

ЛАБОРАТОРНАЯ РАБОТА №2

Команда SELECT

1. Цель работы

1. Изучение основ выборки данных.
2. Изучение конструкции DISTINCT | ALL.
3. Изучение сортировки данных.
4. Изучение конструкции TOP.
5. Изучение конструкции OFFSET и FETCH.

2. Теоретическая часть

Подавляющее большинство пользователей используют SQL для организации выборки данных. Для выборки данных из БД используется запрос SELECT. Он позволяет фильтровать выбранные данные и преобразовать их к нужному виду. Результатом выполнения запроса SELECT является другая таблица, к которой снова может быть применен запрос SELECT.

Полный синтаксис инструкции SELECT сложен, однако основные предложения можно вкратце описать следующим образом:

```
[ WITH { [ XMLNAMESPACES , ] [ <common_table_expression> ] } ]
```

```
SELECT select_list [ INTO new_table ]
```

```
[ FROM table_source ]
```

```
[ WHERE search_condition ]
```

```
[ GROUP BY group_by_expression ]
```

```
[ HAVING search_condition ]
```

```
[ ORDER BY order_expression [ ASC | DESC ] ]
```

Обработка элементов запроса SELECT выполняется в следующей последовательности:

1. FROM – определяет имена используемых таблиц;
2. WHERE – фильтрует строки таблицы в соответствии с заданными условиями;
3. GROUP BY – группирует строки, имеющие одинаковые значение в указанном столбце;
4. HAVING – фильтрует группы строк в соответствии с указанным условием;
5. SELECT – форматирует выходные данные;
6. ORDER BY – сортирует результаты выполнения запроса.

Порядок предложений в запросе SELECT не может быть изменен. Предложения SELECT и FROM являются обязательными, присутствие остальных зависит от контекста.

В предложении **SELECT** указывается список столбцов, которые должны быть возвращены запросом. Можно указать исходные элементы или вычисляемые поля во время выполнения запроса.

Конструкция **DISTINCT** | **ALL** исключает / разрешает вывод повторяющихся строк. Конструкция **ALL** используется по умолчанию.

* означает вывод всех столбцов указанной таблицы. В случае, если выборка производится из нескольких таблиц, перед символом звездочки может указываться имя таблицы.

SQL-запрос может содержать вычисляемые столбцы, значения которых могут определяться на основе значений данных, хранящихся в БД конструкции. Вычисляемым столбцам следует давать название с помощью ключевого слова **AS**.

Вычисляемый столбец можно создать как: <Новое поле> = <выражение>

Если название столбца состоит из нескольких слов, разделенных пробелами, следует их записать в квадратных скобках: [].

Сортировка данных выполняется с помощью команды **ORDER BY**, которая добавляется в конец запроса, после чего перечисляется список столбцов. Для каждого столбца указывается тип сортировки **ASC** | **DESC** (ascending – по возрастанию | descending – по убыванию). **ASC** – по умолчанию, можно не указывать.

Конструкция **TOP <N>** позволяет выбрать определенное количество строк из таблицы. Дополнительный оператор **PERCENT** позволяет выбрать процентное количество строк из таблицы. Дополнительный оператор **WITH TIES** позволяет выбрать все строки с такими же свойствами.

Конструкция **OFFSET <N> ROWS** указывает число строк, которые необходимо пропустить, прежде чем будет начат возврат строк из выражения запроса.

Конструкция **FETCH NEXT <N> ROWS ONLY** указывает число строк, возвращаемых после обработки предложения **OFFSET**.

На языке **T-SQL** регистр не имеет значение (case insensitive).

3. Практическая часть

Дана таблица *Академики*:

ФИО	Дата_рождения	Специализация	Год_присвоения_звания
Аничков Николай Николаевич	1885-11-03	медицина	1939
Бартольд Василий Владимирович	1869-11-15	историк	1913
Белопольский Аристарх Аполлонович	1854-07-13	астрофизик	1903
Бородин Иван Парфеньевич	1847-01-30	ботаник	1902
Вальден Павел Иванович	1863-07-26	химик-технолог	1910
Вернадский Владимир Иванович	1863-03-12	геохимик	1908
Виноградов Павел Гаврилович	1854-11-30	историк	1914
Ипатьев Владимир Николаевич	1867-11-21	химик	1916
Истрин Василий Михайлович	1865-02-22	филолог	1907
Карпинский Александр Петрович	1847-01-07	геолог	1889
Коковцов Павел Константинович	1861-07-01	историк	1906
Курнаков Николай Семёнович	1860-12-06	химик	1913
Марр Николай Яковлевич	1865-01-06	лингвист	1912
Насонов Николай Викторович	1855-02-26	зоолог	1906
Ольденбург Сергей Фёдорович	1863-09-26	историк	1903
Павлов Иван Петрович	1849-09-26	физиолог	1907
Перетц Владимир Николаевич	1870-01-31	филолог	1914
Соболевский Алексей Иванович	1857-01-07	лингвист	1900
Стеклов Владимир Андреевич	1864-01-09	математик	1912

Пример 1: Вывести список академиков:

```
SELECT
```

```
*
```

```
FROM
```

```
Академики
```

Пример 2: Вывести ФИО и дату рождения всех академиков:

```
SELECT
```

```
ФИО, Дата_рождения
```

```
FROM
```

```
Академики
```

Пример 3: Создайте вычисляемое поле «Информация», содержащее информацию об академике в таком виде: «Академик Петров Петр Петрович, специализация: математика»:

```
SELECT
```

'Академик ' + ФИО + ', специализация: ' + Специализация AS
Информация
FROM
Академики

Пример 4: Вывести ФИО академиков и номер следующего года после присвоения звания:

```
SELECT
    ФИО
    , [Через год] = Год_присвоения_звания + 1
FROM
    Академики
```

Пример 5: Выведите список специализаций, убрав дубликаты:

```
SELECT DISTINCT
    Специализация
FROM
    Академики
```

Пример 6: Вывести список академиков, отсортированный по возрастанию года присвоения звания:

```
SELECT
    *
FROM
    Академики
ORDER BY
    Год_присвоения_звания
```

Пример 7: Вывести список академиков, отсортированный по убыванию специализации и по возрастанию ФИО:

```
SELECT
    *
FROM
    Академики
ORDER BY
    Специализация DESC
    , ФИО ASC
```

Пример 8: Вывести первые две строки из списка академиков, отсортированного по возрастанию ФИО:

```
SELECT TOP 2
    *
FROM
    Академики
ORDER BY
    ФИО ASC
```

Пример 9: Вывести первые 30% строк из списка академиков, отсортированного по возрастанию года присвоения звания:

```
SELECT TOP 30 PERCENT
    *
FROM
    Академики
```

ORDER BY

Год_присвоения_звания

Пример 10: Вывести первые четыре строки и всех академиков с таким годом присвоения звания из списка академиков, отсортированного по возрастанию года присвоения звания:

SELECT TOP 4 WITH TIES

*

FROM

Академики

ORDER BY

Год_присвоения_звания

Пример 11: Вывести, начиная с третьего, список академиков, отсортированный по возрастанию ФИО:

SELECT

*

FROM

Академики

ORDER BY

ФИО

OFFSET 2 ROWS

Пример 12: Вывести, начиная с третьего и до десятого, список академиков, отсортированный по возрастанию ФИО:

SELECT

*

FROM

Академики

ORDER BY

ФИО

OFFSET 2 ROWS

FETCH NEXT 8 ROWS ONLY

4. Задание

1. Вывести ФИО, специализацию и дату рождения всех академиков:
2. Создать вычисляемое поле «О присвоении звания», которая содержит информацию об академике в виде: «Петров Петр Петрович получил звание в 1974»
3. Вывести ФИО академиков и вычисляемое поле «Через 5 лет после присвоения звания».
4. Вывести список годов присвоения звания, убрав дубликаты.
5. Вывести список академиков, отсортированный по убыванию даты рождения.
6. Вывести список академиков, отсортированный по убыванию специализации и года присвоения звания, и по возрастанию ФИО.
7. Вывести первую строку из списка академиков, отсортированного по убыванию ФИО.
8. Вывести фамилию академика, который раньше всех получил звания.
9. Вывести первые 10% строк из списка академиков, отсортированного в алфавитном порядке.
10. Вывести первые 5 строк и всех академиков с таким годом присвоения звания из списка академиков, отсортированного по возрастанию года присвоения звания:
11. Вывести, начиная с десятого, список академиков, отсортированный по возрастанию даты рождения:
12. Вывести девятую и десятую строку из списка академиков, отсортированного по возрастанию ФИО.