CST8260 - Lab Subqueries

Objective

In this lab you will look at UNION, JOIN, and subqueries.

Database

world, one of the databases you created in lab 1.

Questions

1. Reverse engineer the world database so that you can use it to help with your queries. **Take a screenshot**.
2. Using a UNION query, create a report that lists the name and the district of the cities in Canada (code CAN) and the Netherlands (code NLD). Note that this query can be written without using a UNION. The purpose of this question is to use UNION. (city table)
3. Using an INNER JOIN..ON, create a report that list the name of cities, the country and the indepyear of all countries whose indepYear field is not null. (city and country tables).
4. Rewrite the previous query using WHERE/FROM.
5. Using a subquery, list the name of the city, the countrycode, the population and the average city population of all cities with a population less than the average population. (city table)
6. Using INNER JOIN ..ON, write a query that displays, for the city of Toronto, the name of the city and the head of state of its country. (city and country tables)
7. Using INNER JOIN ..ON the three tables, write a query that displays the name of the city, the city continent, the city head of state, the year of independance and the percentage of people who speaks English in the city of Toronto. (city, countrylanguage and country tables)
8. Rewrite the previous query using WHERE/FROM.
9. Using a subquery, list the name of the city, the countrycode and the population of the city with the largest population. (city table)
10. Using a subquery, list the name of the city, the countrycode and the population of all the cities whose population is larger than the average population. (city table)
11. What is wrong with the following subquery?
    ```
    SELECT name,countrycode,population,avg(population)
    FROM city
    WHERE population > (SELECT avg(population) FROM city)
    ```
12. Replace the first avg(population) with (SELECT AVG(population) from city)
    What happens? Why?
13. Using a subquery, list the name of the country, the continent and the life expectancy of all countries whose life expectancy is less than the average life expectancy. (country table)

Deliverables

Upload your queries in a file called: lastName_StudentNumber.sql and a screenshot of the world database reversed engineered.