

Задание 99. “Посчитай доход с женской аудитории (доход = сумма(price * items)). Обратите внимание, что в таблице женская аудитория имеет поле user_gender «female» или «f».”

Выделяем сумму произведений столбцов price * items из таблицы Purchases, под именем income_from_female. Выводим только значения с столбцом user_gender начинающиеся на “f”.

```
SELECT SUM(price * items) as income_from_female
FROM Purchases
WHERE user_gender LIKE 'f%'
```

The screenshot shows the SQL Academy online trainer interface. The task description on the left asks to calculate the income from the female audience using the formula: income = sum(price * items). It specifies that the female audience is identified by user_gender 'female' or 'f'. The expected result format is a table with one column, income_from_female, and one row with the value 41000. The SQL query entered in the center is: `SELECT SUM(price * items) as income_from_female FROM Purchases WHERE user_gender LIKE 'f%'`. A green button indicates the solution is correct. The result table on the right shows one row with the value 10483. The database schema on the far right shows the Purchases table with columns: user_id, user_gender (VARCHAR), items, and price.

SQL ACADEMY Курс Тренажёр Ещё ▾ ★ Премиум Войти

Задание 99

Посчитай доход с женской аудитории (доход = сумма(price * items)). Обратите внимание, что в таблице женская аудитория имеет поле user_gender «female» или «f».

Поля в результирующей таблице: `income_from_female`

Формат ожидаемого результата запроса:

	income_from_female
1	41000

Решение задания

Решение верно

Результат запроса

	income_from_female
1	10483

Показать таблицу ▾

Purchases

- user_id
- user_gender VARCHAR
- items
- price

Задание 8. “ В какие города можно улететь из Парижа (Paris) и сколько времени это займёт?

Выделяем столбец town_to и разницу во времени между time_in и time_out из таблицы Trip, под именем flight_time. Выводим только значения с столбцом town_from = 'Paris'.

```
SELECT town_to, TIMEDIFF(time_in, time_out) AS flight_time
FROM Trip
WHERE town_from = 'Paris';
```

The screenshot shows the SQL Academy online SQL trainer interface. The browser address bar displays <https://sql-academy.org/ru/trainer/tasks/8>. The page header includes the SQL Academy logo, navigation links (Курс, Тренажёр, Ещё), and user options (Премиум, Войти).

The task description on the left asks: "В какие города можно улететь из Парижа (Paris) и сколько времени это займёт?". It provides instructions on using the `AS` alias for the output column and the `TIMEDIFF` function for time difference calculation.

The SQL query entered in the editor is:

```
1 SELECT town_to, TIMEDIFF(time_in, time_out) AS
   flight_time
2 FROM Trip
3 WHERE town_from = 'Paris';
```

A green button labeled "Решение верно" (Solution is correct) is visible. Below the query, the results of the query are displayed in a table:

town_to	flight_time
Rostov	03:33:00
London	01:00:00

On the right side of the interface, there is a database schema diagram showing the `Trip` and `Company` tables. The `Trip` table has columns: `id` (INT), `company` (INT), `plane` (VARCHAR), `town_from` (VARCHAR), `town_to` (VARCHAR), `time_out` (DATETIME), and `time_in` (DATETIME). The `Company` table has columns: `id` (INT) and `name` (VARCHAR).

Задание 10. “Вывести вылеты, совершенные с 10 ч. по 14 ч. 1 января 1900 г.”

Выделяем все столбцы из таблицы Trip. Выводим только значения со столбцом time_out значения которого входят в диапазон от '1900-01-01 10:00:00' до '1900-01-01 14:00:00'.

```
SELECT * FROM Trip
WHERE time_out
      BETWEEN
      '1900-01-01 10:00:00'
      AND
      '1900-01-01 14:00:00';
```

The screenshot shows the SQL Academy online SQL trainer interface. The task is "Задание 10" (Task 10) with the description "Вывести вылеты, совершенные с 10 ч. по 14 ч. 1 января 1900 г." (Output flights completed from 10 AM to 14 AM on January 1, 1900). The SQL query entered is:

```
1 SELECT * FROM Trip
2 WHERE time_out
3     BETWEEN
4     '1900-01-01 10:00:00'
5     AND
6     '1900-01-01 14:00:00';
```

The query is executed, and the results are displayed in a table. A green button "Решение верно" (Solution is correct) is visible. The table shows the following data:

	id	company	plane	town_from	town_to	time_out
1	1182	1	TU-134	Moscow	Rostov	1900-01-01 10:00:00
2	7772	5	Boeing	Singapore	London	1900-01-01 10:00:00
3	7774	5	Boeing	Singapore	London	1900-01-01 10:00:00

The interface also shows a sidebar with the task description, a "Решение задания" (Solution of the task) section, and a "Последние отправки" (Last submissions) section. The "Решение задания" section contains a message: "Решения заданий доступны только премиум-пользователям." (Task solutions are available only for premium users). The "Последние отправки" section contains a message: "Войдите, чтобы увидеть здесь свои последние отправки" (Log in to see your last submissions here).

Задание 11. “Выведите пассажиров с самым длинным ФИО. Пробелы, дефисы и точки считаются частью имени.”

Выделяем столбец name из таблицы Passenger. Выводим только значения где длина значения столбца name не превышает длину максимального значения из столбца name в таблице Passenger.

SELECT name

FROM Passenger

WHERE LENGTH(name) = (SELECT MAX(LENGTH(name)) FROM Passenger);

The screenshot shows the SQL Academy online trainer interface. The task description is: "Выведите пассажиров с самым длинным ФИО. Пробелы, дефисы и точки считаются частью имени." The goal is to output the names of passengers with the longest full name. The solution is a SQL query:

```
1 SELECT name
2 FROM Passenger
3 WHERE LENGTH(name) = (SELECT MAX(LENGTH(name))
                        FROM Passenger);
```

 The query is executed, and the result is displayed in a table with two rows: "Catherine Zeta-Jones" and "Leonardo Grant-Baker". The interface also shows a sidebar with navigation links and a database schema diagram on the right.

Задание 11

Выведите пассажиров с самым длинным ФИО. Пробелы, дефисы и точки считаются частью имени.

Поля в результирующей таблице: name

Решение задания

Решения заданий доступны только премиум-пользователям.

Последние отправки

Войдите, чтобы увидеть здесь свои последние отправки

Войти или зарегистрироваться

Решение верно

Отправить

Результат запроса

	name
1	Catherine Zeta-Jones
2	Leonardo Grant-Baker

Показать таблицу

Database Schema:

- Trip: id (INT), company (INT), plane (VARCHAR), town_from (VARCHAR), town_to (VARCHAR), time_out (DATETIME), time_in (DATETIME)
- Company: id (INT), name (VARCHAR)

Задание 13. “Вывести имена людей, у которых есть полный тёзка среди пассажиров

Выделяем столбец name из таблицы Passenger. Выводим только последнее подходящее значение по группировке по столбцу name.

```
SELECT name
FROM Passenger
GROUP BY name
HAVING COUNT(name) > 1;
```

The screenshot shows the SQL Academy online SQL trainer interface. The task is "Задание 13" (Task 13), which asks to output the names of people who have a full duplicate among passengers. The solution is a SQL query: `SELECT name FROM Passenger GROUP BY name HAVING COUNT(name) > 1;`. The interface shows the query is correct ("Решение верно") and displays the result table with one row: "Bruce Willis". The interface also shows the database schema for the "Trip" and "Company" tables.

SQL ACADEMY Курс Тренажёр Ещё

★ Премиум Войти

Задание 13

Вывести имена людей, у которых есть полный тёзка среди пассажиров

Поля в результирующей таблице: name

Решение задания

Решения заданий доступны только премиум-пользователям.

Последние отправки

Войдите, чтобы увидеть здесь свои последние отправки

Войти или зарегистрироваться

```
1 SELECT name
2 FROM Passenger
3 GROUP BY name
4 HAVING COUNT(name) > 1;
5
```

Решение верно

Отправить

Результат запроса

	name
1	Bruce Willis

Показать таблицу

Trip

id	company	plane	town_from	town_to	time_out	time_in
----	---------	-------	-----------	---------	----------	---------

Company

id	name
----	------

Задание 16. “Вывести отсортированный по количеству перелетов (по убыванию) и имени (по возрастанию) список пассажиров, совершивших хотя бы 1 полет.”

Выделяем столбец name и считаем количество всех столбцов под именем count из таблицы Passenger. Присоединяем таблицу Pass_in_trip через столбец passenger и id в Passenger. Группируем по столбцу passenger, выводим только столбцы со значением trip > 0 и сортируем по количеству в перелетов, а совпадающие по количеству по столбцу name.

```
SELECT name, COUNT(*) AS count
FROM Passenger
JOIN Pass_in_trip
ON Passenger.id=Pass_in_trip.passenger
GROUP BY passenger
HAVING COUNT(trip) > 0
ORDER BY COUNT(trip) DESC, name;
```

The screenshot shows the SQL Academy online trainer interface. On the left, the task description for Task 16 is displayed: "Вывести отсортированный по количеству перелетов (по убыванию) и имени (по возрастанию) список пассажиров, совершивших хотя бы 1 полет." Below the description, it lists the fields in the resulting table: name, count. A note indicates that the "as count" construct should be used for aggregation. The main area shows the SQL query solution, which is highlighted in green. Below the query, a green button indicates "Решение верно" (Solution is correct). To the right of the query, there is a "Показать таблицу" (Show table) button. Below this, the result of the query is displayed as a table with two columns: name and count. The table lists five passengers: Michael Caine (4), Mullah Omar (4), Bruce Willis (3), Harrison Ford (3), and Jennifer Lopez (3). On the right side of the interface, there is a database schema diagram showing tables: Trip, Passenger, and Company, with their respective columns and data types.

Задание 16

Вывести отсортированный по количеству перелетов (по убыванию) и имени (по возрастанию) список пассажиров, совершивших хотя бы 1 полет.

Поля в результирующей таблице: name, count

Используйте конструкцию "as count" для агрегатной функции подсчета количества перелетов. Это необходимо для корректной проверки.

Решение задания

Решения заданий доступны только премиум-пользователям.

```
1 SELECT name, COUNT(*) AS count
2 FROM Passenger
3 JOIN Pass_in_trip
4   ON Passenger.id=Pass_in_trip.passenger
5 GROUP BY passenger
6 HAVING COUNT(trip) > 0
7 ORDER BY COUNT(trip) DESC, name;
```

Решение верно

Отправить

Результат запроса

	name	count
1	Michael Caine	4
2	Mullah Omar	4
3	Bruce Willis	3
4	Harrison Ford	3
5	Jennifer Lopez	3

Показать таблицу

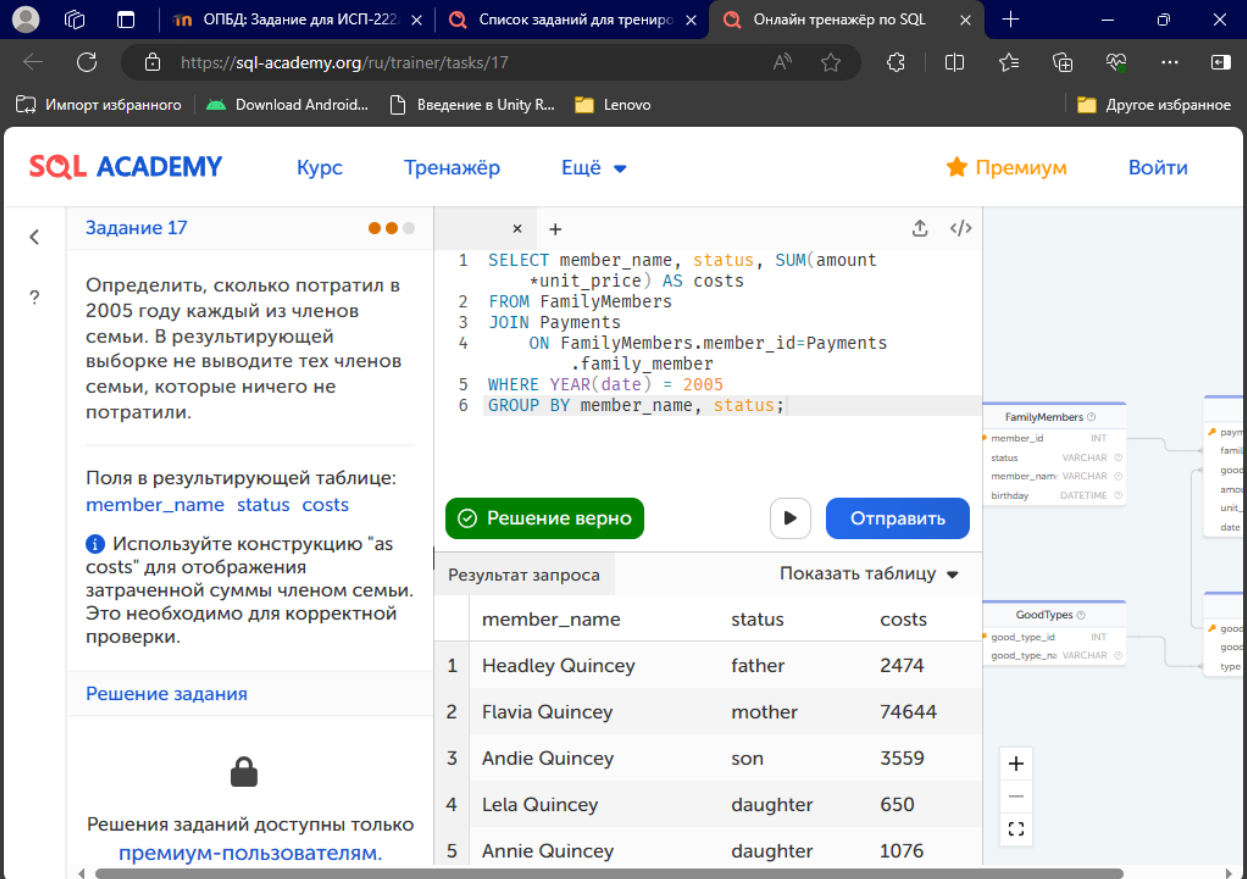
Схема базы данных:

- Trip: id (INT), company (INT), plane (VARCHAR), town_from (VARCHAR), town_to (VARCHAR), time_out (DATETIME), time_in (DATETIME)
- Passenger: id (INT), name (VARCHAR)
- Company: id (INT), name (VARCHAR)

Задание 17. “Определить, сколько потратил в 2005 году каждый из членов семьи. В результирующей выборке не выводите тех членов семьи, которые ничего не потратили.”

Выделяем столбцы member_name, status и сумму произведений значений столбцов amount и unit_price, под именем costs из таблицы FamilyMembers. Присоединяем Payments по столбцу family_member и member_id из FamilyMembers. Выводим только те значения где год в столбце date соответствует 2005. Сортируем по member_name и status;

```
SELECT member_name, status, SUM(amount*unit_price) AS costs
FROM FamilyMembers
JOIN Payments
ON FamilyMembers.member_id=Payments.family_member
WHERE YEAR(date) = 2005
GROUP BY member_name, status;
```



Задание 17

Определить, сколько потратил в 2005 году каждый из членов семьи. В результирующей выборке не выводите тех членов семьи, которые ничего не потратили.

Поля в результирующей таблице: member_name status costs

Используйте конструкцию "as costs" для отображения затраченной суммы членом семьи. Это необходимо для корректной проверки.

Решение задания

Решения заданий доступны только премиум-пользователям.

Решение верно

Результат запроса

	member_name	status	costs
1	Headley Quincey	father	2474
2	Flavia Quincey	mother	74644
3	Andie Quincey	son	3559
4	Lela Quincey	daughter	650
5	Annie Quincey	daughter	1076