

Практическая работа 54 Создание запросов и фильтров

Цель работы: Получить практический опыт создания в Microsoft SQL Server запросов и фильтров.

Перечень оборудования и программного обеспечения

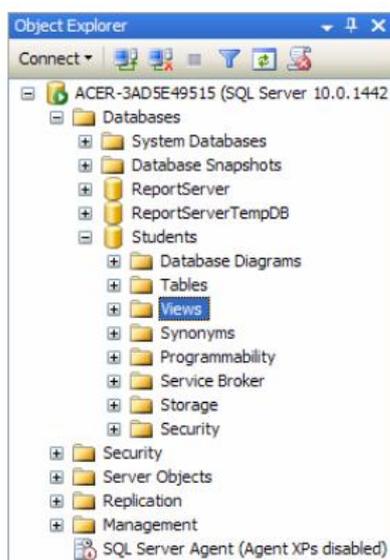
Персональный компьютер
Microsoft Office (Word, Visio, Access)
MS SQL Server 2008 R2

Краткие теоретические сведения

В таблицах баз данных обычно содержится много информации и довольно редко возникает необходимость выбирать все строки таблицы. Гораздо чаще бывает нужно извлечь какую-то часть данных таблицы для каких-либо действий или отчетов. Выборка только необходимых данных включает в себя критерий поиска, также известный под названием предложение фильтрации. В операторе SELECT данные фильтруются путем указания критерия поиска в предложении WHERE. Предложение WHERE указывается сразу после названия таблицы (предложения FROM)

SELECT ...FROM ...WHERE ...

В обозревателе объектов "Microsoft SQL Server" все запросы БД находятся в папке "**Views**"



1. Простое фильтрование оператором WHERE.

Пример запроса для отбора текстовых значений:

```
SELECT * FROM Sumproduct WHERE Product = 'Bikes'
```

Условие отбора взято в одинарные кавычки, что является обязательным при фильтровании текстовых значений. При фильтровании числовых значений кавычки не нужны.

Пример запроса для отбора числовых значений:

```
SELECT * FROM Sumproduct WHERE Amount > 40000 ORDER
```

BY Amount

В этом примере отобрали записи, в которых выручка от реализации составила более 40 тыс. \$ и, дополнительно, все записи отсортировали по возрастанию по полю Amount.

В таблице указан перечень условных операторов, поддерживаемых SQL:

Знак операции	Значение
=	Равно
<>	Не равно
<	Меньше
<=	Меньше или равно
>	Больше
>=	Больше или равно
BETWEEN	Между двумя значениями
IS NULL	Отсутствует запись

2. Фильтрация по диапазону значений (BETWEEN).

Для отбора данных, которые лежат в определенном диапазоне, используется оператор BETWEEN. В следующем запросе будут отображены все значения, лежащие в пределах от 1000 \$ в 2000 \$ включительно, в поле Amount.

```
SELECT * FROM Sumproduct WHERE Amount BETWEEN 1000 AND 2000
```

3. Выборка пустых записей (IS NULL).

В SQL существует специальный оператор для выборки пустых записей (называется NULL). Пустой записью считается любая ячейка в таблице, в которую не введены какие-либо символы. Если в ячейку введен 0 или пробел, то считается, что поле заполнено.

```
SELECT * FROM Sumproduct WHERE Amount IS NULL
```

4. Расширенная фильтрация (AND, OR).

Язык SQL не ограничивается фильтрацией по одному условию, для собственных целей вы можете использовать достаточно сложные

конструкции для выборки данных одновременно по многим критериям. Для этого в SQL есть дополнительные операторы, которые расширяют возможности оператора WHERE. Такими операторами являются: AND, OR, IN, NOT.

```
SELECT * FROM Sumproduct WHERE Amount > 40000 AND City = 'Toronto'
```

```
SELECT * FROM Sumproduct WHERE Month= 'April' OR Month= 'March'
```

Давайте объединим операторы AND и OR. Для этого сделаем выборку велосипедов (*Bikes*) и коньков (*Skates*), которые были проданы в марте (*March*).

```
SELECT * FROM Sumproduct WHERE Product = 'Bikes' OR Product = 'Skates' AND Month= 'March'
```

SQL имеет приоритеты выполнения команд. То есть оператор AND имеет более высокий приоритет, чем оператор OR, поэтому сначала были отобраны записи с коньками, которые проданные в марте, а потом все записи, касающиеся велосипедов.

Итак, чтобы получить правильную выборку, нужно изменить приоритеты выполнения команд. Для этого используем скобки, как в математике. Тогда, сначала будут обработаны операторы в скобках, а затем - все остальные.

```
SELECT * FROM Sumproduct WHERE (Product = 'Bikes' OR Product = 'Skates') AND Month= 'March'
```

5. Расширенная фильтрация (оператор IN).

```
SELECT * FROM Sumproduct WHERE ID IN (4, 12, 58, 67)
```

Оператор IN выполняет ту же функцию, что и OR, однако имеет ряд преимуществ:

- При работе с длинными списками, предложение с IN легче читать;
- Используется меньшее количество операторов, что ускоряет обработку запроса;
- Самое важное преимущество IN в том, что в его конструкции можно использовать дополнительную конструкцию SELECT, что открывает большие возможности для создания сложных подзапросов.

6. Расширенная фильтрация (оператор NOT).

```
SELECT * FROM Sumproduct WHERE NOT City IN ('Toronto', 'Montreal')
```

Ключевое слово NOT позволяет убрать ненужные значения из выборки. Также его особенностью является то, что оно проставляется перед названием столбца, участвующего в фильтровании, а не после.

Для фильтрации данных по критерию соответствия определенной символьной строки заданному шаблону используется оператор LIKE. Шаблон может включать обычные символы и символы-шаблоны. Во время сравнения с шаблоном необходимо, чтобы его обычные символы в точности совпадали с символами, указанными в строке. Символы-шаблоны могут совпадать с произвольными элементами символьной строки. Использование символов-шаблонов с оператором LIKE предоставляет больше возможностей, чем использование обычных операторов сравнения. Шаблон может включать в себя следующие символы-шаблоны.

Символ-шаблон	Описание	Пример
%	Любое количество символов	Инструкция WHERE FName LIKE 'A%' выполняет поиск и выдает всех клиентов, имена которых начинаются на букву 'A'.
_	Любой одиночный символ	Инструкция WHERE LName LIKE '_етров' выполняет поиск и выдает всех клиентов, фамилии которых состоят из шести букв и заканчиваются сочетанием 'етров' (Петров, Ветров и т.п.).
[]	Любой символ, указанный в квадратных скобках	Инструкция WHERE LName LIKE '[Л-С]омов' выполняет поиск и выдает всех клиентов, фамилии которых заканчиваются на 'омов' и начинаются на любую букву в промежутке от 'Л' до 'С', например Ломов, Ромов, Сомов и т.п.
[^]	Любой символ, кроме перечисленных в квадратных скобках	Инструкция WHERE LName LIKE 'ив[^а]%' выполняет поиск и выдает всех клиентов, фамилии которых начинаются на 'ив' и третья буква отличается от 'а'.

Задания

1 Изучить теоретические сведения.

2 В соответствии с вариантом задания выполнить создание запросов и фильтров:

- простое фильтрование;
- фильтрация по диапазону значений;
- расширенная фильтрация;
- расширенная фильтрация (оператор IN);
- запрос с параметром;
- запрос с вычисляемым полем.

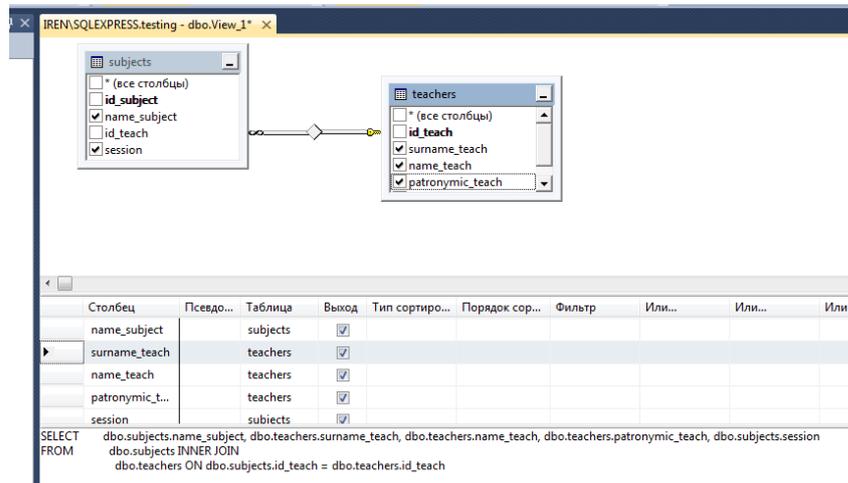
Порядок выполнения работы

Добавление нового запроса. Команда «New Query»

К базе данных можно создавать запросы. В нашем случае нужно создать запрос в соответствии с условием задачи.

Запрос создается с помощью команды «New Query», которая вызовется из контекстного меню. Чтобы вызвать команду, достаточно сделать клик правой кнопкой мышки в области поля, которое выделено для отображения элементов базы данных. Следует отметить, что запросы не сохраняются

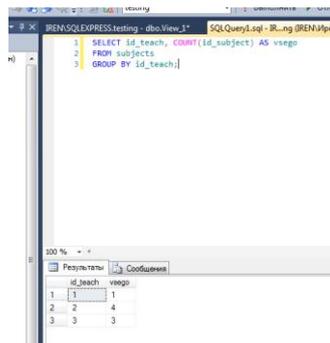
В таблицах нужно выделить поля, которые будут использоваться в запросе. Порядок выбора полей должен соответствовать отображению их в запросе в соответствии с условием задачи.



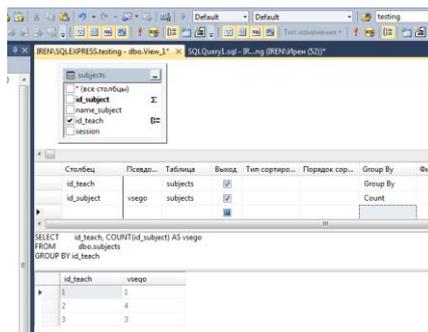
Как видно из рисунка, в нижней части окна отображается запрос на языке SQL, сформированный системой

Добавление вычисляемого поля

Чтобы создать вычисляемое поле, нужно в окне, где отображается SQL-запрос изменить текст этого запроса. Или составить SQL-запрос с помощью скрипта. Например:



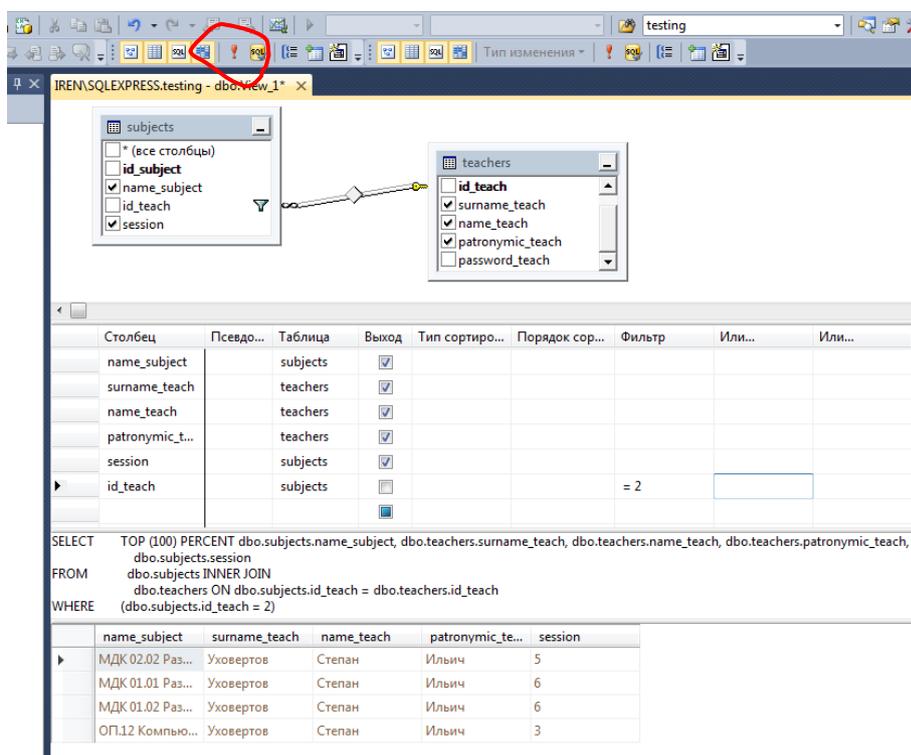
Добавляется вычислительное поле vsego, которое есть вычисление количества предметов для каждого преподавателя



Запуск запроса на выполнение. Команда «Execute SQL»

Чтобы запустить на выполнение запрос, используется кнопка «Execute SQL».

В нижней части окна на рисунке изображен результат выполнения запроса.



Другой способ запуска запроса на выполнение – команда «Execute SQL» из меню «Query Designer».

Содержание отчета

- 1 Название работы
- 2 Цель работы
- 3 Перечень технических средств обучения
- 4 Порядок выполнения работы
- 5 Ответы на контрольные вопросы
- 6 Вывод

Варианты заданий

Варианты заданий представлены в практической работе № 39

Контрольные вопросы:

1. Для чего предназначены запросы к базе данных MS SQL Server?
2. Какими способами можно выполнить запросы к базе данных MS SQL Server?
3. Какая команда осуществляет выборку данных в запросе?
4. С помощью какой команды можно выбрать диапазон данных?
5. Как выбрать записи с пустыми полями в MS SQL Server?
6. Как выполнить сложный запрос с несколькими условиями?
7. Для чего используется оператор IN?
8. В чем преимущества оператора IN?
9. Для чего используется оператор NOT?

Используемая литература

- Г.Н.Федорова Основы проектирования баз данных. М.: Академия, 2020
- Г.Н.Федорова Разработка, администрирование и защита баз данных. М.: Академия, 2018
- <http://www.ikasteko.ru/page/filtracija-dannyh-sql-server>
- https://www.bestprog.net/ru/2018/01/02/an-example-of-creating-a-query-in-the-ms-sql-server-database-the-database-is-located-in-the-local-mdf-file_ru/