

# Базы данных и SQL (семинары)

## Урок 1. Установка СУБД, подключение к БД, просмотр и создание таблиц

### Домашнее задание

1. Создайте таблицу с мобильными телефонами (**mobile\_phones**), используя графический интерфейс. Заполните БД данными. Добавьте скриншот на платформу в качестве ответа на ДЗ

id	product_name	manufacturer	product_count	price
1	iPhone X	Apple	3	76000
2	iPhone 8	Apple	2	51000
3	Galaxy S9	Samsung	2	56000
4	Galaxy S8	Samsung	1	41000
5	P20 Pro	Huawei	5	36000

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' panel with a tree view containing 'lesson\_1', 'mobile\_phones', 'stock\_assortment', 'Views', 'Stored Procedures', 'Functions', 'myfirstdb', 'sakila', 'sys', and 'world'. The 'mobile\_phones' schema is selected. The main workspace shows the 'stock\_assortment' table with the following data:

idstock_assort	product_name	manufacturer	product_count	price
1	iPhone X	Apple	3	76000
2	iPhone 8	Apple	2	51000
3	Galaxy S9	Samsung	2	56000
4	Galaxy S8	Samsung	1	41000
5	P20 Pro	Huawei	5	36000

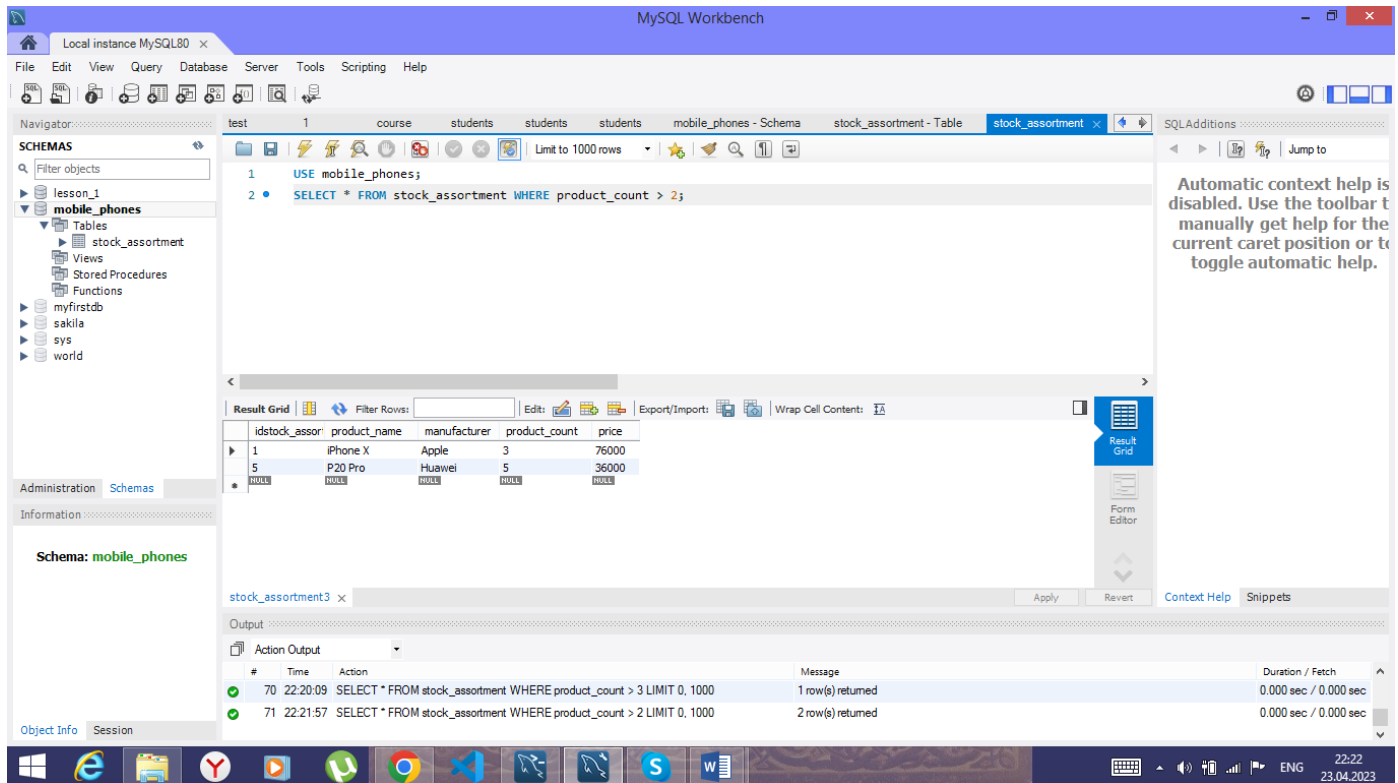
The bottom panel shows the 'Output' tab with the following log entries:

#	Time	Action	Message	Duration / Fetch
95	23:02:10	SELECT * FROM stock_assortment WHERE product_name LIKE '%8%'; LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
96	23:04:33	SELECT * FROM mobile_phones.stock_assortment LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

## 2. Выведите название, производителя и цену для товаров, количество которых превышает 2

USE mobile\_phones;

SELECT \* FROM stock\_assortment WHERE product\_count > 2;



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
```

The result grid displays the following data:

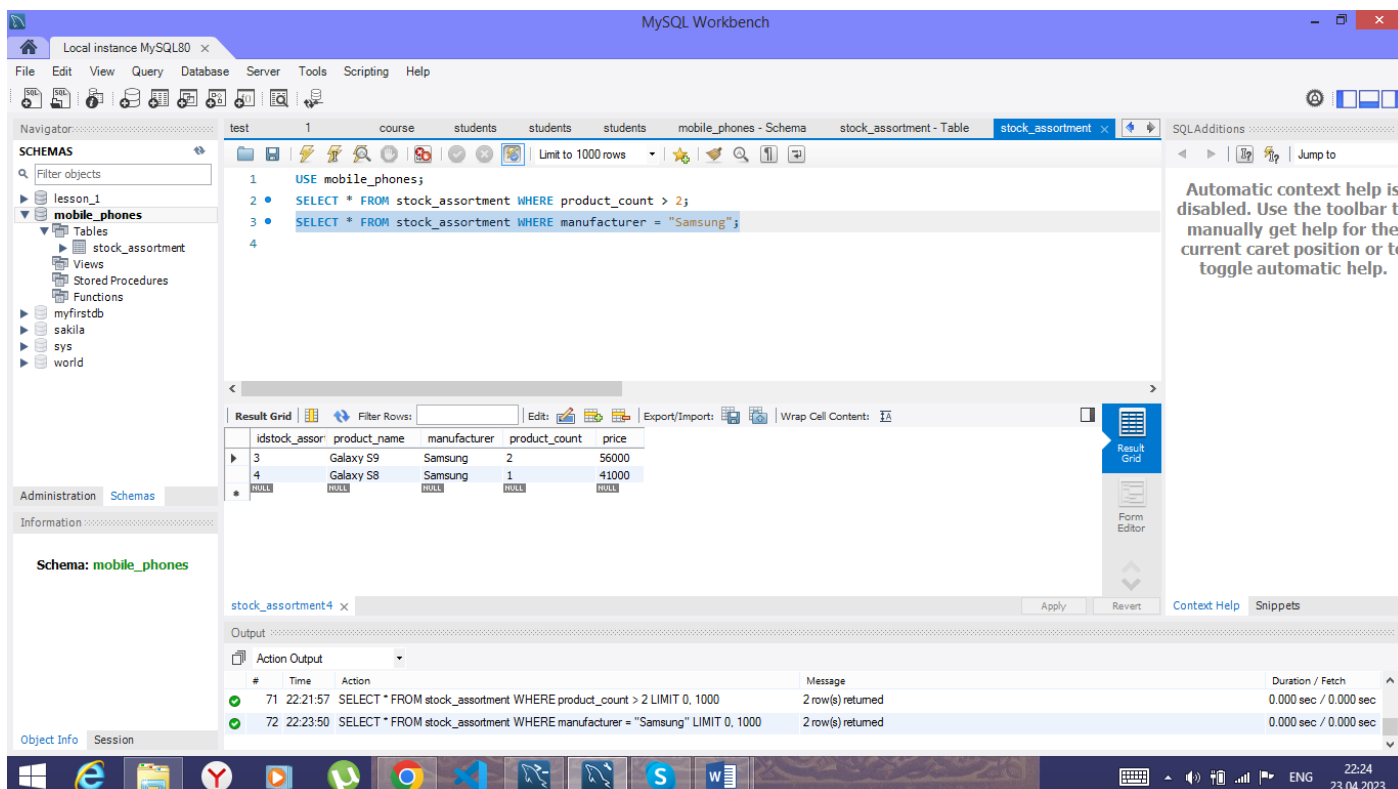
idstock_assor	product_name	manufacturer	product_count	price
1	iPhone X	Apple	3	76000
5	P20 Pro	Huawei	5	36000

The Action Output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
70	22:20:09	SELECT * FROM stock_assortment WHERE product_count > 3 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
71	22:21:57	SELECT * FROM stock_assortment WHERE product_count > 2 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

## 3. Выведите весь ассортимент товаров марки "Samsung"

SELECT \* FROM stock\_assortment WHERE manufacturer = "Samsung";



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
3 SELECT * FROM stock_assortment WHERE manufacturer = "Samsung";
4
```

The result grid displays the following data:

idstock_assor	product_name	manufacturer	product_count	price
3	Galaxy S9	Samsung	2	56000
4	Galaxy S8	Samsung	1	41000

The Action Output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
71	22:21:57	SELECT * FROM stock_assortment WHERE product_count > 2 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
72	22:23:50	SELECT * FROM stock_assortment WHERE manufacturer = "Samsung" LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

#### 4. (по желанию)\* С помощью регулярных выражений найти:

##### 4.1. Товары, в которых есть упоминание "Iphone"

SELECT \* FROM stock\_assortment WHERE product\_name LIKE "iPhone%";

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'mobile\_phones' schema selected. The 'Query Editor' contains the following SQL script:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
3 SELECT * FROM stock_assortment WHERE manufacturer = "Samsung";
4 SELECT * FROM stock_assortment WHERE product_name LIKE "iPhone%";
5
```

The 'Result Grid' shows the results of the fourth query, displaying two rows of iPhone products:

idstock_assor	product_name	manufacturer	product_count	price
1	iPhone X	Apple	3	76000
2	iPhone 8	Apple	2	51000

The 'Output' pane at the bottom shows the execution log with two messages:

- 72 22:23:50 SELECT \* FROM stock\_assortment WHERE manufacturer = "Samsung" LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec
- 73 22:27:56 SELECT \* FROM stock\_assortment WHERE product\_name LIKE "iPhone%" LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec

##### 4.2. Товары, в которых есть упоминание "Samsung"

SELECT \* FROM stock\_assortment WHERE manufacturer LIKE "Samsung%";

The screenshot shows the MySQL Workbench interface. The 'Query Editor' contains the same SQL script as in the previous screenshot, but the fifth query is highlighted:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
3 SELECT * FROM stock_assortment WHERE manufacturer = "Samsung";
4 SELECT * FROM stock_assortment WHERE product_name LIKE "iPhone%";
5 SELECT * FROM stock_assortment WHERE manufacturer LIKE "Samsung%";
```

The 'Result Grid' shows the results of the fifth query, displaying two rows of Samsung products:

idstock_assor	product_name	manufacturer	product_count	price
3	Galaxy S9	Samsung	2	56000
4	Galaxy S8	Samsung	1	41000

The 'Output' pane at the bottom shows the execution log with two messages:

- 73 22:27:56 SELECT \* FROM stock\_assortment WHERE product\_name LIKE "iPhone%" LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec
- 74 22:29:30 SELECT \* FROM stock\_assortment WHERE manufacturer LIKE "Samsung%" LIMIT 0, 1000 2 row(s) returned 0.000 sec / 0.000 sec

### 4.3. Товары, в которых есть ЦИФРЫ

SELECT \* FROM stock\_assortment WHERE product\_name REGEXP '[0-9]';

The screenshot shows the MySQL Workbench interface. The SQL editor contains a query that selects all products from the 'stock\_assortment' table where the product name contains any digit. The query is as follows:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
3 SELECT * FROM stock_assortment WHERE manufacturer = "Samsung";
4 SELECT * FROM stock_assortment WHERE product_name LIKE "iPhone%";
5 SELECT * FROM stock_assortment WHERE manufacturer LIKE "Samsung%";
6 SELECT * FROM stock_assortment WHERE product_name REGEXP '[0-9]';
7 SELECT * FROM stock_assortment WHERE product_name LIKE "%8%";
```

The 'Result Grid' shows the results of the 6th query (SELECT \* FROM stock\_assortment WHERE product\_name REGEXP '[0-9]';). The results are as follows:

idstock_assort	product_name	manufacturer	product_count	price
2	iPhone 8	Apple	2	51000
3	Galaxy S9	Samsung	2	56000
4	Galaxy S8	Samsung	1	41000
5	P20 Pro	Huawei	5	36000

The 'Output' pane shows the execution log for the 6th query, indicating that 4 rows were returned.

### 4.4. Товары, в которых есть ЦИФРА "8"

SELECT \* FROM stock\_assortment WHERE product\_name LIKE '%8%';

The screenshot shows the MySQL Workbench interface. The SQL editor contains a query that selects all products from the 'stock\_assortment' table where the product name contains the digit '8'. The query is as follows:

```
1 USE mobile_phones;
2 SELECT * FROM stock_assortment WHERE product_count > 2;
3 SELECT * FROM stock_assortment WHERE manufacturer = "Samsung";
4 SELECT * FROM stock_assortment WHERE product_name LIKE "iPhone%";
5 SELECT * FROM stock_assortment WHERE manufacturer LIKE "Samsung%";
6 SELECT * FROM stock_assortment WHERE product_name REGEXP '[0-9]';
7 SELECT * FROM stock_assortment WHERE product_name LIKE "%8%";
```

The 'Result Grid' shows the results of the 7th query (SELECT \* FROM stock\_assortment WHERE product\_name LIKE '%8%';). The results are as follows:

idstock_assort	product_name	manufacturer	product_count	price
2	iPhone 8	Apple	2	51000
4	Galaxy S8	Samsung	1	41000

The 'Output' pane shows the execution log for the 7th query, indicating that 2 rows were returned.