Data Mining & Knowledge Extraction
Assignment 2

Deadline

Strict deadline **December 17th 2023 at 23:59**.
The deadline is definitive and it already accounts for vacation days.
No submission after the deadline will be considered.

Instructions

- The assignment must be done individually.
- Implement each query in its own **plain text file**. The file name must be **`query_<number>.sql`** . For example: `query_8.sql` for query 8. The file should not contain anything else apart from the query and in case, the WITH statements you use.
- Each query will be either correct (+1 point) or wrong (+0 points).
- The **attributes** in the results must be **ordered as they appear in the text**. Example: "return the beer name and its manufacturer" should be matched with a "`SELECT name, manf FROM ...`" Failure to properly order the columns leads to (0 points).
- When asked to return information about a relation, select the attributes that identifies it, *i.e.* the primary key. Example: Select all the Beers that, "`SELECT name FROM Beers`".
- Unless otherwise specified, final results of queries must be **distinct**, meaning always use the distinct unless explicitly asked not to do it.
- The queries will be tested against a PostgreSQL database. You can install one on your computer if you want to work locally and do some tests, but always keep in mind that the queries have to work on PostgreSQL.
- Test your queries before submitting! You need to make sure that your queries return the correct results, no matter the content of the database. You need to think if there are any special cases. If you would like to get some real data values to test your queries, you can use the CIA World Factbook: https://www.cia.gov/the-world-factbook/

Delivery

- Go to the following Google Form: https://forms.gle/VvZFcvVbgr3eygq17
- Upload each query file via the corresponding item of the form
- Submit the form

Consider the following relational database schema (a more formal SQL DDL script to create the schema is provided at the end of this document). All attributes denoting percentages are expressed as integers in the interval [0-100].

countries(code: Str, name: Str, capital: Str, area: int)
- code (this is the usual country code, e.g. CDN for Canada, F for France, I for Italy)
- name (the country name)
- capital (the capital city, e.g. Rome for Italy)
- area (The mass land the country occupies in square Km)

populations(country: Str, population: int, children: int, adult: int, birth_rate: int, death_rate: int, sex_ratio: int)
- country is FK to the countries table
- population (this is the number of the people living in the country)
- children (The percentage of the population that are between 0 and 14 years old)
- adult (The percentage of the population that are between 15 and 64 years old)
- birth_rate (births in a year per 1,000 people of the population)
- death_rate (deaths per 1,000 individuals per year)
- sex_ratio (sex ratio at birth: male/female * 100)

economies (country: Str, gdp: int, inflation: int, military: int, poverty: int)
- country is FK to the countries table
- gdp (gross domestic product)
- inflation (annual inflation rate)
- military (military spending as percentage of the GDP)
- poverty (percentage of population below the poverty line)

languages (country: Str, language: Str, percentage: int)
- country is FK to the countries table
- language is a spoken language name
- percentage (percentage of population speaking the language)

All the attributes are considered to be NOT NULL

Create an SQL query that implements each of the following:

1. Find the dominant language (dominant means: spoken by more than 50% of the population) of the countries with the highest male/female ratio.
2. Find the poverty rate in the countries with the largest number of languages spoken.
3. Find all countries where English is the dominant language, and the poverty rate is higher than that of all countries with the name 'USA'.
4. Find the countries with the fastest declining population (a decline is a positive value in the difference death_rate - birth_rate).
5. For each language, find the percentage of the world population that speaks it.
6. For each language, find the percentage of the world population that speaks it, but considering only countries whose population is declining.
7. Assume that all the countries stop military spending, and distribute the money back to their citizens. Find the average, maximum, and minimum increase of GDP per capita due to this action. For the minimum and maximum, also list the country (countries). Your answer should have 3 tuples with 2 columns each. The first should be the maximum increase with the name of the country with the maximum increase. The second should be the minimum increase with the name of the country that has that minimum increase. The third should be the average increase with a NULL value. In case there are more than one countries in each of these categories, with the same value (e.g. more than one country has an increase equal to the maximum value), report all the countries in alphabetical order. In such a case, obviously the answer will contain more than three tuples.
8. Order languages by the average percentage of the adult population of countries in which they are spoken by at least 25% of the population (in the decreasing order). If two languages have the same average percentage, then the languages are ordered in alphabetical order.
9. Find the richest (those that have the highest GDP) countries among the ones whose name starts with a 'C'.
10. Find the code of the 15th largest country in the world (in terms of land mass area). If more than one country has the same area, then the one with a code alphabetically preceding the other comes first in the list. For example, if Canada is the largest country and has a land mass of 1000 square Km, and India has also 1000 sq meters, and USA has 900, then India is the second largest (because it comes after Canada alphabetically), and USA is considered the 3rd largest country).

Script to create the database schema:

```
CREATE TABLE countries
(code varchar(25) not null,
name varchar(25) not null,
capital varchar(25) not null,
area integer not null,
PRIMARY KEY (code));

CREATE TABLE populations
(country varchar(25) not null,
population integer not null,
children integer not null,
adult integer not null,
birth_rate integer not null,
death_rate integer not null,
sex_ratio integer not null,
PRIMARY KEY (country),
FOREIGN KEY (country)
  REFERENCES countries(code));

CREATE TABLE economies
(country varchar(25) not null,
gdp integer not null,
inflation integer not null,
military integer not null,
poverty integer not null,
PRIMARY KEY (country),
FOREIGN KEY (country)
  REFERENCES countries(code));

CREATE TABLE languages
(country varchar(25) not null,
language varchar(23) not null,
percentage integer not null,
PRIMARY KEY (country, language),
  FOREIGN KEY (country)
  REFERENCES countries(code));
```