**Задача 1**

Дано: три таблицы.

Destination (id, name, id\_status).

Tickets (id, id\_destination, lowest\_price, highest\_pice).

Status (id, name).

Необходимо: составить запросы, которые помогут получить следующую информацию:

Уникальные названия маршрутов (destination.name), для которых существуют билеты (есть запись в tickets). Вывести только названия.

Дополните предыдущий запрос: ограничьте маршруты статусом «Без визы».

Найдите маршруты, максимальная цена которых выше общей средней. Общая средняя находится как среднее значение lowest\_price и highest\_price. Вывести названия и высшую цену.

1. Запрос для получения уникальных названий маршрутов, для которых существуют билеты: SELECT DISTINCT destination.name

FROM Destination

JOIN Tickets ON Destination.id = Tickets.id\_destination;

1. Дополненный запрос для ограничения маршрутов статусом «Без визы»:

SELECT DISTINCT destination.name

FROM Destination

JOIN Tickets ON Destination.id = Tickets.id\_destination

JOIN Status ON Destination.id\_status = Status.id

WHERE Status.name = 'Без визы';

1. Запрос для поиска маршрутов, максимальная цена которых выше общей средней:

SELECT destination.name, MAX(tickets.highest\_price)

FROM Destination

JOIN Tickets ON Destination.id = Tickets.id\_destination

WHERE (tickets.highest\_price + tickets.lowest\_price)/2 > (SELECT AVG((tickets.highest\_price + tickets.lowest\_price)/2) FROM Tickets);

GROUP BY destination.name;

**Задача2**

Даны 2 таблицы

User (id\_user, user\_name, user\_surname, user\_weight, age)

visits(id\_visit, id\_user, hours\_spent, class\_name, data)

Необходимо: составить запросы, которые помогут получить следующую информацию:

**Необходимо:** составить запросы, которые помогут получить следующую информацию:

Список уникальных классов. Вывести только названия.

Количество часов, проведенных на занятиях, для каждого пользователя. Вывести фамилию, имя и количество часов.

Средний возраст пользователей, посещающих класс Flex.

1. Список уникальных классов:

SELECT DISTINCT class\_name FROM visits;

1. Количество часов, проведенных на занятиях, для каждого пользователя:

SELECT visits.id\_user, user.user\_surname, user.user\_name, SUM(visits.hours\_spent) AS total\_hours

FROM visits INNER JOIN user ON visits.id\_user = user.id\_user

GROUP BY visits.id\_user, user.user\_surname, user.user\_name;

1. Средний возраст пользователей, посещающих класс Flex:

SELECT AVG(user.age) AS average\_age

FROM user

INNER JOIN visits ON user.id\_user = visits.id\_user

WHERE visits.class\_name = 'Flex';

**Задача 3**

Даны 2 таблицы

Book (id\_book, title, id\_author, pages, year\_puplish)

Author (id\_author, fill\_name, century)

Необходимо**:** составить запросы, которые помогут получить следующую информацию:

Уникальные названия всех книг, опубликованных после 1990 года. Вывести только названия.

Для каждого автора найти сумму напечатанных страниц. Вывести полное имя автора и сумму страниц.

Подсчитать количество книг авторов каждого века. Вывести век и количество книг.

1.Запрос для получения уникальных названий всех книг, опубликованных после 1990 года:

SELECT DISTINCT title

FROM Book

WHERE year\_puplish > 1990;

2. Запрос для получения полного имени автора и суммы напечатанных страниц для каждого автора:

SELECT Author.fill\_name, SUM(Book.pages) AS total\_pages

FROM Author JOIN Book ON Author.id\_author = Book.id\_author

GROUP BY Author.fill\_name;

3.Запрос для подсчета количества книг авторов каждого века:

SELECT Author.century, COUNT(Book.id\_book) AS book\_count

FROM Author JOIN Book ON Author.id\_author = Book.id\_author

GROUP BY Author.century;