

## Лабораторная работа № 14

1. Тема : Управление базой данных с помощью SQL – операторов.

2. . Цель работы :

- Освоить приемы создания и модификации базы данных с помощью запросов.
- Освоить приемы управления базой данных с помощью запросов.

3. Литература:

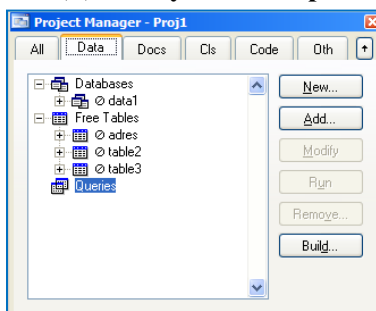
1. Омельченко Л. ДШевякова, VisualFoxPro 9.0 Санкт-Петербург, 2009
2. Т.В.Мухина, VisualFoxPro 9.0 Санкт-Петербург, 2009
3. В.Клепинин, Т.Агафонова, VisualFoxPro 9.0 Санкт-Петербург, 2008
4. Голицына О.Л., Максимов Н.В., Попов И.И. Базы данных Учебное пособие. М.Форум-Инфра-М,2009
5. Фуфаев Э.В., Фуфаев Д.Э. Базы данных.Учебное пособие.М. Издат.центр «Академия», 2006

4. Теоретические сведения

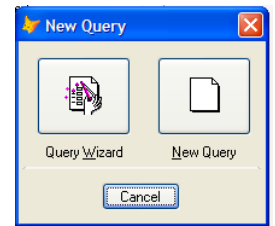
Мощным средством выборки данных является оператор **SELECT**, который можно создать с помощью **Построителя** или **вручную**.

1. Задание выборки с помощью Построителя.

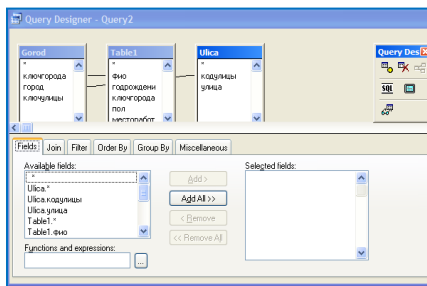
В этом случае оператор **SELECT** конструируется автоматически и в результате получается **файлзапроса** с расширением **.QRP**. Далее файл запроса ставится на выполнение с помощью одного из элементов управления. Для запуска **Построителя** надо выполнить :



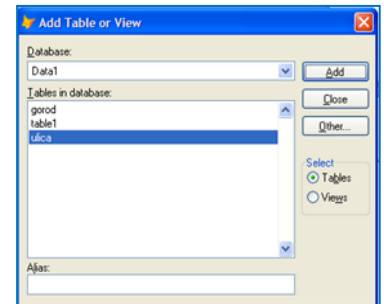
**а) В окне Диспетчера проектов (ProjectManager)** выбрать вкладку **Данные (Data)**, раскрыть группу **Запросы (Queries)** и нажать кнопку **Новый (New)** Далее выбрать кнопку «NewQuery».



**б) Работа с Построителем запросов.**



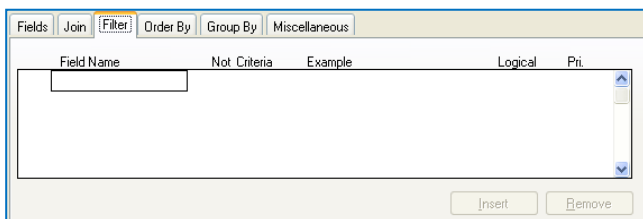
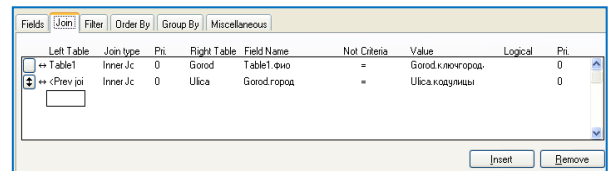
На экран выводится диалоговая панель **Построителя запросов(QueryDesigner)**. В рабочее поле Построителя переносятся таблицы, данные из которых будут использованы в запросе (выборке). Таблицы переносятся в рабочее окно с помощью диалоговой панели **Добавить таблицу в просмотр (AddTableorView)** по известным правилам.



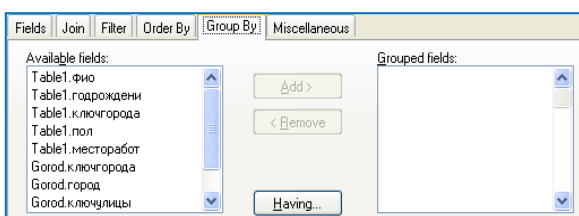
В нижней части диалогового окна Построителя запросов имеется:

**Вкладка Поля (Fields)** предназначена для выбора полей из разных таблиц, данные из которых будут помещены в результирующую таблицу. Отбор полей производится традиционным способом, т.е. перемещение имен полей из окна «Available fields» в окно «Selected fields». В поле ввода «FunctionandExpression» можно задать выражение для вычисляемого поля, либо явно, либо воспользовавшись кнопкой с помощью специального редактора.

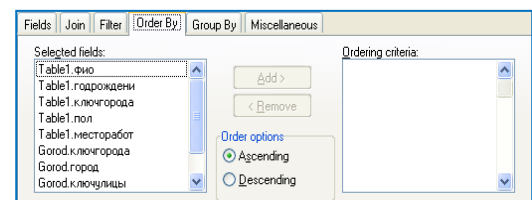
**Вкладка «Join»** определяет условия объединения таблиц. Если взаимосвязи между таблицами были установлены ранее, то они уже будут представлены в окне. При этом можно добавить новые взаимосвязи или удалить ненужные взаимосвязи



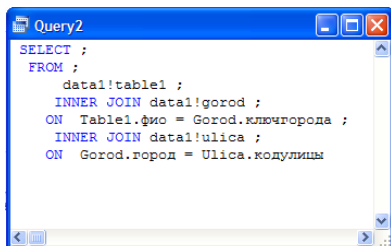
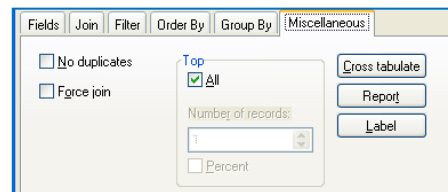
**Вкладка «Filter»** определяет условия для отбора записей Например отобразить записи для покупателя с указанной фамилией, или все покупки на указанную дату (т.е. задание условия WHERE).



**Вкладки «Orderby» и «Groupby»** определяют критерии сортировки и группировки данных соответственно.

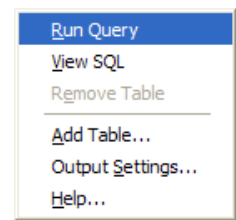


**Вкладка «Miscellaneous»** определяет дополнительные параметры выборки. Например, как поступить с повторяющимися значениями, ограничить количество отображенных записей или указать процент выбора записей.



Вызвав правой кнопкой мыши на экран **контекстное меню**, можно просмотреть созданный запрос с помощью команды **«View SQL»**

При закрытии окна «QueryDesigner» надо указать имя и место хранения созданного файла запроса. Далее с помощью команды **DO** файл запроса может быть поставлен на выполнение.



## Основные функции языка SQL

### Статистические функции.

- AVG () – среднее значение
- MAX (), MIN() – минимальное или максимальное значение
- SUM () – сумма значений
- COUNT – количество значений

### Строковые функции.

- UPPER (Str) – преобразование в верхний регистр
- LOWER (Str) – преобразование в нижний регистр
- TRIM (Str) – удаление пробелов в начале и конце строки
- SUBSTRING (Str FROM n1 TO n2) – выделение подстроки из строки
- CAST (<Expression> AS <Type>) – приведение выражения к нужному типу
- || – сцепление двух строк

### Функции декодирования даты и времени

- EXTRACT (<Элемент> FROM <выражение>) – из выражения содержащего значения даты или времени, извлекается значение, соответствующее указанному элементу.

## 2. Задание выборки (фильтрации) данных ручным способом.

**Выборка данных** из таблиц заключается в отборе полей и записей, **удовлетворяющих заданным условиям**. Результат выполнения запроса, на основании которого выбираются записи, **называется выборкой**. Данные можно выбирать из одной или нескольких таблиц с помощью оператора **Select**. Это один из важнейших операторов **SQL**. Он используется для отбора записей, удовлетворяющих сложным критериям отбора, и имеет формат:

```
SELECT [ DISTINCT ]
<Список полей> | *
FROM <Список таблиц>
[WHERE <Условие отбора>]
[ORDER BY <Список полей для сортировки>]
[GROUP BY <Список полей для группировки>]
[HAVING <Условие группирования>]
[UNION <Вложенный оператор SELECT>]
```

Результатом выполнения оператора **SELECT** является набор данных, в котором могут присутствовать повторяющиеся записи (одинаковые значения всех полей). Опция **DISTINCT** допускает присутствие только уникальных записей.

**<Список полей>** - это список имен полей разделенных запятыми, которые будут включены в результирующий набор данных. Если вместо списка полей стоит знак \* (звездочка), то выбираются все поля из указанной, ниже таблицы (или таблиц). Для обращения к полям из разных таблиц используется составное имя, т.е. имя таблицы точка имя поля. Кроме физических полей можно использовать и вычисляемые поля, для этого указывают не имя поля, а выражение, которое может не содержать имени поля. Поля берутся из таблиц, имена которых указаны в опции **FROM <список таблиц>** . Имена таблиц перечисляются **через запятую**.

Для более гибкой манипуляции данными возможно отбирать нужные записи в результирующий набор данных с помощью опции **WHERE <условие отбора>**, которое представляет собой

**<выражение 1><операция сравнения><выражение 2>**.

Кроме того, вместо операции сравнения возможно **сравнение по шаблону LIKE** где:

% - замещает любое кол-во символов;

\_ - замещение одного символа.

Также допускается использование служебных слов:

- IS NULL** - проверка нулевого значения,  
**BETWEEN** - проверка вхождения значения в диапазон,  
**IN** - проверка вхождения значения в список.

Для создания сложных критериев отбора используют сцепление простых критериев логическими операциями (**AND, OR, NOT**).

Для **сортировки** результирующего набора данных используют **опция ORDER BY <список полей>**. Если указано несколько полей, то при совпадении значений первого поля производится сортировка по значениям второго поля и т.д. Поля в списке обозначаются именами или номерами по порядку, как указано в <списке полей> в операторе **SELECT**. По умолчанию сортировка производится по возрастанию, для сортировки по убыванию нужно указать описатель **DESC**.

Записи набора данных могут быть **сгруппированы** по некоторому признаку. В группу входят записи, имеющие одинаковые значения полей указанных в операнде:

**GROUP BY <список полей для группировки>**

**HAVING <условие группировки>**

Опция **UNION** предназначена для более тщательного отбора данных. Внутри опции может располагаться вложенный оператор **SELECT... со всеми присущими ему опциями**.

### **Вопросы для самопроверки.**

1. Что такое выборка данных?
2. Какой командой выполняется выборка данных?
3. Укажите назначение опции **WHERE**.
4. Укажите назначение опции **ORDER**.
5. Укажите назначение опции **GROUP BY**.
6. Как в выборке задать название столбцов буквами русского алфавита?
7. Как в выборке задать вычисляемое поле?

### **ЗАДАНИЕ**

**1. Открыть проект со связанными таблицами (Город, Улица). Сделать выборку только по г. Таганрог, только по ул. Ленина, только по фамилии**

**2. Результаты выборки направить :**

- а) в запрос \_\_\_\_\_
- б) в просмотр \_\_\_\_\_
- в) в файл \_\_\_\_\_
- г) в отчет \_\_\_\_\_