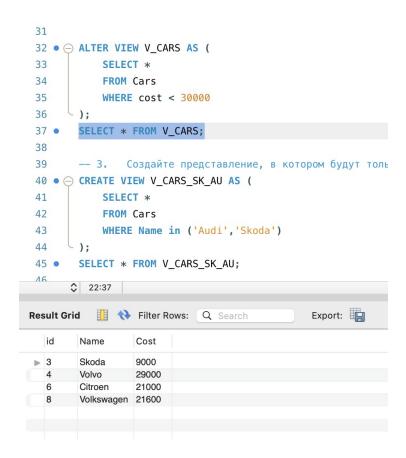
```
19
       ('Volkswagen', 21600);
 20
SELECT *
22
 23
          FROM Cars
 24
          WHERE cost < 25000
     );
 25
 26
      SELECT * FROM V_CARS;
27 •
28
29
       -- 2. Изменить в существующем представлении
     ♦ 22:27
Result Grid III 💎 Filter Rows: Q Search
                                        Export:
      Name
               Cost
▶ 3
       Skoda
               9000
  4
       Volvo
               29000
  6
       Citroen
               21000
       Volkswagen 21600
```



```
-- 3. Создайте представление, в котором будут только автомобили марки "Шкод
40 • 

CREATE VIEW V_CARS_SK_AU AS (
41
              SELECT *
              FROM Cars
42
               WHERE Name in ('Audi', 'Skoda')
43
44
        );
45 •
         SELECT * FROM V_CARS_SK_AU;
46
47
          -- Задача 4 Вывести название и цену для всех анализов, которые продавались 5
48
49 • ○ CREATE TABLE Analysis (
              an_id INT AUTO_INCREMENT PRIMARY KEY,
50
        28:45 VARCHAR(50)
 51
tesult Grid 🎚 🛟 Filter Rows: 🔾 Search
                                                              Export:
   id
          Name Cost
▶ 1
          Audi
                 52642
      Skoda 9000
L01 •
      SELECT a.an_name, a.an_cost, a.an_price,o.ord_datetime,WEEK(o.ord_datetime,1) as week_number, DAYNAME(o.ord_datetime) as week_name
      FROM Analysis a

INNER JOIN Orders o ON a.an_id = o.ord_an

WHERE (YEAR(ord_datetime)=2020 and MONTH(ord_datetime)= 2 and DAY(ord_datetime) = 5)

OR (WEEK(ord_datetime,1) = WEEK('2020-02-05',1)+1)
103
105
L06
      ORDER BY o.ord_datetime;
LO8 — Задача 5
LO9 • ⊖ CREATE TABLE Station (
    id INT AUTO_INCREMENT PRIMARY KEY,
110
                                 Export:
Result Grid 🎚 🛟 Filter Rows: Q Search
           an_cost an_price ord_datetime
                                       week_number week_name
 Кровь с вены 67 223 2020-02-05 12:00:00 6
УЗИ почек 72 203 2020-02-05 14:00:00 6
Mult gest-plac/sac NOS 61 202 2020-02:00 6:00:00:00 7
УЗИ почек 72 203 2020-02:11 11:00:00 7
КТ вн органов 81 162 2020-02:11 15:00:00 7
 124 •
           SELECT *,
  125 — CASE WHEN
                TIMEDIFF(LEAD(station_time, 1, '') OVER (PARTITION
  126
  127
  128
                 TIME_FORMAT(TIMEDIFF(LEAD(station_time, 1, '') OVE
  129
          END as 'time_to_the_next_station'
  130
            FROM Station;
  131
           $ 18:108
 Export:
     id
            train_id station
                                     station_ti... time_to_the_next_stat...
   ▶ 1
             110
                      San Francisko 10:00:00
                                                  00:54:00
     2
             110
                      Redwood City 10:54:00
                                                   00:08:00
     3
             110
                      Palo Alto
                                      11:02:00
                                                   01:33:00
     4
             110
                      San Jose
                                      12:35:00
             120
                      San Francisko 11:00:00
                                                   01:49:00
     6
             120
                      Palo Alto
                                     12:49:00
                                                   00:41:00
             120
                      San Jose
                                      13:30:00
```