Important Notes:

All queries should be submitted in a sql file with the query number as the file name. set prefix as 3-2 to differentiate the query file for the day.

The query definition should be in this file.

CASE 1

Table: Activity

Column Name	Type
player_id	int
device_id	int
event_date	date
games_played	int

SQL Schema
(player_id, event_date) is the primary key of this table.
This table shows the activity of players of some games.
Each row is a record of a player who logged in and played a number of games
(possibly 0) before logging out on someday using some device.

Input:
Activity table:

player_id	device_id	event_date	games_played
1	2	2016-03-01	5
1	2	2016-05-02	6
2	3	2017-06-25	1
3	1	2016-03-02	0
3	4	2018-07-03	5

Question 1: Write an SQL query to report the first login date for each player. Return the result table in any order.

The query result format is in the following example.
Output:

player_id	first_login
1	2016-03-01
2	2017-06-25
3	2016-03-02

Question 2: Write an SQL query to report the device that is first logged in for each player. Return the result table in any order.

The query result format is in the following example.

Output:

player_id	device_id
1	2
2	3
3	1

Question 3: Write an SQL query to report for each player and date, how many games played so far by the player. That is, the total number of games played by the player until that date. Check the example for clarity. Return the result table in any order.

The query result format is in the following example.

Input:
Activity table:

player_id	device_id	event_date	games_played
1	2	2016-03-01	5
1	2	2016-05-02	6
1	3	2017-06-25	1
3	1	2016-03-02	0

| 3 | 4 | 2018-07-03 | 5 |
+-----------+-----------+------------+--------------+

Output:

player_id	event_date	games_played_so_far
1	2016-03-01	5
1	2016-05-02	11
1	2017-06-25	12
3	2016-03-02	0
3	2018-07-03	5

Explanation:
For the player with id 1, 5 + 6 = 11 games played by 2016-05-02, and 5 + 6 + 1 = 12 games played by 2017-06-25.
For the player with id 3, 0 + 5 = 5 games played by 2018-07-03.
Note that for each player we only care about the days when the player logged in.

CASE 2

Write an SQL query to report all the classes that have at least five students. Return the result table in any order.

Table: Courses

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| student     | varchar |
| class       | varchar |
+-------------+---------+
```

(student, class) is the primary key column for this table.
Each row of this table indicates the name of a student and the class in which they are enrolled.

The query result format is in the following example.

Example 1:

Input:
Courses table:

```
+---------+----------+
| student | class    |
+---------+----------+
| A       | Math     |
| B       | English  |
| C       | Math     |
| D       | Biology  |
| E       | Math     |
| F       | Computer |
| G       | Math     |
| H       | Math     |
| I       | Math     |
+---------+----------+
```

Output:

```
+---------+
| class   |
+---------+
| Math    |
+---------+
```

Explanation:
- Math has 6 students, so we include it.
- English has 1 student, so we do not include it.
- Biology has 1 student, so we do not include it.

- Computer has 1 student, so we do not include it.

CASE 3

Write an SQL query to report the name, population, and area of the big countries.

Table: World

Column Name	Type
name	varchar
continent	varchar
area	int
population	int
gdp	int

name is the primary key column for this table.
Each row of this table gives information about the name of a country, the continent to which it belongs, its area, the population, and its GDP value.

A country is big if:

- it has an area of at least three million (i.e., 3000000 km^2), or
- it has a population of at least twenty-five million (i.e., 25000000).

The query result format is in the following example.

Example 1:

Input:
World table:

name	continent	area	population	gdp
Afghanistan	Asia	652230	25500100	20343000000
Albania	Europe	28748	2831741	12960000000
Algeria	Africa	2381741	37100000	188681000000

```
| Andorra     | Europe    | 468     | 78115      | 3712000000
|
| Angola      | Africa    | 1246700 | 20609294   | 100990000000
|
+-------------+-----------+---------+------------+--------------
+
Output:
+-------------+------------+---------+
| name        | population | area    |
+-------------+------------+---------+
| Afghanistan | 25500100   | 652230  |
| Algeria     | 37100000   | 2381741 |
+-------------+------------+---------+
```