CPSC 304 Project Cover Page

Milestone #: ___4_____

Date: _____April 2, 2025_______

Group Number: ____44______

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Danesh Rahmani	99189482	e2m2w	drahmani97@gmail.com
Vansh Chanana	59845008	w0f1n	vanshchanana2004@gmail.com
Hardit Singh	31250616	e7r5s	hardits925@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Description:

This project's objective was to make a global information system database where users can find information about and understand relationships between countries and their various attributes through a web application in one place. They can also gain insights into different demographics and characteristics of countries.

What did the project accomplish?

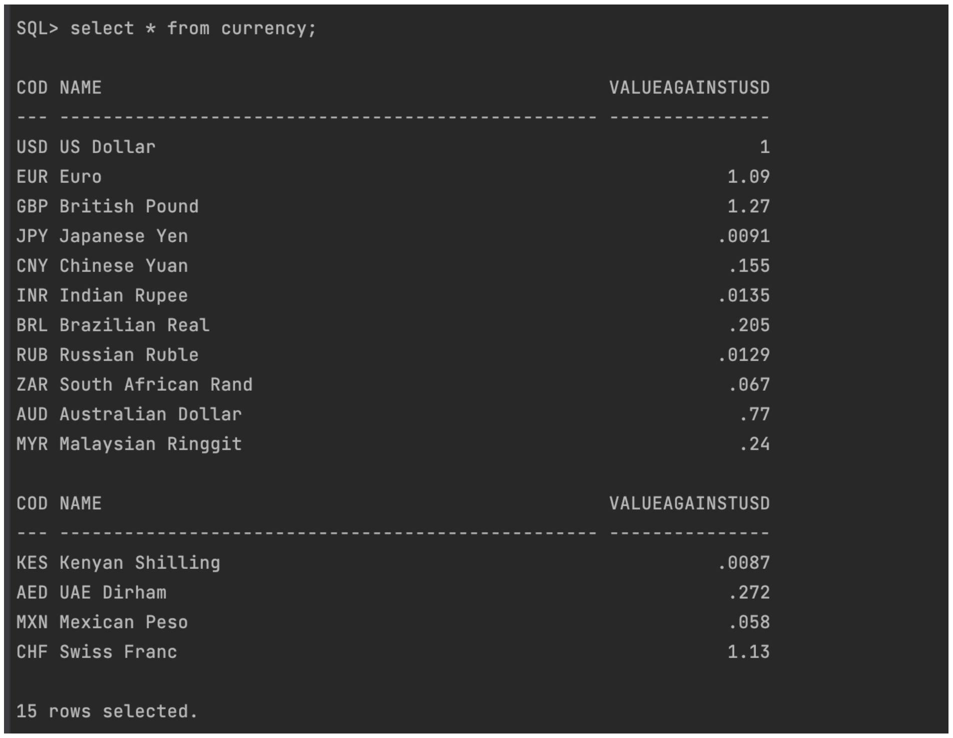
The project gives users an interactive way to know more about domains of countries like GDP, average city population, languages / language families and more. This application also lets users add , edit and remove information through the backend logic written in SQL in addition to a simple and user-friendly interface.

Schema:

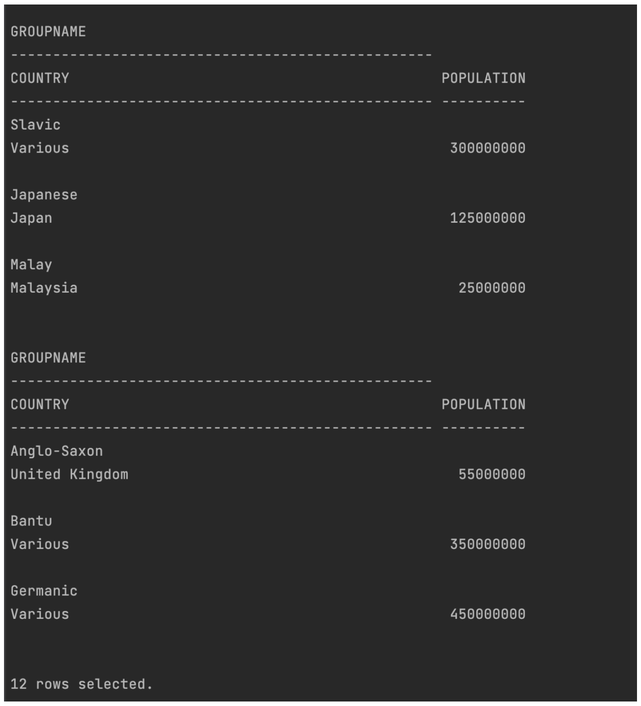
EntitySet or Relationship	**Corresponding Schema**	**Candidate Keys (Non PK are listed)**	**Other Constraints**
Continent	Continent(ContinentName: char[13], NumCountries: int, SurfaceArea: int)	{NumCountries, SurfaceArea}	CHECK NumCountries and SurfaceArea are positive values
Religion	Religion(ReligionName: char[20], Followers: int)	N/A	Followers' DEFAULT value is 0 if it is unknown. CHECK that Followers is a positive value.
Language	Language(LanguageName: char[20], ScriptType: char[20], LanguageFamily: char[20])	N/A	
Ethnic Group	EthnicGroup(GroupName: char[20], Country: char[31], Population: int)	N/A	Population DEFAULT value is 0 if it is unknown
Currency	Currency(Code: char[3], Name: char[20], valueAgainstUSD: int)	Name	Name is Unique
Timezone	Timezone(Code: char[3], Name: char[20], utcOffset: char[6])	{Name, utcOffset}	utcOffset is NOT NULL

Country Climate	CountryClimate(**CountryID:** int, avgTemp: int, avgPrecipitation: int)	N/A	CHECK that avgPrecipitation is not negative
Climate Type	ClimateType(ClimateTypeID: int, ClimateTypeName: char[20])	ClimateTypeName	ClimateTypeName is NOT NULL
City	City(CityName: char[168], **CountryID:** int, population: int)	N/A	CountryID is NOT NULL. CHECK that population is not negative
City ISA	Capital(**CountryID:** int, **CityName**: char[168], since: int) PortCity(**CountryID:** int, **CityName: char[168]**)	N/A	N/A
Country	Country(ID: int, Name: char[20], Population: int, GDP: int, SurfaceArea: int, **CurrencyCode:** char[3]) (CurrencyCode cannot be null)	Name	Name is UNIQUE, CurrencyCode is NOT NULL. *Population >= 0, GDP >= 0, SurfaceArea >= 0*
IsOfficial Language	IsOfficialLanguage(**CountryID:** int, **LanguageName**: char[20])	N/A	N/A
Has Timezone	HasTimezone(**CountryID**: int, **TimezoneCode:** char[5])	N/A	N/A
HasBorder	HasBorder(**Country1ID:** int, **Country2ID**: int, Length: int)	N/A	CHECK Country1ID !== Country2ID
HasEthnic Group	HasEthnicGroup(**CountryID**: int, **EthnicGroupName:** char[20])	N/A	N/A
Has Climate Type	HasClimateType(**ClimateTypeID:** int, **CountryID**: int)	N/A	N/A
Speaks Language	SpeaksLanguage(**LanguageName**: char[20], **CountryID:** int, **EthnicGroupName**: char[20], percentPopulation: int)	N/A	Percent Population DEFAULT value is 0 if unknown
LocatedIn	LocatedIn(**CountryID:** int, **ContinentName:** char[13])	N/A	N/A
Practices Religion	PracticesReligion(**CountryID:** int, **ReligionName:** char[20])	N/A	N/A

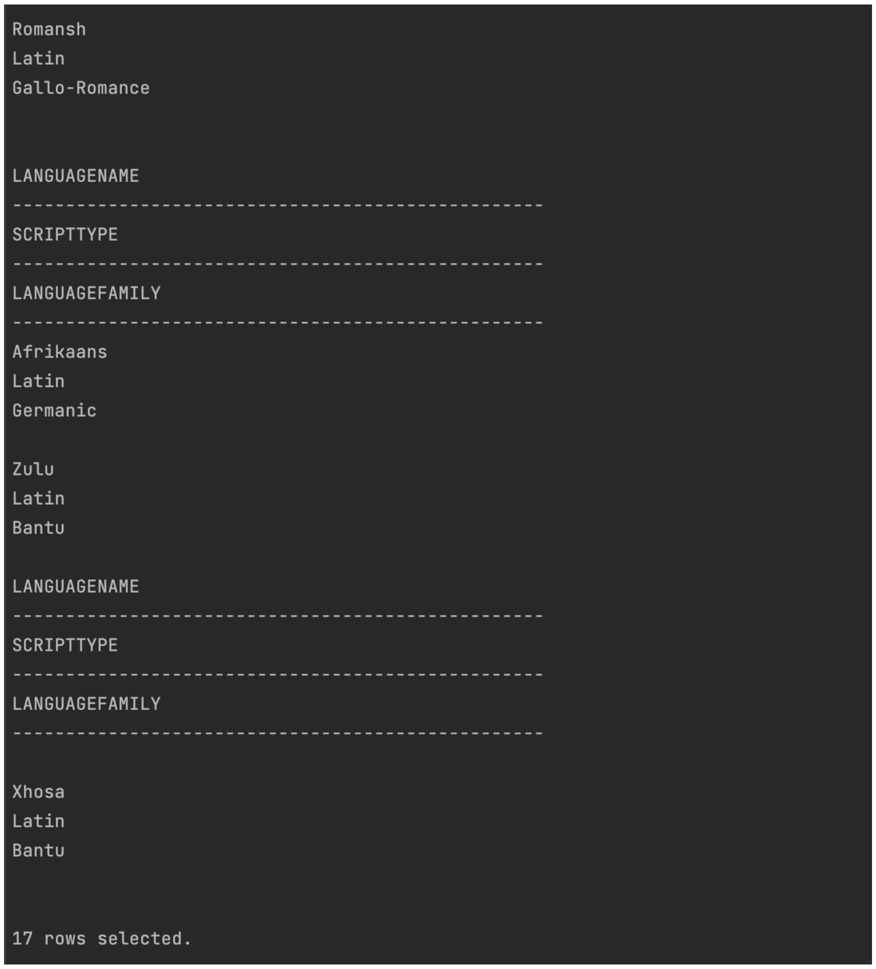
Screenshots from all tables created by running the initializationn.sql file:



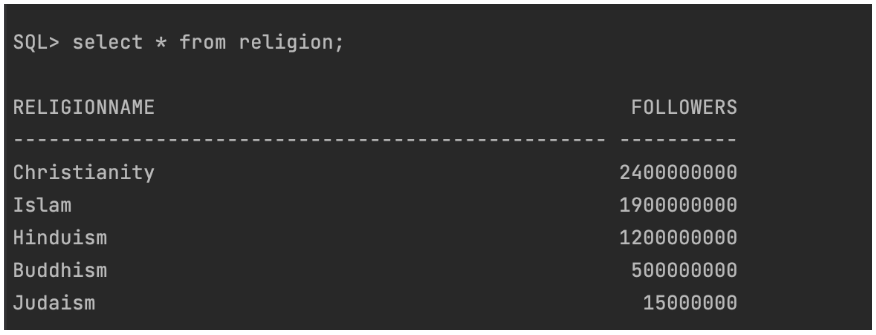
```
SQL> select * from currency;

COD NAME                                               VALUEAGAINSTUSD
--- -------------------------------------------------- ---------------
USD US Dollar                                                        1
EUR Euro                                                          1.09
GBP British Pound                                                 1.27
JPY Japanese Yen                                                 .0091
CNY Chinese Yuan                                                  .155
INR Indian Rupee                                                 .0135
BRL Brazilian Real                                                .205
RUB Russian Ruble                                                .0129
ZAR South African Rand                                            .067
AUD Australian Dollar                                              .77
MYR Malaysian Ringgit                                              .24

COD NAME                                               VALUEAGAINSTUSD
--- -------------------------------------------------- ---------------
KES Kenyan Shilling                                              .0087
AED UAE Dirham                                                    .272
MXN Mexican Peso                                                  .058
CHF Swiss Franc                                                   1.13

15 rows selected.
```



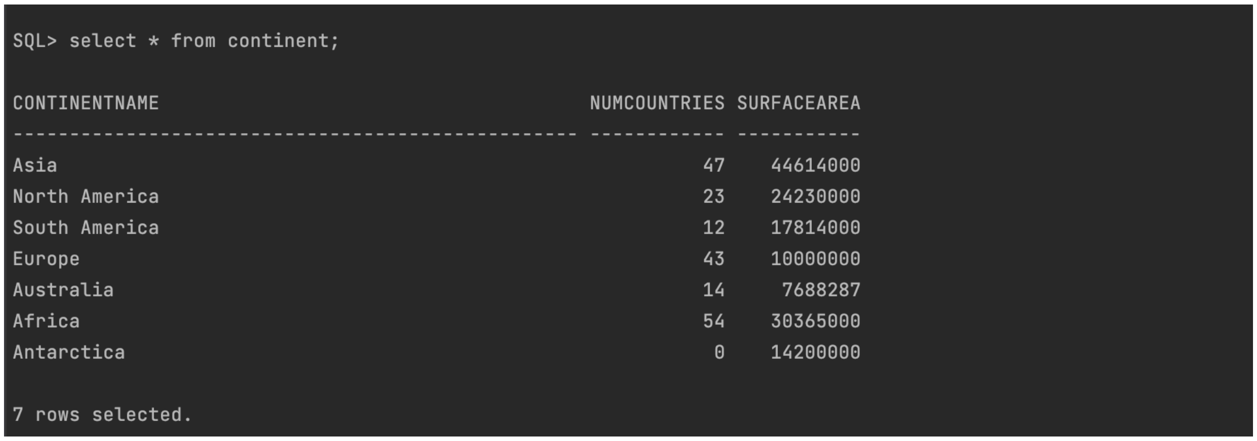
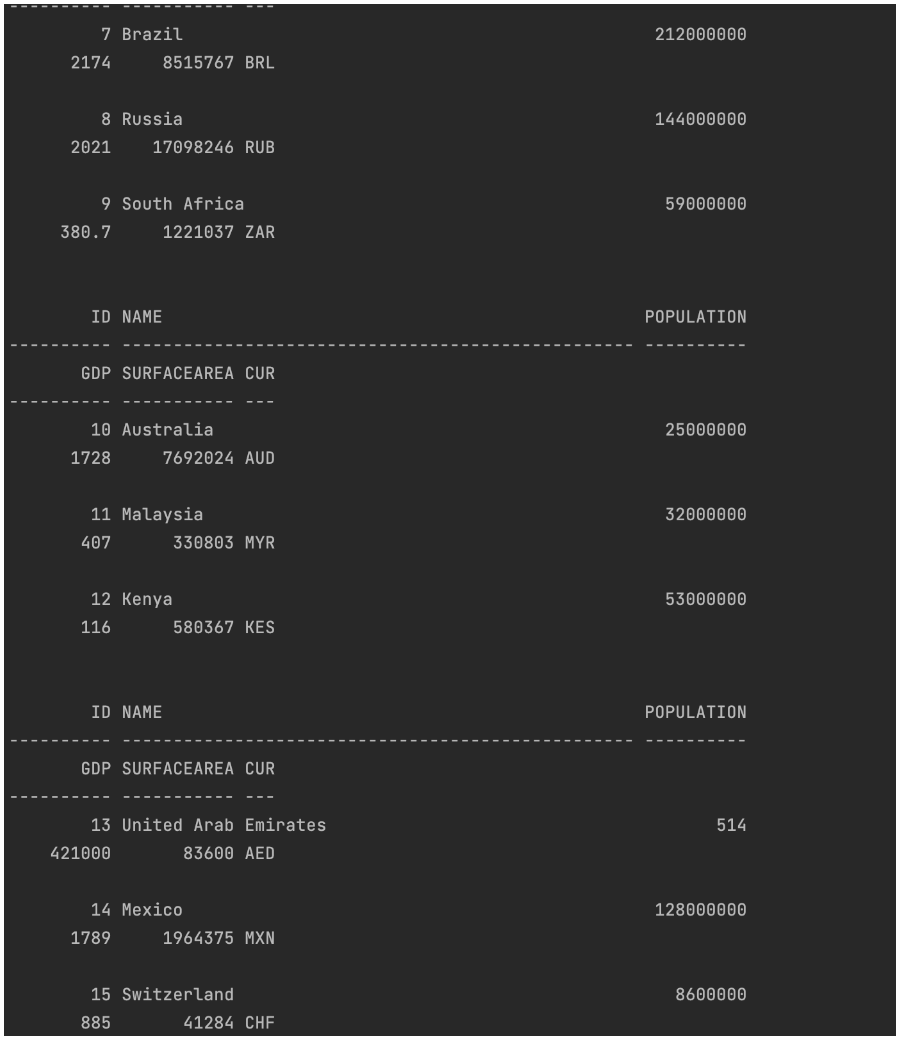
```
GROUPNAME
--------------------------------------------------
COUNTRY                                            POPULATION
-------------------------------------------------- ----------
Slavic
Various                                             300000000

Japanese
Japan                                               125000000

Malay
Malaysia                                             25000000


GROUPNAME
--------------------------------------------------
COUNTRY                                            POPULATION
-------------------------------------------------- ----------
Anglo-Saxon
United Kingdom                                       55000000

Bantu
Various                                             350000000

Germanic
Various                                             450000000


12 rows selected.
```

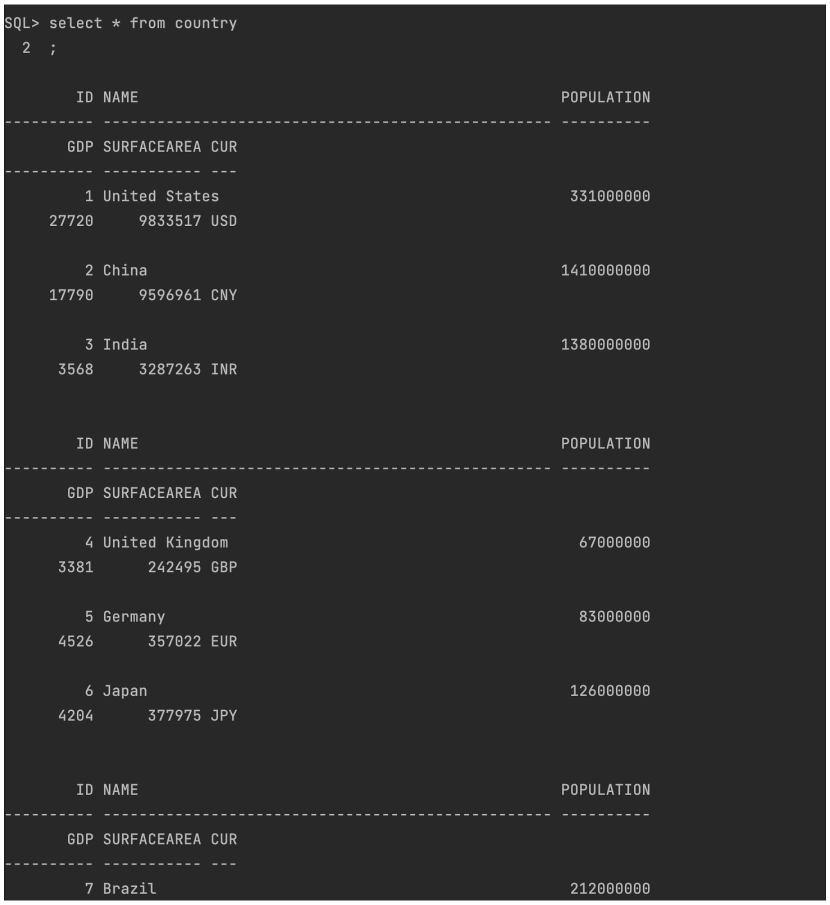
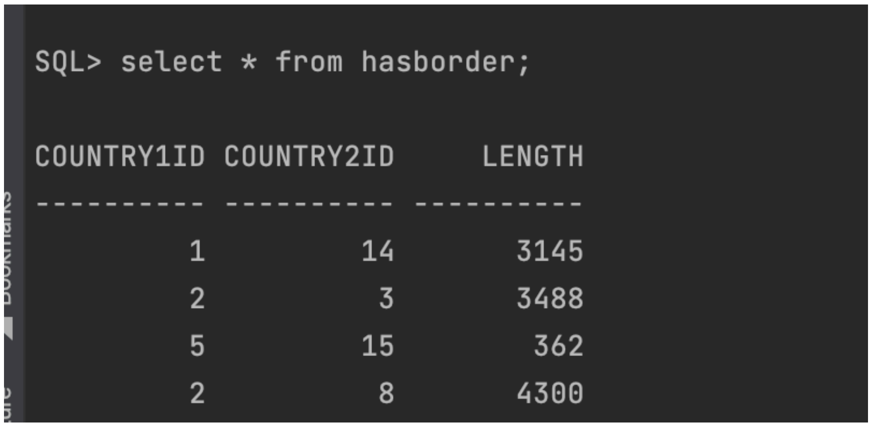


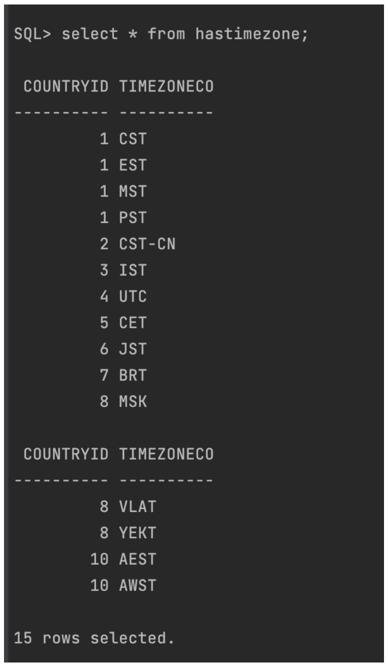
```
Romansh
Latin
Gallo-Romance


LANGUAGENAME
--------------------------------------------------
SCRIPTTYPE
--------------------------------------------------
LANGUAGEFAMILY
--------------------------------------------------
Afrikaans
Latin
Germanic

Zulu
Latin
Bantu

LANGUAGENAME
--------------------------------------------------
SCRIPTTYPE
--------------------------------------------------
LANGUAGEFAMILY
--------------------------------------------------

Xhosa
Latin
Bantu


17 rows selected.
```



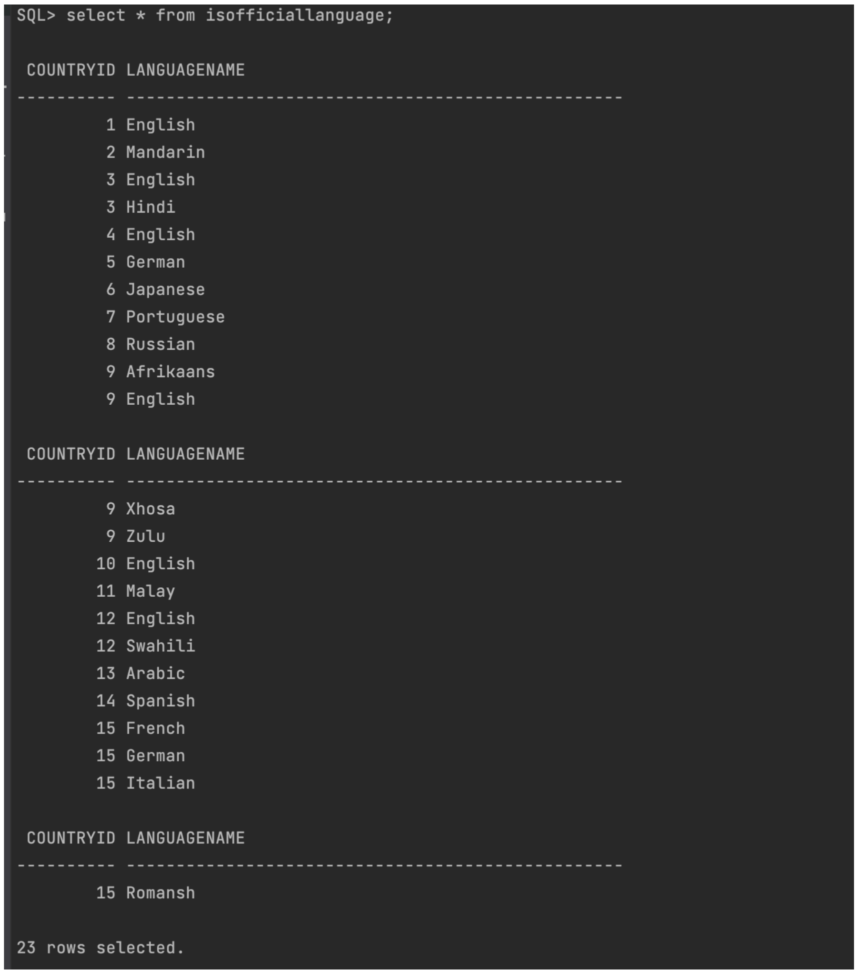
```
SQL> select * from religion;

RELIGIONNAME                                        FOLLOWERS
-------------------------------------------------- ----------
Christianity                                       2400000000
Islam                                              1900000000
Hinduism                                           1200000000
Buddhism                                            500000000
Judaism                                              15000000
```


```
SQL> select * from continent;

CONTINENTNAME                                       NUMCOUNTRIES SURFACEAREA
--------------------------------------------------- ------------ -----------
Asia                                                          47    44614000
North America                                                 23    24230000
South America                                                 12    17814000
Europe                                                        43    10000000
Australia                                                     14     7688287
Africa                                                        54    30365000
Antarctica                                                     0    14200000

7 rows selected.
```

```
---------- ----------- ---
         7 Brazil                                              212000000
      2174     8515767 BRL

         8 Russia                                              144000000
      2021    17098246 RUB

         9 South Africa                                         59000000
     380.7     1221037 ZAR


        ID NAME                                               POPULATION
---------- -------------------------------------------------- ----------
       GDP SURFACEAREA CUR
---------- ----------- ---
        10 Australia                                            25000000
      1728     7692024 AUD

        11 Malaysia                                             32000000
       407      330803 MYR

        12 Kenya                                                53000000
       116      580367 KES


        ID NAME                                               POPULATION
---------- -------------------------------------------------- ----------
       GDP SURFACEAREA CUR
---------- ----------- ---
        13 United Arab Emirates                                      514
    421000       83600 AED

        14 Mexico                                              128000000
      1789     1964375 MXN

        15 Switzerland                                           8600000
       885       41284 CHF
```

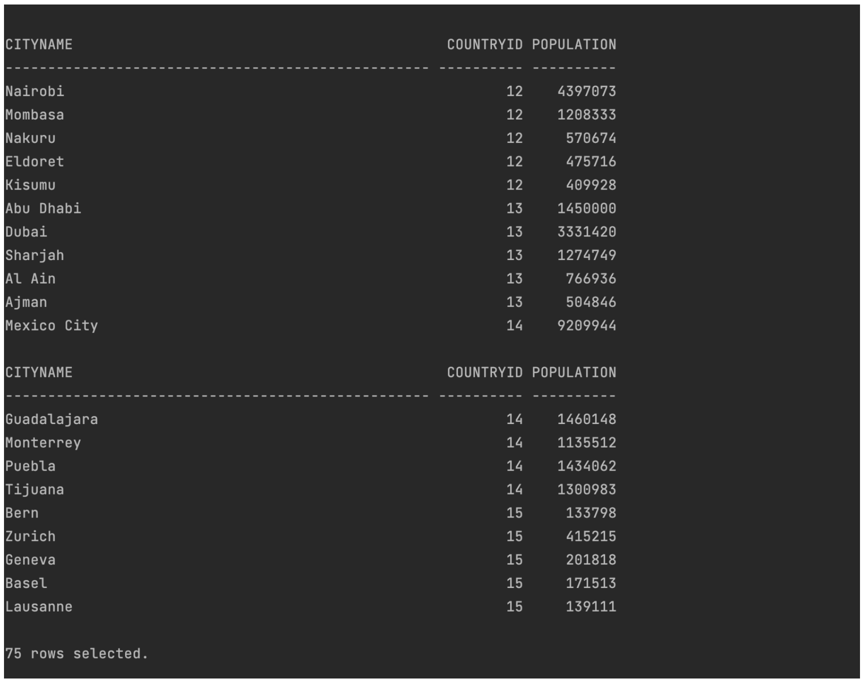

```
SQL> select * from country
  2  ;

        ID NAME                                               POPULATION
---------- -------------------------------------------------- ----------
       GDP SURFACEAREA CUR
---------- ----------- ---
         1 United States                                       331000000
     27720     9833517 USD

         2 China                                              1410000000
     17790     9596961 CNY

         3 India                                              1380000000
      3568     3287263 INR


        ID NAME                                               POPULATION
---------- -------------------------------------------------- ----------
       GDP SURFACEAREA CUR
---------- ----------- ---
         4 United Kingdom                                       67000000
      3381      242495 GBP

         5 Germany                                              83000000
      4526      357022 EUR

         6 Japan                                               126000000
      4204      377975 JPY


        ID NAME                                               POPULATION
---------- -------------------------------------------------- ----------
       GDP SURFACEAREA CUR
---------- ----------- ---
         7 Brazil                                              212000000
```

```
SQL> select * from hasborder;

COUNTRY1ID COUNTRY2ID     LENGTH
---------- ---------- ----------
         1         14       3145
         2          3       3488
         5         15        362
         2          8       4300
```



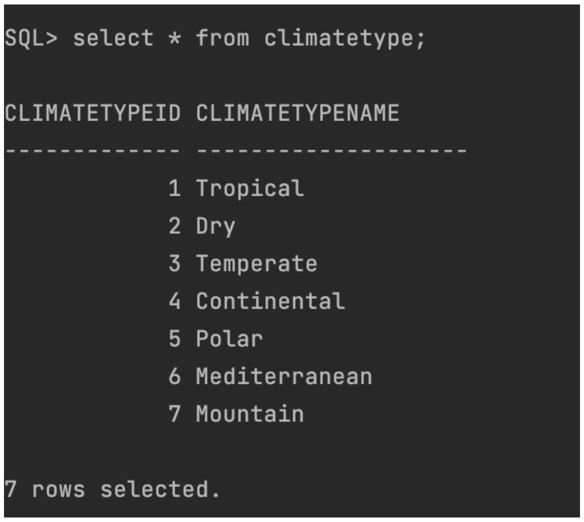
```
SQL> select * from hastimezone;

 COUNTRYID TIMEZONECO
---------- ----------
         1 CST
         1 EST
         1 MST
         1 PST
         2 CST-CN
         3 IST
         4 UTC
         5 CET
         6 JST
         7 BRT
         8 MSK

 COUNTRYID TIMEZONECO
---------- ----------
         8 VLAT
         8 YEKT
        10 AEST
        10 AWST

15 rows selected.
```



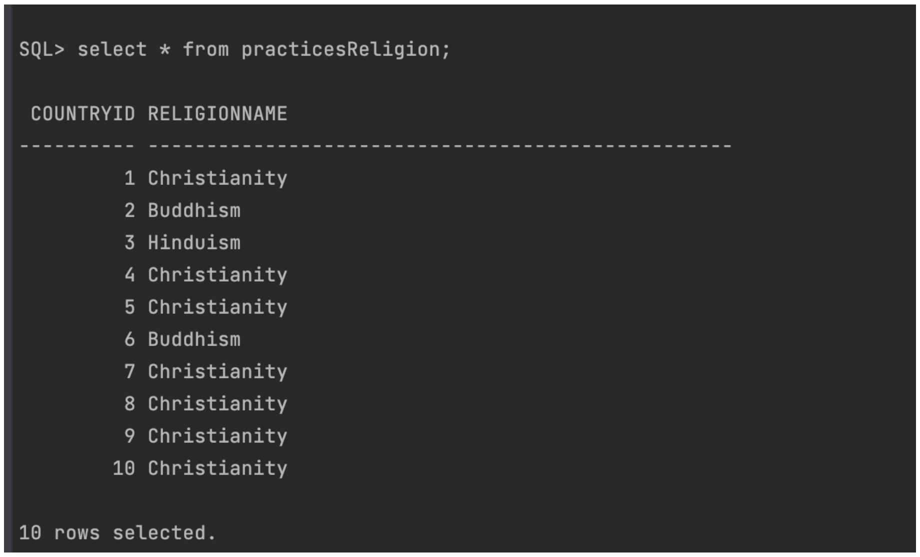
```
SQL> select * from isofficiallanguage;

 COUNTRYID LANGUAGENAME
---------- --------------------------------------------------
         1 English
         2 Mandarin
         3 English
         3 Hindi
         4 English
         5 German
         6 Japanese
         7 Portuguese
         8 Russian
         9 Afrikaans
         9 English

 COUNTRYID LANGUAGENAME
---------- --------------------------------------------------
         9 Xhosa
         9 Zulu
        10 English
        11 Malay
        12 English
        12 Swahili
        13 Arabic
        14 Spanish
        15 French
        15 German
        15 Italian

 COUNTRYID LANGUAGENAME
---------- --------------------------------------------------
        15 Romansh

23 rows selected.
```



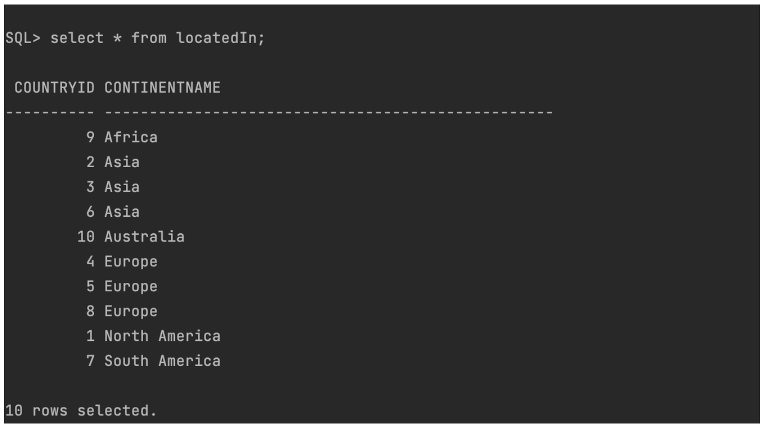
```
CITYNAME                                         COUNTRYID POPULATION
------------------------------------------------ ---------- ----------
Nairobi                                                 12    4397073
Mombasa                                                 12    1208333
Nakuru                                                  12     570674
Eldoret                                                 12     475716
Kisumu                                                  12     409928
Abu Dhabi                                               13    1450000
Dubai                                                   13    3331420
Sharjah                                                 13    1274749
Al Ain                                                  13     766936
Ajman                                                   13     504846
Mexico City                                             14    9209944

CITYNAME                                         COUNTRYID POPULATION
------------------------------------------------ ---------- ----------
Guadalajara                                             14    1460148
Monterrey                                               14    1135512
Puebla                                                  14    1434062
Tijuana                                                 14    1300983
Bern                                                    15     133798
Zurich                                                  15     415215
Geneva                                                  15     201818
Basel                                                   15     171513
Lausanne                                                15     139111

75 rows selected.
```



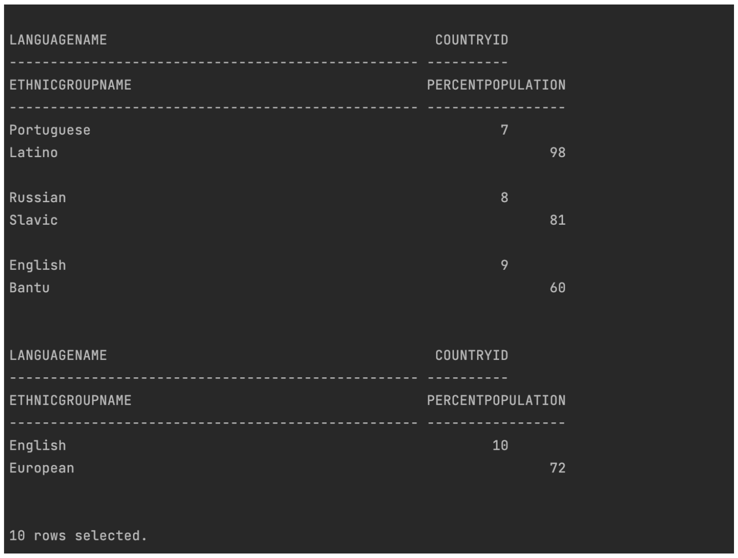
```
SQL> select * from climatetype;

CLIMATETYPEID CLIMATETYPENAME
------------- --------------------
            1 Tropical
            2 Dry
            3 Temperate
            4 Continental
            5 Polar
            6 Mediterranean
            7 Mountain

7 rows selected.
```



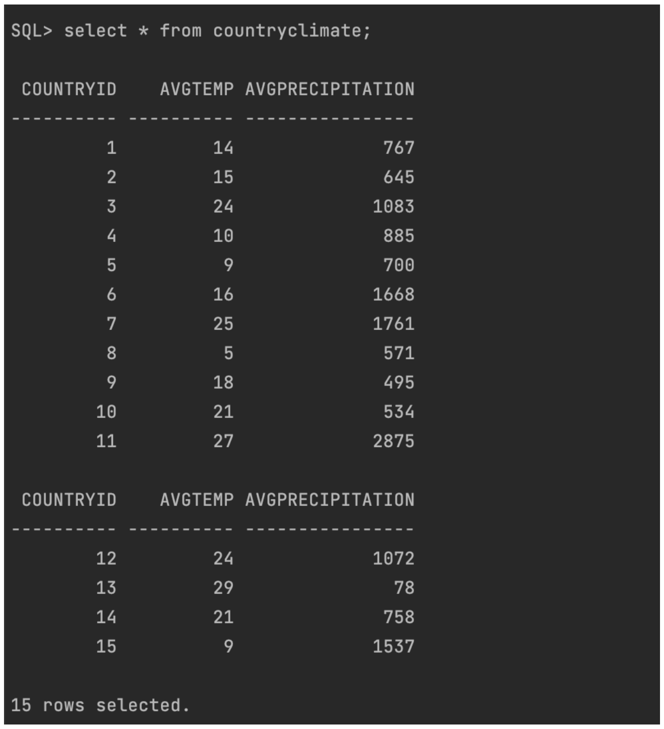
```
SQL> select * from practicesReligion;

 COUNTRYID RELIGIONNAME
---------- --------------------------------------------------
         1 Christianity
         2 Buddhism
         3 Hinduism
         4 Christianity
         5 Christianity
         6 Buddhism
         7 Christianity
         8 Christianity
         9 Christianity
        10 Christianity

10 rows selected.
```



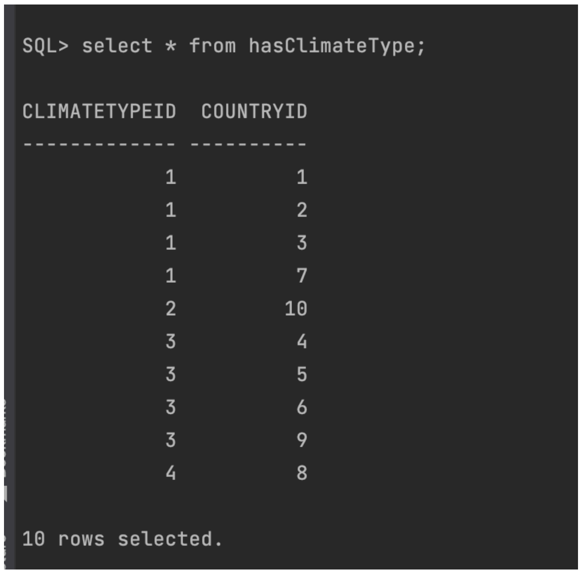
```
SQL> select * from locatedIn;

 COUNTRYID CONTINENTNAME
---------- --------------------------------------------------
         9 Africa
         2 Asia
         3 Asia
         6 Asia
        10 Australia
         4 Europe
         5 Europe
         8 Europe
         1 North America
         7 South America

10 rows selected.
```



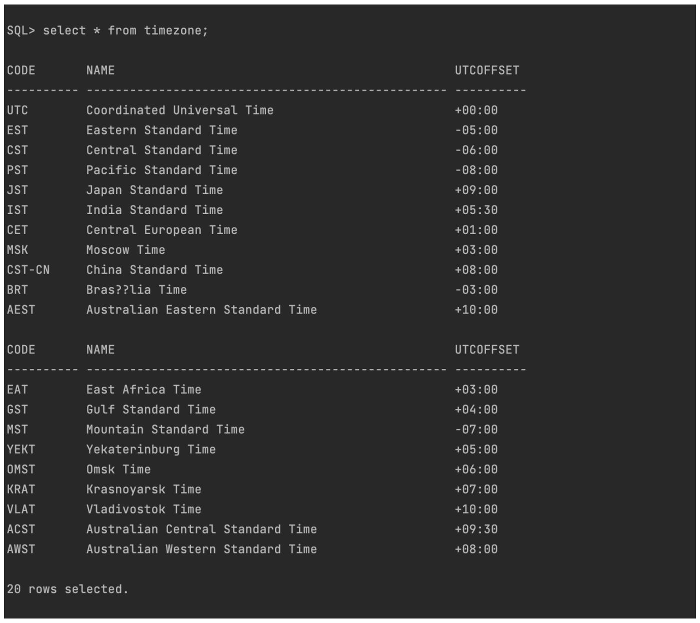
```
LANGUAGENAME                                        COUNTRYID
-------------------------------------------------- ----------
ETHNICGROUPNAME                                    PERCENTPOPULATION
-------------------------------------------------- -----------------
Portuguese                                                  7
Latino                                                            98

Russian                                                     8
Slavic                                                            81

English                                                     9
Bantu                                                             60


LANGUAGENAME                                        COUNTRYID
-------------------------------------------------- ----------
ETHNICGROUPNAME                                    PERCENTPOPULATION
-------------------------------------------------- -----------------
English                                                    10
European                                                          72


10 rows selected.
```



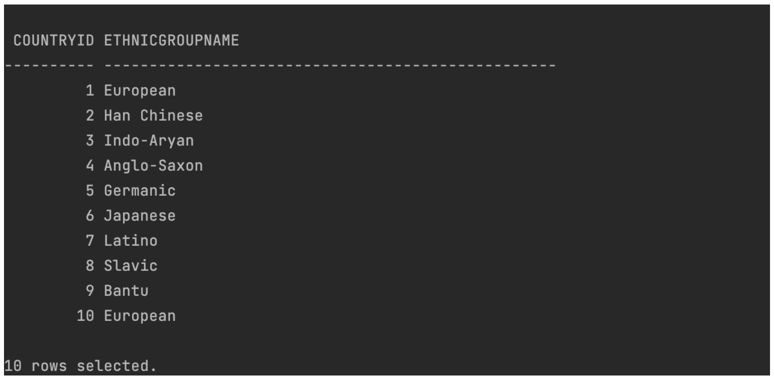
```
SQL> select * from countryclimate;

 COUNTRYID    AVGTEMP AVGPRECIPITATION
---------- ---------- ----------------
         1         14              767
         2         15              645
         3         24             1083
         4         10              885
         5          9              700
         6         16             1668
         7         25             1761
         8          5              571
         9         18              495
        10         21              534
        11         27             2875

 COUNTRYID    AVGTEMP AVGPRECIPITATION
---------- ---------- ----------------
        12         24             1072
        13         29               78
        14         21              758
        15          9             1537

15 rows selected.
```



```
SQL> select * from hasClimateType;

CLIMATETYPEID  COUNTRYID
------------- ----------
            1          1
            1          2
            1          3
            1          7
            2         10
            3          4
            3          5
            3          6
            3          9
            4          8

10 rows selected.
```



```
SQL> select * from timezone;

CODE       NAME                                               UTCOFFSET
---------- -------------------------------------------------- ----------
UTC        Coordinated Universal Time                         +00:00
EST        Eastern Standard Time                              -05:00
CST        Central Standard Time                              -06:00
PST        Pacific Standard Time                              -08:00
JST        Japan Standard Time                                +09:00
IST        India Standard Time                                +05:30
CET        Central European Time                              +01:00
MSK        Moscow Time                                        +03:00
CST-CN     China Standard Time                                +08:00
BRT        Bras??lia Time                                     -03:00
AEST       Australian Eastern Standard Time                   +10:00

CODE       NAME                                               UTCOFFSET
---------- -------------------------------------------------- ----------
EAT        East Africa Time                                   +03:00
GST        Gulf Standard Time                                 +04:00
MST        Mountain Standard Time                             -07:00
YEKT       Yekaterinburg Time                                 +05:00
OMST       Omsk Time                                          +06:00
KRAT       Krasnoyarsk Time                                   +07:00
VLAT       Vladivostok Time                                   +10:00
ACST       Australian Central Standard Time                   +09:30
AWST       Australian Western Standard Time                   +08:00

20 rows selected.
```



```
 COUNTRYID ETHNICGROUPNAME
---------- --------------------------------------------------
         1 European
         2 Han Chinese
         3 Indo-Aryan
         4 Anglo-Saxon
         5 Germanic
         6 Japanese
         7 Latino
         8 Slavic
         9 Bantu
        10 European

10 rows selected.
```

Changes in final schema:

Country : a) Changed the GDP data type from INT to FLOAT to accommodate decimal values.
b) Changed ON DELETE SET DEFAULT to ON DELETE SET NULL so there is no default value set when the record is deleted.

Continent : Changed CHECK (numCountries > 0) to numCountries >= 0 for continents like Antarctica which does not have a country.

Currency : Changed DECIMAL(10,8) to FLOAT to maintain consistency.

SQL Script

The list of all queries (Drop table statements, create table statements, insert statements) can be found in the initialization.sql file under the **databases directory** in the project repository.

Queries

QUERY 1

INSERT:

- insertCity function in cityService.js under the services directory. (line 28)
- Route for the same in cityController.js under the controllers directory. (line 24)
- Integration with frontend : scripts.js (line 84)

SQL Query:

INSERT INTO City (CityName, CountryID, Population)
VALUES (:cityName, :countryId, :population)

QUERY 2

UPDATE:

- updateCurrency function in currencyService.js under the services directory. (Line 28)
- Route for the same in currencyController.js under the controllers directory. (Line 21)
- Integration with frontend : scripts.js (line 121)

SQL Query:

UPDATE Currency
SET Name = :currencyName, ValueAgainstUSD = :valueAgainstUSD
WHERE Code = :currencyCode

QUERY 3

DELETE:

- deleteCity function in cityService.js under the services directory. (Line 65)
- Route for the same in cityController.js under the controllers directory. (Line 46)
- Integration with frontend : scripts.js (line 160)

SQL Query:

DELETE FROM City
WHERE CityName = :cityName AND CountryID = :countryID

QUERY 4

SELECTION:

- countrySelection function in countryService.js in the services directory. (Line 29)
- Route for the same in countryController.js under the controllers directory. (Line 24)
- Integration with frontend : scripts.js (line 197)

SQL Query:

QUERY 5

PROJECTION:

- getLanguageFamily function in languageService.js in the services directory. (Line 51)
- Route for the same in languageController.js under the controllers directory. (Line 45)
- Integration with frontend : scripts.js (line 245)

SQL Query:

```
SELECT ${columnsStr}
FROM Language
ORDER BY LanguageName
```

QUERY 6

JOIN:

- getCountriesByLanguage function in languageService.js in the services directory. (Line 26)
- Route for the same in languageController.js under the controllers directory. (line 27)
- Integration with frontend : scripts.js (line 314)

SQL Query:

```
SELECT c.Name AS CountryName, s.EthnicGroupName, s.percentPopulation
FROM SpeaksLanguage s, Country c, Language l
WHERE s.CountryID = c.ID
  AND s.LanguageName = l.LanguageName
  AND s.LanguageName = :languageName
```

QUERY 7

Aggregation with GROUP BY:

- getCountriesByLanguage function in countryService.js in the services directory. (Line 101)
- Route for the same in languageController.js under the controllers directory. (line 46)
- Integration with frontend : scripts.js (line 348)

SQL Query:

```
SELECT Co.ID as CountryID,
       Co.Name as CountryName,
       AVG(Ci.Population) as AvgCityPopulation,
       COUNT(Ci.CityName) as CityCount
FROM Country Co
JOIN City Ci ON Co.ID = Ci.CountryID
GROUP BY Co.ID, Co.Name
ORDER BY AvgCityPopulation DESC
```

Explanation: This query finds the average population of cities across each country. The output is in descending order of average city population.

QUERY 8

Aggregation with HAVING:

- getWidelySpokenLanguages function in languageService.js in the services directory. (Line 96)
- Route for the same in languageController.js under the controllers directory. (line 67)
- Integration with frontend : scripts.js (line 380)

SQL Query:

```
SELECT LanguageName, COUNT(CountryID) AS CountryCount
      FROM IsOfficialLanguage
      GROUP BY LanguageName
      HAVING COUNT(CountryID) >= 2
```

Explanation: This query selects languages that are spoken in strictly 2 or more countries.

QUERY 9

Aggregation with nested GROUP BY:

- lowestAvgGDPAcrossContinents function in GDPService.js in the services directory. (Line 25)

- Route for the same in GDPController.js under the controllers directory. (line 21)
- Integration with frontend : scripts.js (line 441)

SQL Query:

```
SELECT l.ContinentName
FROM Country c, LocatedIn l
WHERE c.ID = l.CountryID
GROUP BY l.ContinentName
HAVING AVG(c.GDP) <= ALL (
   SELECT AVG(c2.GDP)
   FROM Country c2, LocatedIn l2
   WHERE c2.ID = l2.CountryID
   GROUP BY l2.ContinentName
)
```

Explanation: This query gives us the continent with the lowest avg gdp across all countries.

QUERY 10
DIVISION:

- countriesWithAllClimateTypes function in countryService.js in the services directory. (Line 141)
- Route for the same in countryController.js under the controllers directory. (line 56)
- Integration with frontend : scripts.js (line 411)

SQL Query:

```
SELECT C.ID, C.Name
FROM Country C
WHERE NOT EXISTS (
  SELECT CT.ClimateTypeID
  FROM ClimateType CT
  WHERE NOT EXISTS (
     SELECT 1
     FROM HasClimateType HCT
     WHERE HCT.CountryID = C.ID
     AND HCT.ClimateTypeID = CT.ClimateTypeID
  )
)
ORDER BY C.Name
```

Explanation: This query gets countries which have all climate types specified in the climate relation.