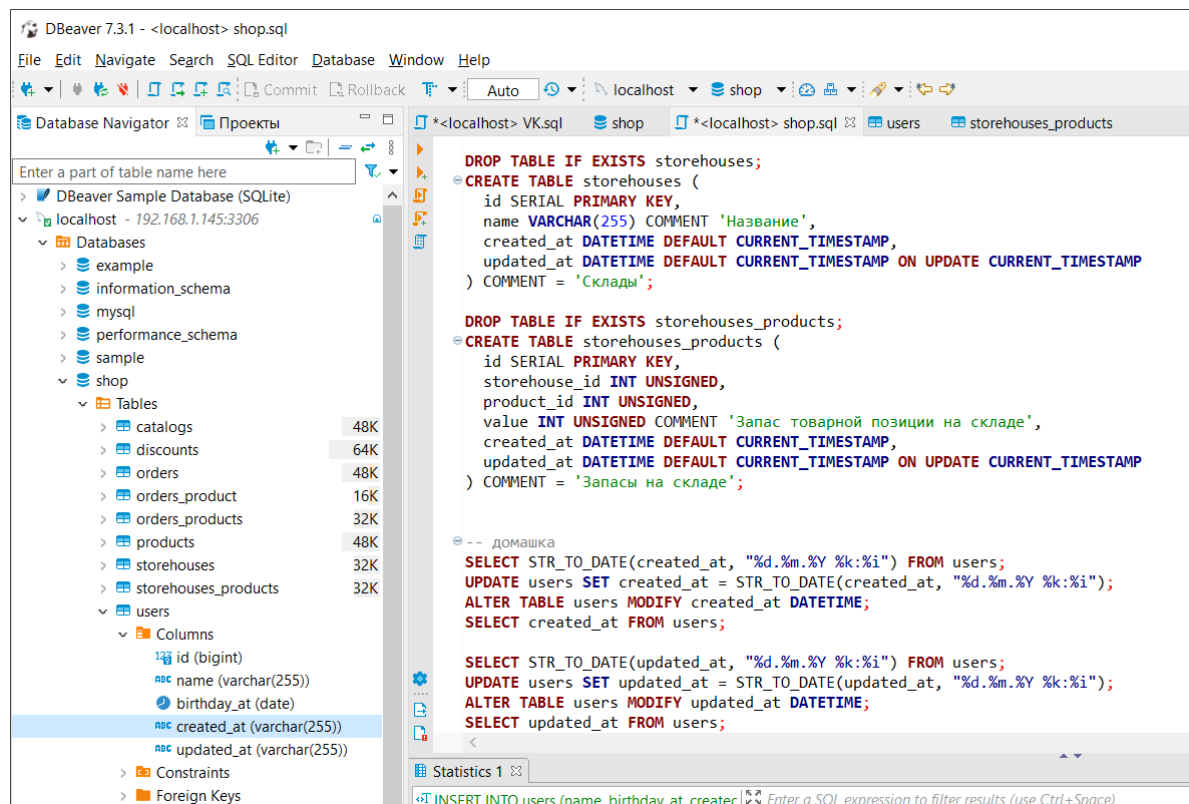


Практическое задание по теме «Операторы, фильтрация, сортировка и ограничение»

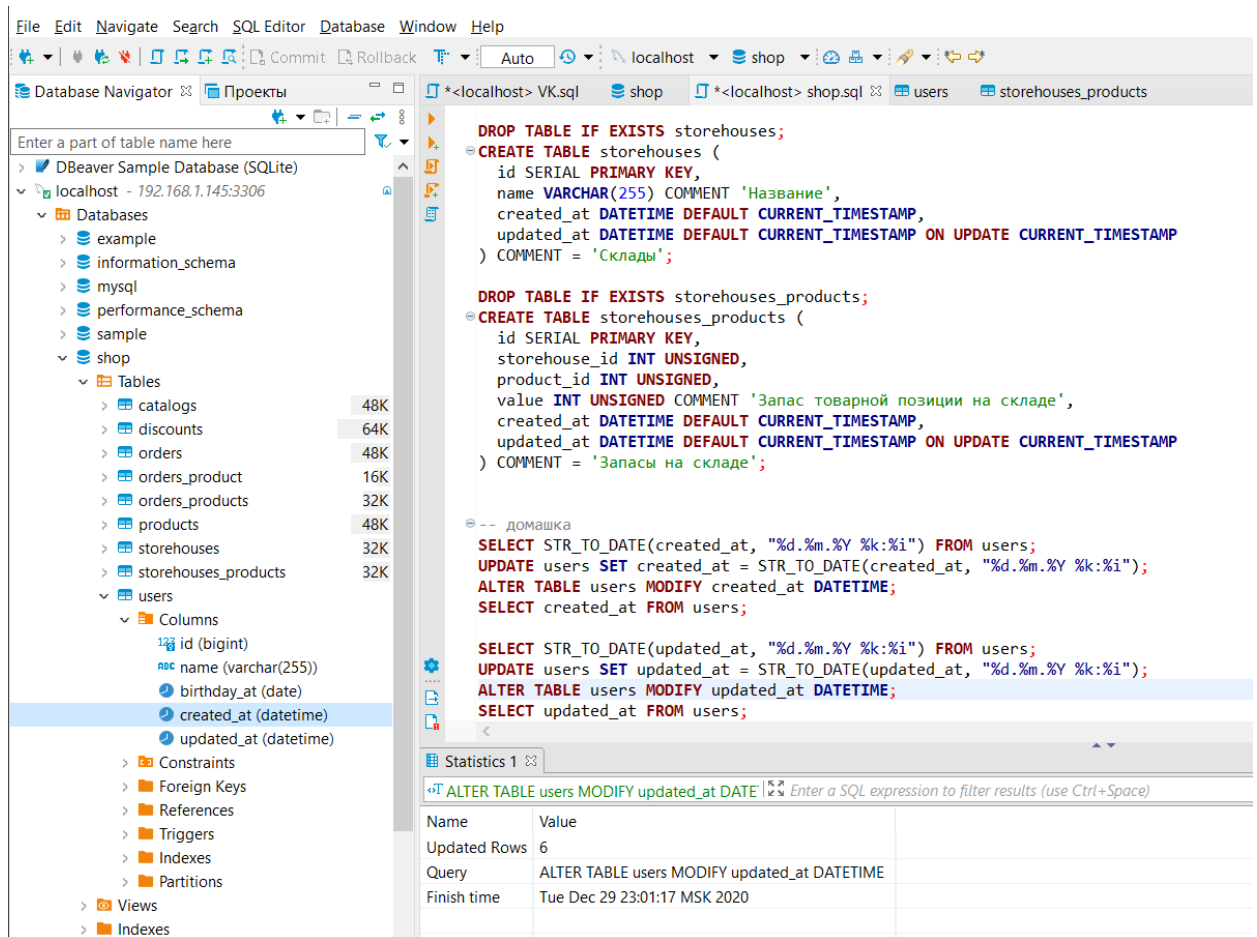
1 задание. Заполнил в таблице users поля created_at и updated_at, сделал их varchar.

2 задание. Таблица users была неудачно спроектирована. Записи created_at и updated_at были заданы типом VARCHAR и в них долгое время помещались значения в формате 20.10.2017 8:10. Необходимо преобразовать поля к типу DATETIME, сохранив введенные ранее значения.



```
SELECT STR_TO_DATE(created_at, "%d.%m.%Y %k:%i") FROM users;
UPDATE users SET created_at = STR_TO_DATE(created_at, "%d.%m.%Y %k:%i");
ALTER TABLE users MODIFY created_at DATETIME;
```

```
SELECT STR_TO_DATE(updated_at, "%d.%m.%Y %k:%i") FROM users;
UPDATE users SET updated_at = STR_TO_DATE(updated_at, "%d.%m.%Y %k:%i");
ALTER TABLE users MODIFY updated_at DATETIME;
```



3 задание. В таблице складских запасов `storehouses_products` в поле `value` могут встречаться самые разные цифры: 0, если товар закончился и выше нуля, если на складе имеются запасы. Необходимо отсортировать записи таким образом, чтобы они выводились в порядке увеличения значения `value`. Однако нулевые запасы должны выводиться в конце, после всех записей

Сделал вот так:

```

SELECT * FROM storehouses_products
ORDER BY CASE WHEN value = 0 THEN 1 ELSE 0 end, value;

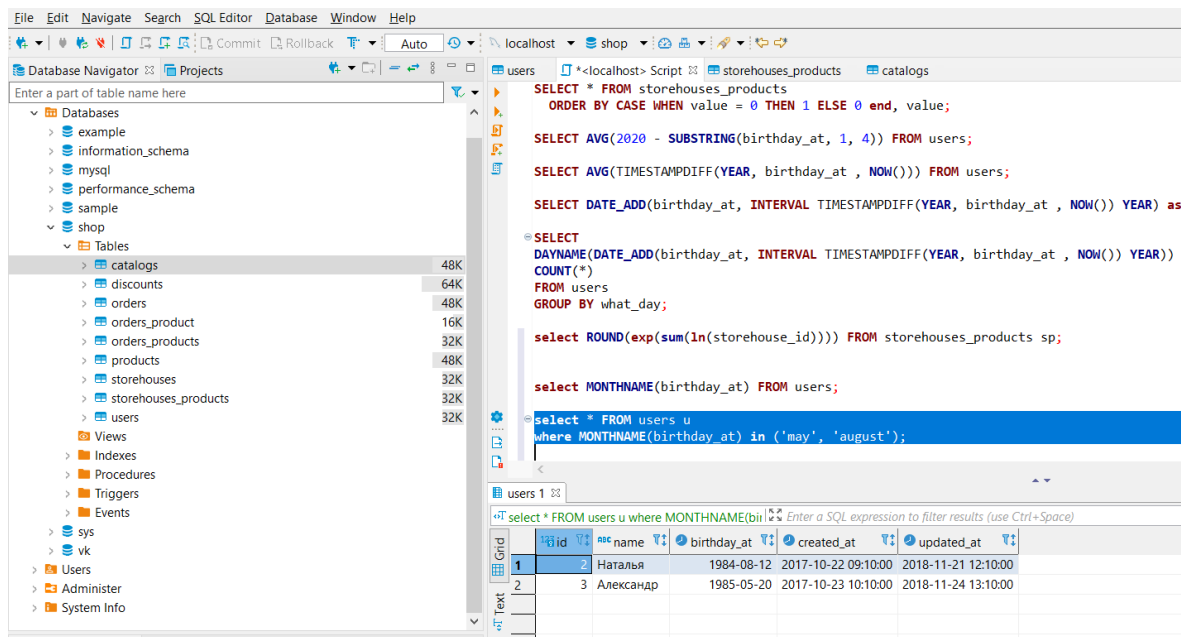
```

4 задание. (по желанию) Из таблицы `users` необходимо извлечь пользователей, родившихся в августе и мае. Месяцы заданы в виде списка английских названий (`may`, `august`)

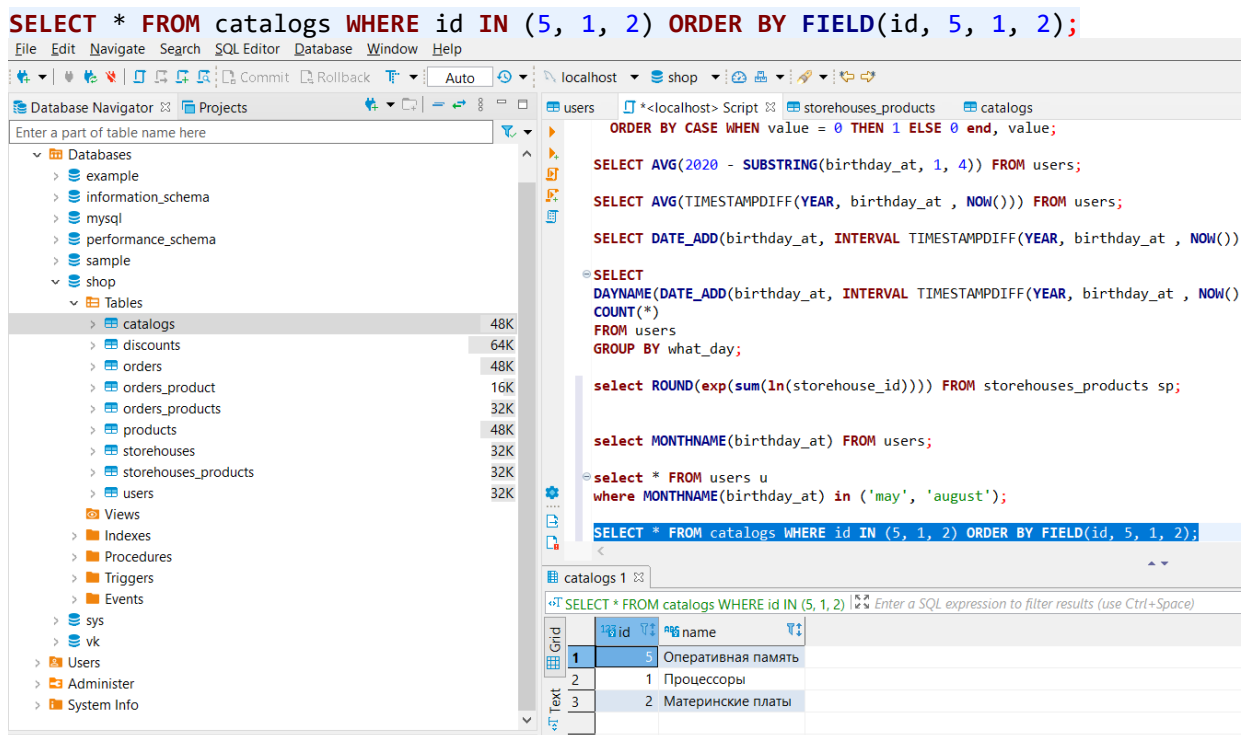
```

select * FROM users where MONTHNAME(birthday_at) in ('may', 'august');

```



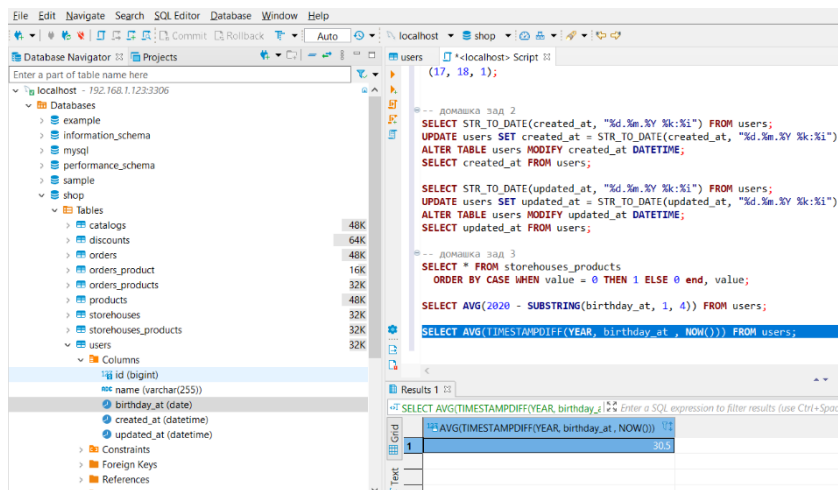
5 задание. (по желанию) Из таблицы catalogs извлекаются записи при помощи запроса. SELECT * FROM catalogs WHERE id IN (5, 1, 2); Отсортируйте записи в порядке, заданном в списке IN



Практическое задание по теме «Операторы, фильтрация, сортировка и ограничение»

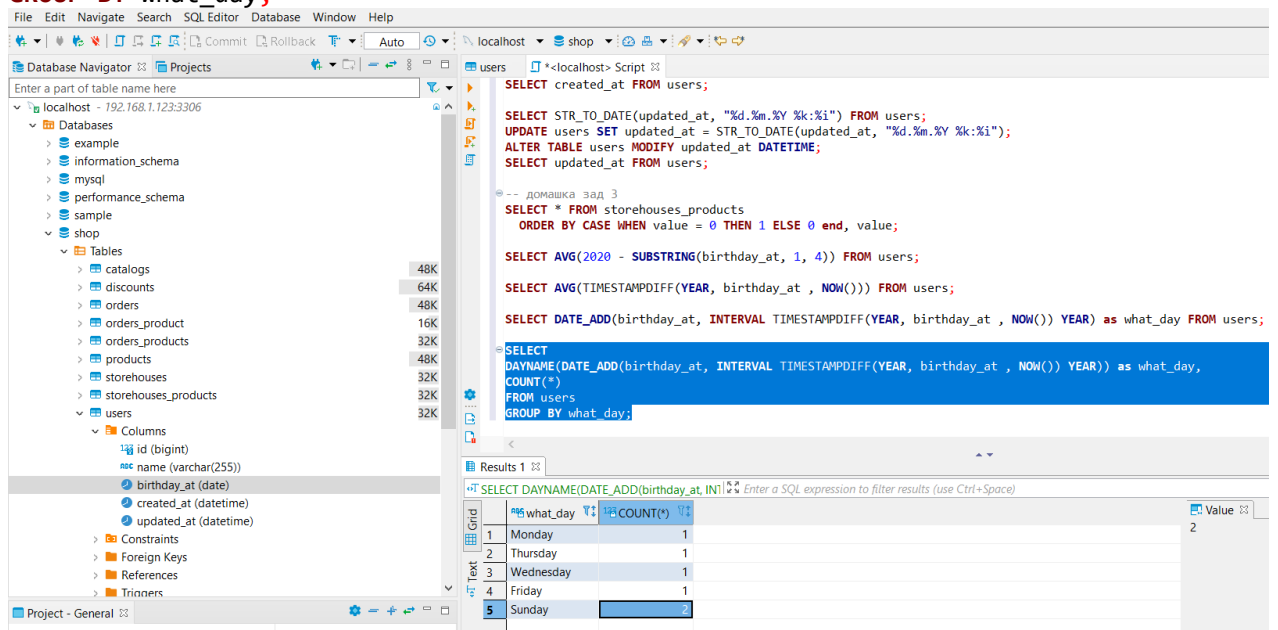
1 Задание. Средний возраст пользователей в таблице users посчитал командой:

```
SELECT AVG(TIMESTAMPDIFF(YEAR, birthday_at , NOW())) FROM users;
```



2 задание. Подсчитайте количество дней рождения, которые приходятся на каждый из дней недели. Следует учесть, что необходимы дни недели текущего года, а не года рождения.

```
SELECT
DAYNAME(DATE_ADD(birthday_at, INTERVAL TIMESTAMPTDIFF(YEAR, birthday_at , NOW())
YEAR)) as what_day,
COUNT(*)
FROM users
GROUP BY what_day;
```



3 задание. (по желанию) Подсчитайте произведение чисел в столбце таблицы. Нашел только вот такое решение. Например, посчитал произведение столбца storehouse_id таблицы storehouses_products:

```
select ROUND(exp(sum(ln(storehouse_id)))) FROM storehouses_products
```