend
Skidki(1,k)=1000           'предельное значение для кол. товара
r1.MoveFirst
Do While Not r1.EOF       'Цикл по табл.Заказы
z=r1!Kolichestvo
k=1
While (z>Skidki(1,k)) And (k<n)
k=k+1
Wend
s=(r1!Vsego_cena)*Skidki(2,k)\100
r1.Edit
```

```

r1!Skidka=s
r1.Update
r1.MoveNext
Loop
End Sub

```

- Закрывать редактор VBA, закрыть Конструктор формы и запустить форму.
- **Запрос на обновление:** вычисление величины цены товара со скидкой (с помощью **SQL** или в режиме **Конструктора** с помощью **Построителя**):

```
UPDATE Table_Detali_zakazov SET Cena_so_skidkoi=Vsego_cena-Skidka;
```

(в каждой строчке поля Cena_so_skidkoi будет вычислена разность полной цены и скидки).

6. Выполнить запросы на удаление.

- Запрос на удаление данных из таблицы «**TABLE_ZAKAZ**»:
- **Запрос→Конструктор→TABLE_Zakaz.** Правой кнопкой мыши щелкнуть по полю и выбрать **Вид запроса→Удаление.**
- **Вид→Режим SQL.**

```
DELETE *
FROM TABLE_Zakaz;
```

- Запрос на удаление данных из таблицы «**TABLE_Detali_zakazov**»:

```
DELETE *
FROM TABLE_Detali_Zakazov;
```

7. Добавить в таблицу «**TABLE_ZAKAZ**» новое поле *ИТОГО*, в котором рассчитать общую стоимость всего заказа. Для этого выбрать данные из таблицы «**TABLE_Detali_zakazov**» (объединить данные, относящиеся к одному заказу, из таблицы деталей заказов и найти сумму со скидкой для всех товаров, входящих в заказ).

Подсказка: Задайте структуру временной таблицы «**Temp**» с полями: *Id_zakaza*, *Itogo*, куда с помощью запроса перемести суммарные данные из таблицы «**TABLE_Detali_zakazov**» с помощью запроса с группировкой данных по полю *Id_zakaza*:

```
INSERT INTO Temp ( Id_zakaza, Itogo )
SELECT id_zakaza, Sum(Cena_so_skidkoi)
FROM TABLE_Detali_zakazov
GROUP BY Id_zakaza;
```

Затем выполните еще один запрос **на обновление** для переноса значения из поля *Итого* (*Itogo*) таблицы «**Temp**» в аналогичное поле таблицы «**Table_Zakaz**».

III. Формирование данных для филиала 2.

После выполнения п.7 будет сформировано две таблицы фактов, содержащие столько записей, сколько было указано заказов при запуске первого сценария (кнопка 0). Однако обе таблицы содержат данные, относящиеся только к одному филиалу.

Необходимо дополнить полученные таблицы данными о продажах во втором филиале. Для выполнения этого необходимо повторить все указанные операции, перечисленные в части II лабораторной работы, для второго филиала. Чтобы номера заказов, выполненные во втором филиале, не совпадали с номерами заказов, необходимо, чтобы поле *Id_zakaza* в обеих таблицах фактов начинались с числа, превышающего число записей в таблице «**Table_Zakaz**».

Для этого необходимо:

- добавить кнопку для формирования заказов в филиале **2**;
- скопировать листинг *Программы 1*.
- в программе параметр *i* изменить на $i+1000$.

Лабораторная работа № 7. Оценка эффективности работы филиалов

Задание. Импортировать все таблицы, созданные в лабораторной работе №6, в СУБД Oracle.

Внимание! Имя пользователя должно совпадать с фамилией студента.

Запустить **Oracle Database 10g Express Edition**→**Oracle Database Home Page**, вкладка **SQL SQL Commands**. Выполнить запросы к созданной БД с целью оценки эффективности работы филиалов.

Перечень SQL запросов, необходимых для выполнения работы:

1. Подсчитать общую сумму заказов по годам для каждого из филиалов.
2. Подсчитать прибыль предприятия по годам для каждого из филиалов. Для этого, учитывая разницу между оптовыми и розничными ценами подсчитать сначала разницу между заработанными и затраченными средствами, потом вычесть издержки, связанные с зарплатами продавцов. Считать зарплату продавцов одинаковой и равной 10000 руб. в мес. Количество продавцов определяется таблицами **«Продавцы филиалов»**.
3. Выбрать лучший филиал по общей выручке за 2012 год.
4. Определить лучшего продавца по каждому из филиалов за 2012 год.
5. Подсчитать среднюю выручку на одного продавца в каждом из филиалов.
6. Определить 3 самых ходовых товара по каждому из филиалов и по всему предприятию в целом.
7. Определить самые плохо продаваемые товары за 2012 год.
8. Сделать выводы и выработать рекомендации о развитии продаж в будущем году.
9. Сохраните весь набор полученных SQL-запросов в один файл.
10. Выполнить письменный отчет по всей работе, включив данные (скриншоты) базовых таблиц, тексты запросов, результаты выполнения.