https://sql-academy.org/ru/profile/21144

БД «Авиаперелеты»

|  |  |
| --- | --- |
| 1. Вывести имена всех когда-либо обслуживаемых пассажиров авиакомпаний  Поля в результирующей таблице: name | SELECT name  FROM Passenger |
| 2. Вывести названия всеx авиакомпаний  Поля в результирующей таблице: name | SELECT name  FROM Company |
| 3. Вывести все рейсы, совершенные из Москвы  Поля в результирующей таблице: \* | SELECT \*  FROM Trip  WHERE town\_from = "Moscow" |
| 4. Вывести имена людей, которые заканчиваются на "man"  Поля в результирующей таблице: name | SELECT name  FROM Passenger  WHERE name LIKE "%man" |
| 5. Вывести количество рейсов, совершенных на TU-134  Поля в результирующей таблице: count  Используйте конструкцию "as count" для агрегатной функции подсчета количества рейсов. Это необходимо для корректной проверки. | SELECT COUNT(\*) as count  FROM Trip  WHERE plane = "TU-134" |
| 6. Какие компании совершали перелеты на Boeing  Поля в результирующей таблице: name | SELECT DISTINCT name  FROM Company, Trip  WHERE plane="Boeing" AND Trip.company = Company.id |
| 7. Вывести все названия самолётов, на которых можно улететь в Москву (Moscow)  Поля в результирующей таблице: plane | SELECT DISTINCT plane  FROM Trip  WHERE town\_to = "Moscow" |
| 8. В какие города можно улететь из Парижа (Paris) и сколько времени это займёт?  Поля в результирующей таблице: town\_to, flight\_time  Используйте конструкцию "as flight\_time" для вывода необходимого времени. Это необходимо для корректной проверки.  Формат для вывода времени: HH:MM:SS | SELECT town\_to, TIMEDIFF(time\_in, time\_out) as flight\_time  FROM Trip  WHERE town\_from = "Paris" |
| 9. Какие компании организуют перелеты с Владивостока (Vladivostok)?  Поля в результирующей таблице: name | SELECT name  FROM Company, Trip  WHERE town\_from = "Vladivostok" AND Company.id = Trip.company |
| 10. Вывести вылеты, совершенные с 10 ч. по 14 ч. 1 января 1900 г.  Поля в результирующей таблице: \* | SELECT \*  FROM Trip  WHERE time\_out BETWEEN "1900-01-01T10:00:00.000Z" AND "1900-01-01T14:00:00.000Z" |
| 11. Вывести пассажиров с самым длинным именем  Поля в результирующей таблице: name | SELECT name  FROM Passenger  ORDER BY LENGTH(name) DESC  LIMIT 1 |
| 12. Вывести id и количество пассажиров для всех прошедших полётов  Поля в результирующей таблице: trip, count  Используйте конструкцию "as count" для агрегатной функции подсчета количества пассажиров. Это необходимо для корректной проверки. | SELECT Trip.id as trip, COUNT(passenger) as count  FROM Trip, Pass\_in\_trip  WHERE Pass\_in\_trip.trip = Trip.id  GROUP BY Trip.id |
| 13. Вывести имена людей, у которых есть полный тёзка среди пассажиров  Поля в результирующей таблице: name | SELECT name  FROM Passenger  GROUP BY name  HAVING COUNT(name) > 1 |
| 14. В какие города летал Bruce Willis  Поля в результирующей таблице: town\_to | SELECT Trip.town\_to  FROM Trip, Pass\_in\_trip, Passenger  WHERE Passenger.name = "Bruce Willis" AND Passenger.id = Pass\_in\_trip.passenger AND Pass\_in\_trip.trip = Trip.id |
| 15. Во сколько Стив Мартин (Steve Martin) прилетел в Лондон (London)  Поля в результирующей таблице: time\_in | SELECT time\_in  FROM Trip, Pass\_in\_trip, Passenger  WHERE Passenger.name = "Steve Martin" AND town\_to = "London" AND Trip.id = Pass\_in\_trip.trip AND Pass\_in\_trip.passenger = Passenger.id |
| 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. | SELECT name, COUNT(\*) AS count  FROM Passenger, Pass\_in\_trip  WHERE Passenger.id = Pass\_in\_trip.passenger  GROUP BY Pass\_in\_trip.passenger  ORDER BY COUNT(\*) DESC, name ASC |
| 28. Сколько рейсов совершили авиакомпании с Ростова (Rostov) в Москву (Moscow)?  Поля в результирующей таблице: count  Используйте конструкцию "as count" для агрегатной функции подсчета количества рейсов. Это необходимо для корректной проверки. | SELECT count(\*) as count  FROM Trip  WHERE town\_from = "Rostov" AND town\_to = "Moscow" |
| 29. Выведите имена пассажиров улетевших в Москву (Moscow) на самолете TU-134  Поля в результирующей таблице: name | SELECT DISTINCT name  FROM Passenger, Trip, Pass\_in\_trip  WHERE town\_to = "Moscow" AND plane = "TU-134" AND Passenger.id = Pass\_in\_trip.passenger AND Pass\_in\_trip.trip = Trip.id |
| 30. Выведите нагруженность (число пассажиров) каждого рейса (trip). Результат вывести в отсортированном виде по убыванию нагруженности.  Поля в результирующей таблице: trip, count  Используйте конструкцию "as count" для агрегатной функции подсчета числа пассажиров на рейсе. Это необходимо для корректной проверки. | SELECT trip, COUNT(passenger) as count  FROM Pass\_in\_trip  GROUP BY trip  ORDER BY COUNT(passenger) DESC |
| 55. Удалить компании, совершившие наименьшее количество рейсов. | DELETE FROM Company  WHERE Company.id IN (  SELECT company  FROM Trip  GROUP BY company  HAVING COUNT(id) =  (SELECT MIN(count)  FROM (  SELECT COUNT(id) AS count  FROM Trip  GROUP BY company  ) AS min\_count  )  ) |
| 56. Удалить все перелеты, совершенные из Москвы (Moscow). | DELETE FROM Trip  WHERE town\_from = "Moscow" |
| 64. Выведите имена всех пар пассажиров, летевших вместе на одном рейсе два или более раз, и количество таких совместных рейсов. В passengerName1 разместите имя пассажира с наименьшим идентификатором.  Поля в результирующей таблице: passengerName1, passengerName2, count  Используйте конструкции "as passengerName1" и "as passengerName2" для вывода имен пассажиров.  Используйте конструкцию "as count" для агрегатной функции подсчета количества рейсов. Это необходимо для корректной проверки. |  |

БД «Семья»

|  |  |
| --- | --- |
| 17. Определить, сколько потратил в 2005 году каждый из членов семьи  Поля в результирующей таблице: member\_name, status, costs  Используйте конструкцию "as costs" для отображения затраченной суммы членом семьи. Это необходимо для корректной проверки. | SELECT member\_name, status, SUM(unit\_price\*amount) as costs  FROM FamilyMembers, Payments  WHERE FamilyMembers.member\_id=Payments.family\_member AND YEAR(Payments.date)=2005  GROUP BY Payments.family\_member |
| 18. Узнать, кто старше всех в семьe  Поля в результирующей таблице: member\_name | SELECT member\_name  FROM FamilyMembers  WHERE birthday=  (SELECT MIN(birthday)  FROM FamilyMembers) |
| 19. Определить, кто из членов семьи покупал картошку (potato)  Поля в результирующей таблице: status | SELECT DISTINCT status  FROM FamilyMembers, Payments, Goods  WHERE good\_name="potato" AND FamilyMembers.member\_id=Payments.family\_member AND Payments.good=Goods.good\_id |
| 20. Сколько и кто из семьи потратил на развлечения (entertainment). Вывести статус в семье, имя, сумму  Поля в результирующей таблице: status, member\_name, costs  Используйте конструкцию "as costs" для отображения затраченной суммы членом семьи. Это необходимо для корректной проверки. | SELECT status, member\_name, SUM(amount\*unit\_price) AS costs  FROM FamilyMembers, Payments, Goods, GoodTypes  WHERE GoodTypes.good\_type\_name="entertainment" AND FamilyMembers.member\_id=Payments.family\_member AND Payments.good=Goods.good\_id AND Goods.type=GoodTypes.good\_type\_id  GROUP BY status, member\_name |
| 21. Определить товары, которые покупали более 1 раза  Поля в результирующей таблице: good\_name | SELECT good\_name  FROM Goods, Payments  WHERE Goods.good\_id=Payments.good  GROUP BY good  HAVING COUNT(good) > 1;  или  SELECT good\_name  FROM Goods  JOIN Payments ON Goods.good\_id=Payments.good  GROUP BY good  HAVING COUNT(good) > 1; |
| 22. Найти имена всех матерей (mother)  Поля в результирующей таблице: member\_name | SELECT member\_name  FROM FamilyMembers  WHERE status = "mother"; |
| 23. Найдите самый дорогой деликатес (delicacies) и выведите его стоимость  Поля в результирующей таблице: good\_name, unit\_price | SELECT good\_name, unit\_price  FROM Goods  JOIN Payments ON (good\_id=good)  JOIN GoodTypes ON (type=good\_type\_id)  WHERE unit\_price = (SELECT MAX(unit\_price)  FROM Payments  JOIN Goods ON (good\_id=good)  JOIN GoodTypes ON (type=good\_type\_id)  WHERE good\_type\_name='delicacies'); |
| 24. Определить кто и сколько потратил в июне 2005  Поля в результирующей таблице: member\_name, costs  Используйте конструкцию "as costs" для отображения затраченной суммы членом семьи. Это необходимо для корректной проверки. | SELECT member\_name, SUM(amount\*unit\_price) AS costs  FROM FamilyMembers  JOIN Payments ON FamilyMembers.member\_id=Payments.family\_member  WHERE MONTH(date)=6 AND YEAR(date)=2005  GROUP BY member\_name |
| 25. Определить, какие товары имеются в таблице Goods, но не покупались в течение 2005 года  Поля в результирующей таблице: good\_name | SELECT good\_name  FROM Goods  WHERE good\_id not in (SELECT good  FROM Payments  WHERE YEAR(date)=2005) |
| 26. Определить группы товаров, которые не приобретались в 2005 году (но покупались когда-либо)  Поля в результирующей таблице: good\_type\_name | SELECT good\_type\_name  FROM GoodTypes  WHERE good\_type\_id not in (SELECT good\_type\_id  FROM GoodTypes  JOIN Goods ON GoodTypes.good\_type\_id=Goods.type  JOIN Payments ON Goods.good\_id=Payments.good  WHERE YEAR(date)=2005) |
| 27. Узнать, сколько потрачено на каждую из групп товаров в 2005 году. Вывести название группы и сумму  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. | SELECT good\_type\_name, SUM(amount\*unit\_price) as costs  FROM GoodTypes  JOIN Goods ON GoodTypes.good\_type\_id=Goods.type  JOIN Payments ON Goods.good\_id=Payments.good  WHERE YEAR(date)=2005  GROUP BY good\_type\_name |
| 31. Вывести всех членов семьи с фамилией Quincey.  Поля в результирующей таблице: \* | SELECT \*  FROM FamilyMembers  WHERE member\_name LIKE "%Quincey" |
| 32. Вывести средний возраст людей (в годах), хранящихся в базе данных. Результат округлите до целого в меньшую сторону.  Поля в результирующей таблице: age  Используйте конструкцию "as age" для агрегатной функции подсчета среднего возраста. Это необходимо для корректной проверки. | SELECT FLOOR(AVG(YEAR(current\_date())-YEAR(birthday))) AS age  FROM FamilyMembers |
| 33. Найдите среднюю стоимость икры. В базе данных хранятся данные о покупках красной (red caviar) и черной икры (black caviar).  Поля в результирующей таблице: cost  Используйте конструкцию "as cost" для агрегатной функции подсчета средней цены икры. Это необходимо для корректной проверки. | SELECT AVG(unit\_price) AS cost  FROM Payments  JOIN Goods ON Payments.good=Goods.good\_id  WHERE good\_name LIKE "%caviar" |
| 51. |  |
| 52. |  |
| 53. |  |
| 54. |  |

БД «Schedule»

|  |  |
| --- | --- |
| 34. |  |