

Задание 27. “Узнайте, сколько было потрачено на каждую из групп товаров в 2005 году. Выведите название группы и потраченную на неё сумму. Если потраченная сумма равна нулю, т.е. товары из этой группы не покупались в 2005 году, то не выводите её.

Выделяем столбец `good_type_name` и сумму произведения `amount` и `unit_price` под именем `costs` из таблицы `GoodTypes`. Присоединяем таблицу `Goods` через `good_type_id=type`. Присоединяем таблицу `Payments` через `good_id=good`. Выводим только значения с годом в таблице `date` равным 2005. Сортируем по `good_type_name`.

```
SELECT good_type_name, SUM(amount*unit_price) AS costs
FROM GoodTypes
JOIN Goods ON good_type_id=type
JOIN Payments ON good_id=good
WHERE YEAR(date)=2005
GROUP BY good_type_name;
```

The screenshot shows the SQL Academy online trainer interface. On the left, the task description for 'Задание 27' is displayed. The main area shows the SQL query solution, which has been marked as 'Решение верно' (Solution is correct). Below the query, the 'Результат запроса' (Query result) is shown as a table with two columns: 'good_type_name' and 'costs'. The table contains five rows of data. On the right side, there is a database schema diagram showing tables like 'FamilyMembers', 'GoodTypes', 'Goods', and 'Payments' with their respective columns and data types.

Задание 27

Узнайте, сколько было потрачено на каждую из групп товаров в 2005 году. Выведите название группы и потраченную на неё сумму. Если потраченная сумма равна нулю, т.е. товары из этой группы не покупались в 2005 году, то не выводите её.

Поля в результирующей таблице: `good_type_name costs`

Используйте конструкцию "as costs" для отображения затраченной суммы на конкретную группу товаров. Это необходимо для корректной проверки.

Решение задания

```
1 SELECT good_type_name, SUM(amount*unit_price)
   AS costs
2 FROM GoodTypes
3 JOIN Goods ON good_type_id=type
4 JOIN Payments ON good_id=good
5 WHERE YEAR(date)=2005
6 GROUP BY good_type_name;
```

Решение верно

Отправить

Результат запроса

	good_type_name	costs
1	communal payments	4100
2	food	1198
3	delicacies	650
4	entertainment	1495
5	education	6460

Показать таблицу

FamilyMembers

- member_id INT
- status VARCHAR
- member_name VARCHAR
- birthday DATETIME

GoodTypes

- good_type_id INT
- good_type_name VARCHAR

Goods

- good_id INT
- good_name VARCHAR
- unit_price DECIMAL

Payments

- payment_id INT
- good_id INT
- member_id INT
- amount DECIMAL
- date DATETIME

Задание 29. “ Выведите имена пассажиров улетевших в Москву (Moscow) на самолете TU-134”

Выделяем исключая дублирование столбец name из таблицы Passenger. Присоединяем таблицу Pass_in_trip через passenger и id в таблице Passenger. Присоединяем таблицу Trip через id и trip в таблице Passenger. Выводим только значения с town_to равным 'Moscow' и plane равным 'TU-134'.

```
SELECT DISTINCT name
FROM Passenger
JOIN Pass_in_trip
ON Pass_in_trip.passenger=Passenger.id
JOIN Trip
ON Trip.id=Pass_in_trip.trip
WHERE town_to='Moscow' AND plane='TU-134';
```

SQL ACADEMY Курс Тренажёр Ещё

Задание 29

Выведите имена пассажиров улетевших в Москву (Moscow) на самолете TU-134

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 DISTINCT name
2 FROM Passenger
3 JOIN Pass_in_trip
4 ON Pass_in_trip.passenger=Passenger.id
5 JOIN Trip
6 ON Trip.id=Pass_in_trip.trip
7 WHERE town_to='Moscow' AND plane='TU-134';
```

Решение верно

Отправить

Результат запроса

	name
1	Bruce Willis
2	Ray Liotta
3	Nikole Kidman
4	Jennifer Lopez
5	Kurt Russell

Показать таблицу

Схема базы данных:

- Trip
 - id (INT)
 - company (INT)
 - plane (VARCHAR)
 - town_from (VARCHAR)
 - town_to (VARCHAR)
 - time_out (DATETIME)
 - time_in (DATETIME)
- Company
 - id (INT)
 - name (VARCHAR)
- Passenger
 - id (INT)
 - name (VARCHAR)
- Pass_in_trip
 - passenger (INT)
 - trip (INT)

Задание 30. “Выведите нагруженность (число пассажиров) каждого рейса (trip). Результат вывести в отсортированном виде по убыванию нагруженности.”

Выделяем столбец trip, считаем Passenger и выводим под полем count из таблицы Pass_in_trip. Группируем по столбцу trip, и сортируем по count в обратном порядке.

```
SELECT trip, COUNT(Passenger) as count
FROM Pass_in_trip
GROUP BY trip
ORDER BY count DESC;
```

The screenshot shows the SQL Academy online trainer interface. The task description is: "Выведите нагруженность (число пассажиров) каждого рейса (trip). Результат вывести в отсортированном виде по убыванию нагруженности." The solution SQL query is:

```
1 SELECT trip, COUNT(Passenger) as count
2 FROM Pass_in_trip
3 GROUP BY trip
4 ORDER BY count DESC;
```

 The result table shows 5 rows of data. A green button indicates the solution is correct. The interface also shows database schemas for Trip and Company tables.

	trip	count
1	7771	6
2	7772	5
3	1181	4
4	1123	3
5	1145	2

Задание 31. “Вывести всех членов семьи с фамилией Quincey.”

Выделяем все столбцы из таблицы FamilyMembers. Выводим только те значения где столбец member_name заканчивается на “Quincey”.

```
SELECT *
```

```
FROM FamilyMembers
```

```
WHERE member_name LIKE '%Quincey';
```

The screenshot shows the SQL Academy online trainer interface. The task is "Задание 31" (Task 31) with the description "Вывести всех членов семьи с фамилией Quincey." (Output all family members with the surname Quincey). The SQL query entered is:

```
1 SELECT *
2 FROM FamilyMembers
3 WHERE member_name LIKE '%Quincey';
```

The query is executed, and the result is displayed in a table. A green button "Решение верно" (Solution is correct) is visible. The table shows 5 rows of data:

	member_id	status	member_name	birthday
1	1	father	Headley Quincey	1960-05-10
2	2	mother	Flavia Quincey	1963-02-15
3	3	son	Andie Quincey	1983-06-20
4	4	daughter	Lela Quincey	1985-06-25
5	5	daughter	Annie Quincey	1988-04-10

The interface also shows a sidebar with the task description, a "Решение задания" (Solution to the task) button, and a "Войти или зарегистрироваться" (Log in or register) button. The database schema for FamilyMembers is visible on the right:

```
FamilyMembers
├── member_id INT
├── status VARCHAR
├── member_name VARCHAR
└── birthday DATETIME
```

Задание 32. “ Вывести средний возраст людей (в годах), хранящихся в базе данных. Результат округлите до целого в меньшую сторону.”

Выделяем разность дат сегодня и birthday из таблицы FamilyMembers, преобразуем делением под годовой формат с плавающей запятой. Округляем до целого в меньшую сторону. Считаем среднее по всему столбцу и округляем значение в меньшую сторону. Выводим под age.

```
SELECT FLOOR(AVG(FLOOR(DATEDIFF(NOW(), birthday)/365))) AS age
FROM FamilyMembers;
```

The screenshot shows the SQL Academy online trainer interface. The task description on the left states: "Вывести средний возраст людей (в годах), хранящихся в базе данных. Результат округлите до целого в меньшую сторону." The solution SQL query is entered in the central editor: `1 SELECT FLOOR(AVG(FLOOR(DATEDIFF(NOW(), birthday)/365))) AS age FROM FamilyMembers;`. A green button indicates "Решение верно" (Solution is correct). Below the query, the result of the query is displayed in a table with one row:

	age
1	46

. On the right, the database schema for FamilyMembers is shown, including fields: member_id (INT), status (VARCHAR), member_name (VARCHAR), and birthday (DATETIME). A message at the bottom states: "Решения заданий доступны только премиум-пользователям." (Task solutions are available only for premium users).

Задание 33. “ Найдите среднюю цену икры на основе данных, хранящихся в таблице Payments. В базе данных хранятся данные о покупках красной (red caviar) и черной икры (black caviar). В ответе должна быть одна строка со средней ценой всей купленной когда-либо икры..”

Выделяем среднее значение со столбца unit_price под cost из таблицы Payments. Выводим только те значения где таблица good входит в столбец good_id из Goods в котором good_name заканчивается на caviar.

```
SELECT AVG(unit_price) AS cost
FROM Payments
WHERE good IN (SELECT good_id FROM Goods
WHERE good_name LIKE '%caviar');
```

The screenshot shows the SQL Academy online trainer interface. On the left, the task description for 'Задание 33' is displayed. The main area shows the SQL query being executed, which is the same as the one provided in the text. 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 shown in a table with one row and one column, 'cost', with the value '170'. On the right side of the interface, there is a database schema diagram showing tables like FamilyMembers, GoodTypes, and Payments with their respective columns and data types.

SQL ACADEMY Курс Тренажёр Ещё ▾ ★ Премиум Войти

Задание 33

Найдите среднюю цену икры на основе данных, хранящихся в таблице Payments. В базе данных хранятся данные о покупках красной (red caviar) и черной икры (black caviar). В ответе должна быть одна строка со средней ценой всей купленной когда-либо икры.

Поля в результирующей таблице: cost

Используйте конструкцию "as cost" для агрегатной функции подсчета средней цены икры. Это необходимо для корректной проверки.

Решение задания

```
1 SELECT AVG(unit_price) AS cost
2 FROM Payments
3 WHERE good IN (SELECT good_id FROM Goods
4 WHERE good_name LIKE '%caviar');
```

Решение верно

Отправить

Результат запроса

cost
1 170

Показать таблицу ▾

FamilyMembers

- member_id INT
- status VARCHAR
- member_name VARCHAR
- birthday DATETIME

GoodTypes

- good_type_id INT
- good_type_name VARCHAR

Payments

- payment_id INT
- good_id INT
- amount DECIMAL
- unit_price DECIMAL
- date DATETIME

Задание 35. “ Сколько различных кабинетов школы использовались 2 сентября 2019 года для проведения занятий?”

Выделяем количество значений из столбца classroom в таблице Schedule под именем count. Выводим только те значения где date равен '2019-09-02'.

```
SELECT COUNT(classroom) AS count
```

```
FROM Schedule
```

```
WHERE date='2019-09-02';
```

The screenshot shows the SQL Academy online trainer interface. The task description on the left asks for the number of different classrooms used on September 2, 2019. The SQL query in the center is:

```
1 SELECT COUNT(classroom) AS count
2 FROM Schedule
3 WHERE date='2019-09-02';
```

 A green button indicates the solution is correct. The result table on the right shows one row with the count 5. The database schema on the far right includes tables like Class and Schedule. A message at the bottom states that solutions are only available for premium users.

Задание 35

Сколько различных кабинетов школы использовались 2 сентября 2019 года для проведения занятий?

Поля в результирующей таблице:
count

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

Решение задания

Решения заданий доступны только премиум-пользователям.

```
1 SELECT COUNT(classroom) AS count
2 FROM Schedule
3 WHERE date='2019-09-02';
```

Решение верно

Отправить

Результат запроса

	count
1	5

Показать таблицу

Class

- id (INT)
- name (VARCHAR)

Schedule

- id (INT)
- date (DATE)
- class (INT)
- number_pair (INT)
- teacher (INT)
- subject (INT)
- classroom (INT)

https://sql-academy.org/ru/premium