BUAN 6320

Database Foundations for Business Analytics

Assignment 1

Problem 1

Create the following table in your database with the following schema:

Table: `Customer`

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| id          | int     |
| name        | varchar |
| referee_id  | int     |
+-------------+---------+
id is the primary key column for this table.
Each row of this table indicates the id of a customer, their name, and the id
of the customer who referred them.
```

Add the following data to your tables:

Input:

```
Customer table:
+----+------+------------+
| id | name | referee_id |
+----+------+------------+
| 1  | Will | null       |
| 2  | Jane | null       |
| 3  | Alex | 2          |
| 4  | Bill | null       |
| 5  | Zack | 1          |
| 6  | Mark | 2          |
+----+------+------------+
```

Write an SQL query to report the IDs of the customer that are not referred by the customer with id = 2.

Return the result table in any order.

The results should be:

Output:

```
+------+
| name |
+------+
| Will |
| Jane |
| Bill |
| Zack |
+------+
```

Problem 2

Assume you have a table in your database called ‘World’ with the following schema:

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.
```

It contains the following records of data:

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 |
+-------------+-----------+---------+------------+--------------+
```

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

Return the result table in any order.

A country is big if:

- it has an area of at least three million (i.e., 3000000 km2), or
- it has a population of at least twenty-five million (i.e., 25000000).

The results should be:

Output:

```
+-------------+------------+---------+
| name        | population | area    |
+-------------+------------+---------+
| Afghanistan | 25500100   | 652230  |
| Algeria     | 37100000   | 2381741 |
+-------------+------------+---------+
```

Problem 3

Assume you have a table in database called ‘Cinema’ with the following schema:

Table: `Cinema`

```
+----------------+----------+
| Column Name    | Type     |
+----------------+----------+
| id             | int      |
| movie          | varchar  |
| description    | varchar  |
| rating         | float    |
+----------------+----------+
id is the primary key for this table.
Each row contains information about the name of a movie, its genre, and its
rating.
rating is a 2 decimal places float in the range [0, 10]
```

The table has the following records of data:

Input:

Cinema table:

id	movie	description	rating
1	War	great 3D	8.9
2	Science	fiction	8.5
3	irish	boring	6.2
4	Ice song	Fantacy	8.6
5	House card	Interesting	9.1

Write an SQL query to report the movies with an odd-numbered ID and a description that is not "boring".

Return the result table ordered by rating in **descending** order.

The results should be:

Output:

id	movie	description	rating
5	House card	Interesting	9.1
1	War	great 3D	8.9