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

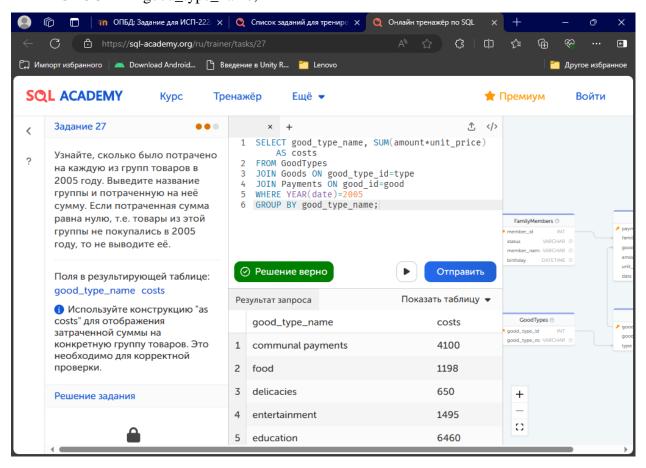
FROM GoodTypes

JOIN Goods ON good_type_id=type

JOIN Payments ON good_id=good

WHERE YEAR(date)=2005

GROUP BY good_type_name;



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

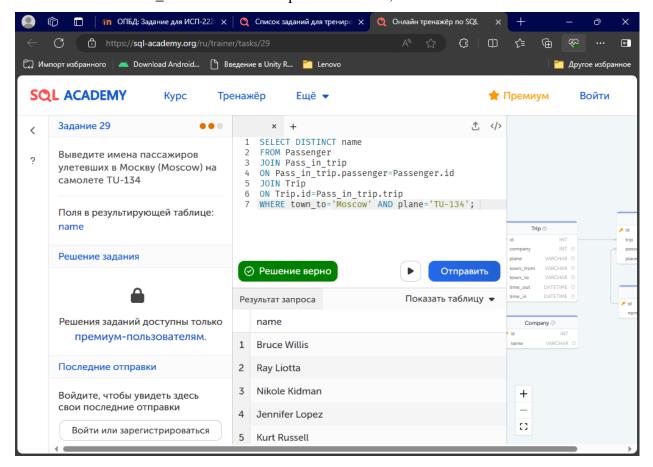
JOIN Pass_in_trip

ON Pass_in_trip.passenger=Passenger.id

JOIN Trip

ON Trip.id=Pass_in_trip.trip

WHERE town_to='Moscow' AND plane='TU-134';



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

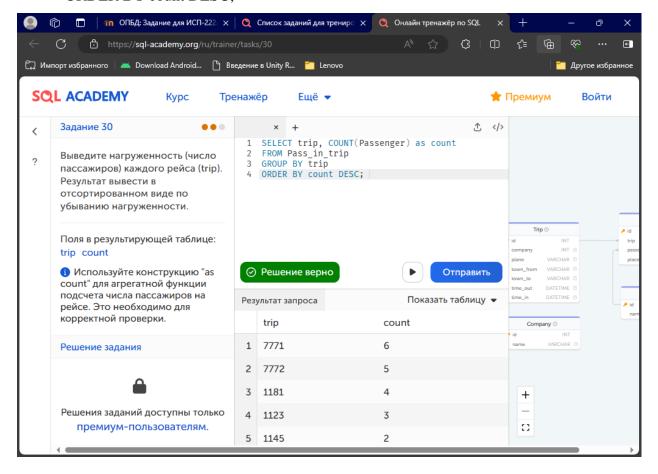
Выделяем столбец trip, считаем Passenger и выводим под полем count из таблицы Pass_in_trip. Группируем по столбцу trip, и сортируем по count в обратном порядке.

SELECT trip, COUNT(Passenger) as count

FROM Pass_in_trip

GROUP BY trip

ORDER BY count DESC;



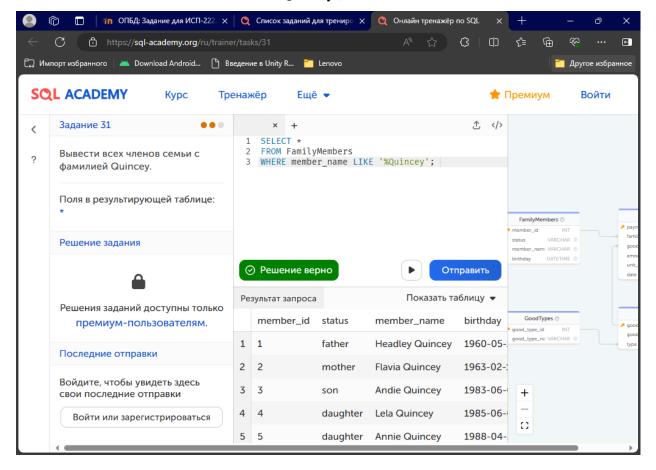
Задание 31. "Вывести всех членов семьи с фамилией Quincey."

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

SELECT *

FROM FamilyMembers

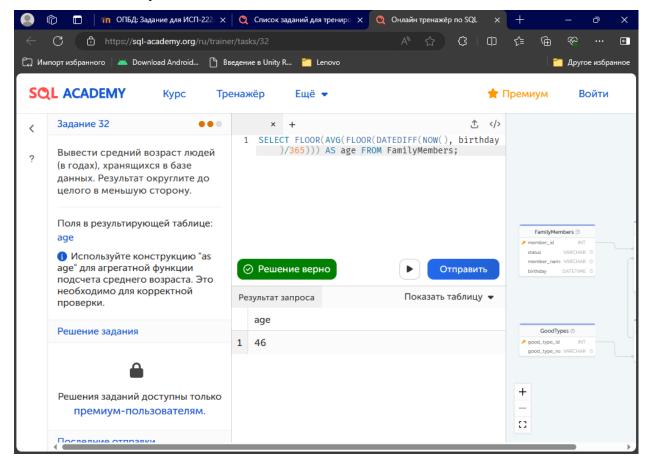
WHERE member_name LIKE '%Quincey';



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;



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.."

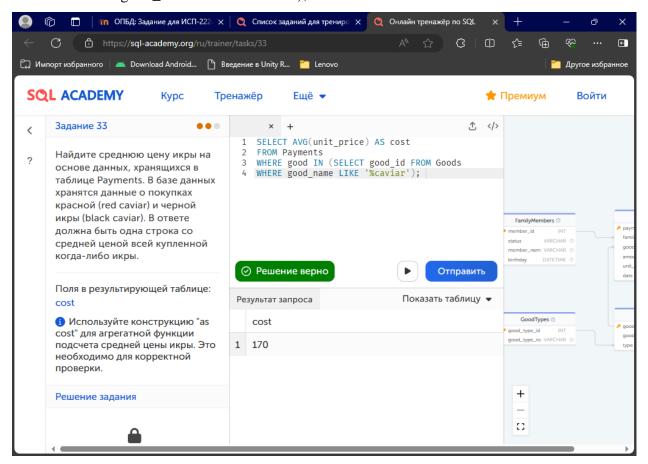
Выделяем среднее значение со столбца unit_price под cost из таблицы Payments. Выводим только те значения где таблица good входит в столбец good_id из Goods в котором good_name заканчивается на caviar.

SELECT AVG(unit_price) AS cost

FROM Payments

WHERE good IN (SELECT good_id FROM Goods

WHERE good_name LIKE '%caviar');



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 тоько те значения где date paseн '2019-09-02'.

SELECT COUNT(classroom) AS count

FROM Schedule

WHERE date='2019-09-02';

