Tasks

PART 1. Essay question: explaining the steps you would take to set up and create this database.

A database in SQL Server is made up of a collection of tables that stores a specific set of structured data. A table contains a collection of rows, also referred to as records or tuples, and columns, also referred to as attributes. Each column in the table is designed to store a certain type of information, for example, dates, names, dollar amounts, and numbers. One example of Relational databases is MySQL.

MySQL has many uses and is used in business for data and business analytics, development, and data science. It is a great tool to communicate with data and is the standard language for relational database management systems that uses statements to perform tasks such as create a table or update data on a database, or retrieve data from a date. Gathering all information needed to record in the database is critrical. It requires dividing information items into major entities or subjects, such as Products or Orders. Each subject then becomes a table.

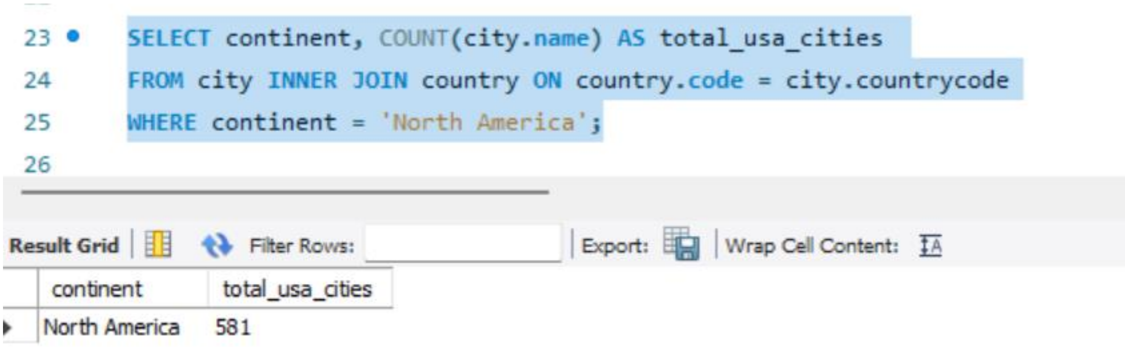
Once tables are created, it is important to make necessary relationships between tables. (e.g., how sales relate to inventory and customers)? One of the most common relationships between tables is the one-to-many relationship. Relationships between tables normally rely on the primary key (unique identifier) in one of the tables. An example { SELECT Title, Domestic_sales, International_sales FROM movies INNER JOIN Boxoffice ON movies.Id = Boxoffice.Movie_id }.

Equally important is mentioning SQL commands that are used to create the database, which is { CREATE DATABASE *databasename} and* for creating tables using of SQL syntax such such as { CREATE TABLE *new_table_name* AS SELECT *column1, column2,...* FROM *existing_table_name*
WHERE ....;}. To input initial data into the database, the use of SQL INSERT statements. INSERT INTO *table_name* (*column1*, *column2*, *column3*, ...) and
VALUES (*value1*, *value2*, *value3*, ...). This is crucial and can be measured to ensure the database remains accurate and up-to-date *bY using The UPDATE statement to modify the existing records in a table*

Data security represents the cornerstone of any backup strategy. Ensuring the safety and integrity of data is a paramount responsibility for data management and protection. The resilience of SQL database is enhanced by implementing reliable backup strategies in case of loss of data which can be caused by hardware failure, security breaches, or human error can cripple your business operations. Regular backups is critical to ensure the possibility to recover databases to mitigate the impact of data loss and demonstrates adherence to the legal frameworks. Some of starategies to backup database include, for example, using phpMyAdmin which is a web-based tool that makes it easy to manage MySQL databases. It offers a user-friendly interface for performing backups as following:Log into phpMyAdmin, Select the database you want to back up, Click the “Export” tab, Choose the export method and format, Download the backup file.The mysqldump utility is a command-line tool that generates logical backups: {mysqldump -u user -p database_name > backup.sql},copy, this command exports the entire database to a SQL file, which can be restored later using:{mysql -u user -p database_name < backup.sql}.

PART 2. SQL

1. **Count Cities in USA:** *Scenario:* You've been tasked with conducting a demographic analysis of cities in the United States. Your first step is to determine the total number of cities within the country to provide a baseline for further analysis.
   SELECT continent, COUNT(city.name) AS total_usa_cities
   FROM city INNER JOIN country ON country.code = city.countrycode
   WHERE continent = 'North America';



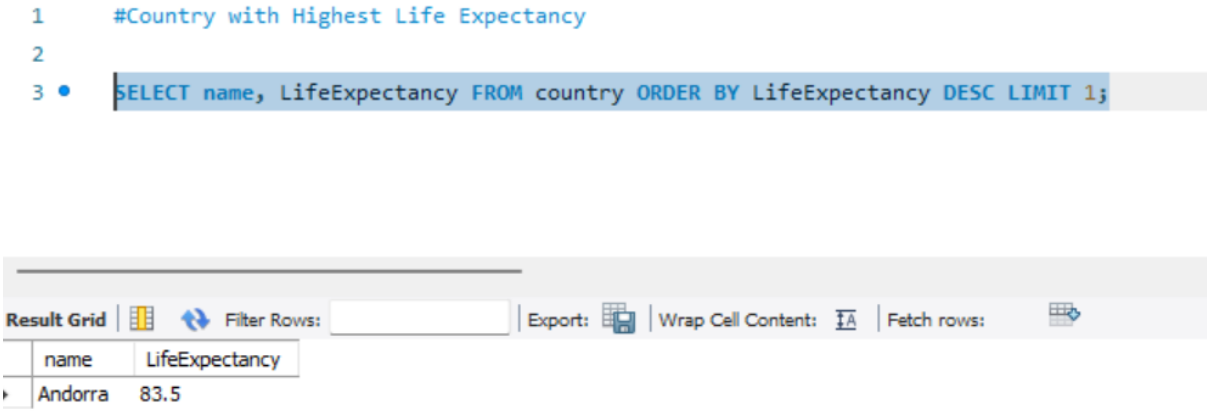
```
SELECT continent, COUNT(city.name) AS total_usa_cities
FROM city INNER JOIN country ON country.code = city.countrycode
WHERE continent = 'North America';
```

continent	total_usa_cities
North America	581

2. **Country with Highest Life Expectancy:** *Scenario:* As part of a global health initiative, you've been assigned to identify the country with the highest life expectancy. This information will be crucial for prioritizing healthcare resources and interventions.

   SELECT name, LifeExpectancy FROM country ORDER BY LifeExpectancy DESC LIMIT 1;

Result:



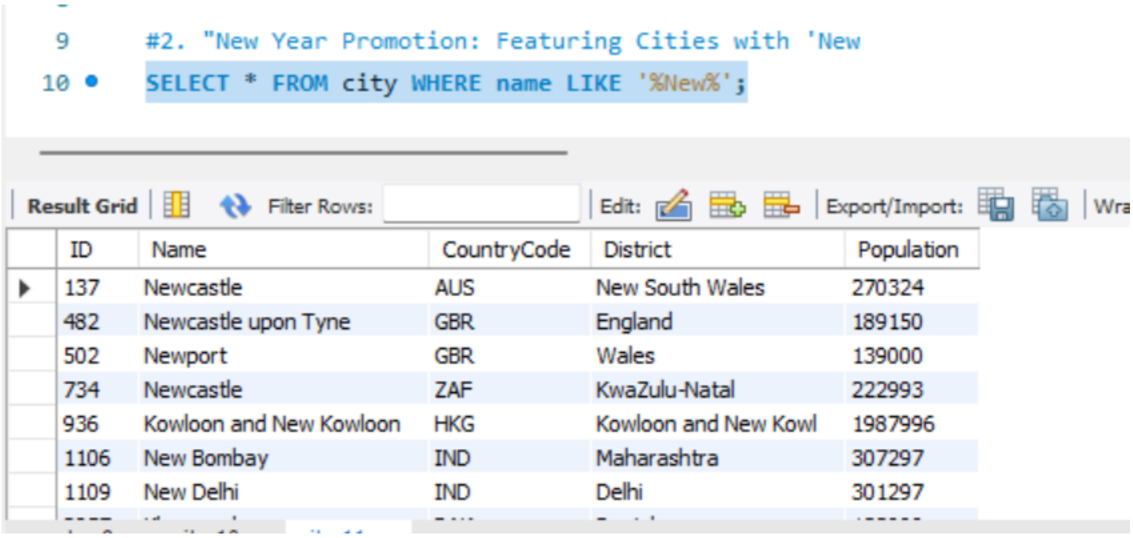
```
#Country with Highest Life Expectancy

SELECT name, LifeExpectancy FROM country ORDER BY LifeExpectancy DESC LIMIT 1;
```

name	LifeExpectancy
Andorra	83.5

3. **"New Year Promotion: Featuring Cities with 'New :** *Scenario:* In anticipation of the upcoming New Year, your travel agency is gearing up for a special promotion featuring cities with names including the word 'New'. You're tasked with swiftly compiling a list of all cities from around the world. This curated selection will be essential in creating promotional materials and enticing travellers with exciting destinations to kick off the New Year in style.

   *Statement: SELECT * FROM city WHERE name LIKE '%New%';*

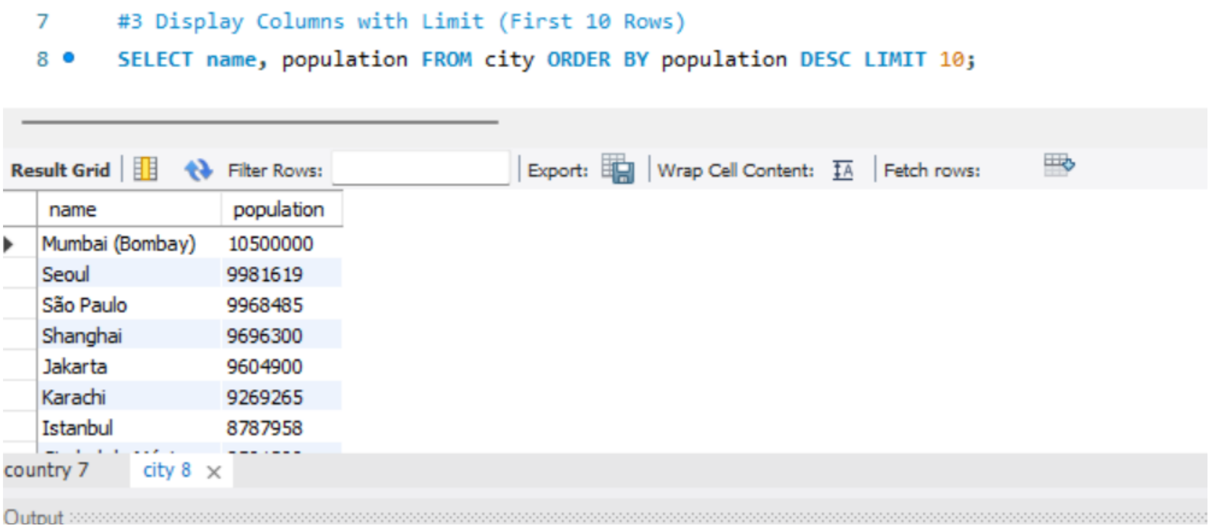


```
#2. "New Year Promotion: Featuring Cities with 'New
SELECT * FROM city WHERE name LIKE '%New%';
```

ID	Name	CountryCode	District	Population
137	Newcastle	AUS	New South Wales	270324
482	Newcastle upon Tyne	GBR	England	189150
502	Newport	GBR	Wales	139000
734	Newcastle	ZAF	KwaZulu-Natal	222993
936	Kowloon and New Kowloon	HKG	Kowloon and New Kowl	1987996
1106	New Bombay	IND	Maharashtra	307297
1109	New Delhi	IND	Delhi	301297

4. **Display Columns with Limit (First 10 Rows):** *Scenario:* You're tasked with providing a brief overview of the most populous cities in the world. To keep the report concise, you're instructed to list only the first 10 cities by population from the database.

Statement: SELECT name, population FROM city ORDER BY population DESC LIMIT 10;

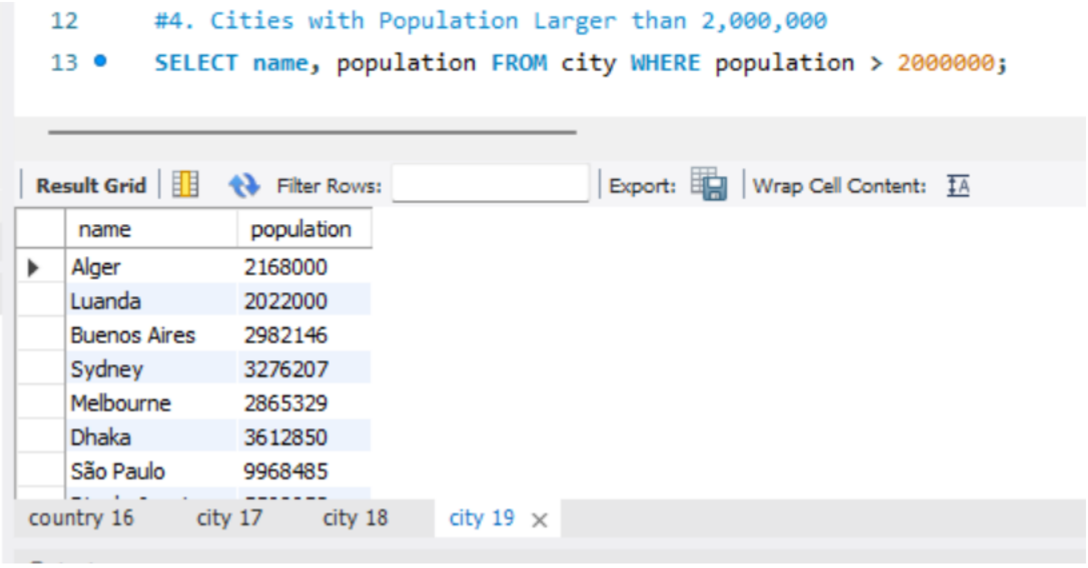


```
#3 Display Columns with Limit (First 10 Rows)
SELECT name, population FROM city ORDER BY population DESC LIMIT 10;
```

name	population
Mumbai (Bombay)	10500000
Seoul	9981619
São Paulo	9968485
Shanghai	9696300
Jakarta	9604900
Karachi	9269265
Istanbul	8787958

5. **Cities with Population Larger than 2,000,000:** *Scenario:* A real estate developer is interested in cities with substantial population sizes for potential investment opportunities. You're tasked with identifying cities from the database with populations exceeding 2 million to focus their research efforts.

   Statement : SELECT name, population FROM city WHERE population > 2000000;



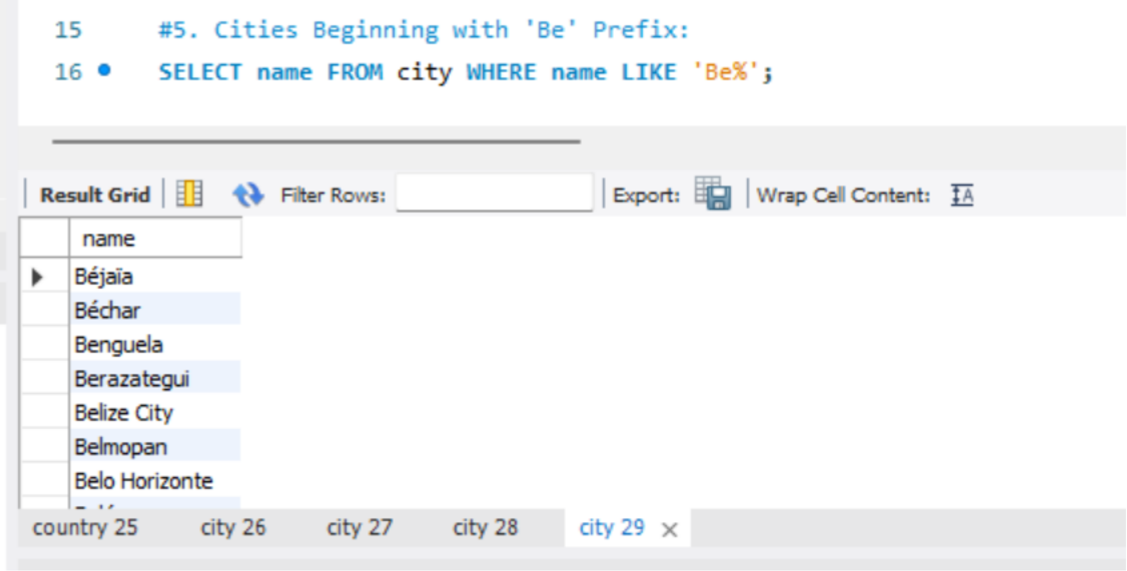
```
#4. Cities with Population Larger than 2,000,000
SELECT name, population FROM city WHERE population > 2000000;
```

name	population
Alger	2168000
Luanda	2022000
Buenos Aires	2982146
Sydney	3276207
Melbourne	2865329
Dhaka	3612850
São Paulo	9968485

6. **Cities Beginning with 'Be' Prefix:** *Scenario:* A travel blogger is planning a series of articles featuring cities with unique names. You're tasked with

compiling a list of cities from the database that start with the prefix 'Be' to assist in the blogger's content creation process.

Statement: *SELECT name FROM city WHERE name LIKE 'Be%';*



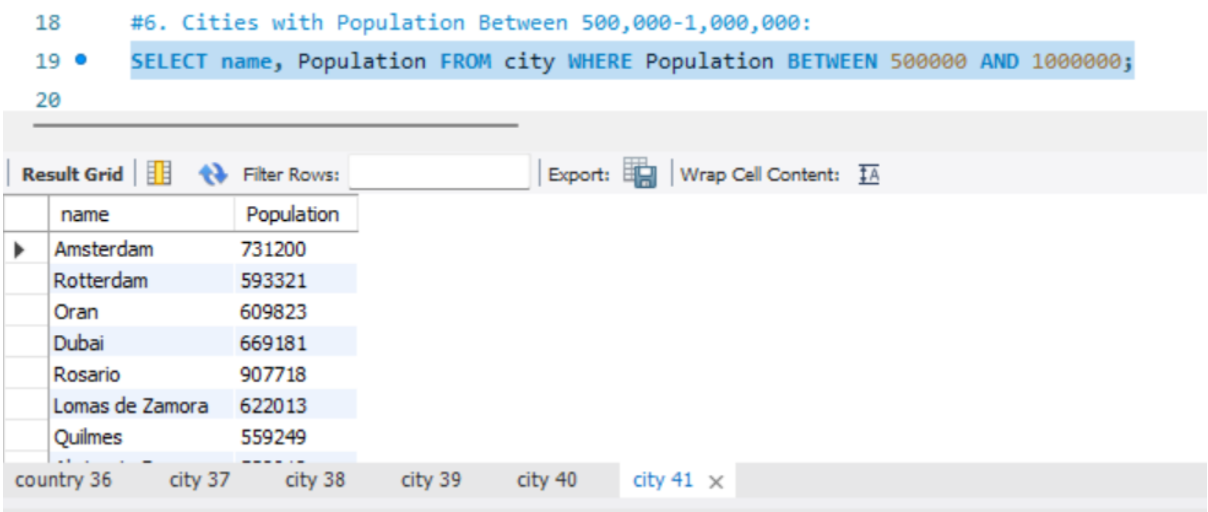
```
#5. Cities Beginning with 'Be' Prefix:
SELECT name FROM city WHERE name LIKE 'Be%';
```

name
Béjaïa
Béchar
Benguela
Berazategui
Belize City
Belmopan
Belo Horizonte

7. **Cities with Population Between 500,000-1,000,000:** *Scenario:* An urban planning committee needs to identify mid-sized cities suitable for infrastructure development projects. You're tasked with identifying cities with populations ranging between 500,000 and 1 million to inform their decision-making process.

   STATEMENT:

   SELECT name, Population FROM city WHERE Population BETWEEN 500000 AND 1000000;



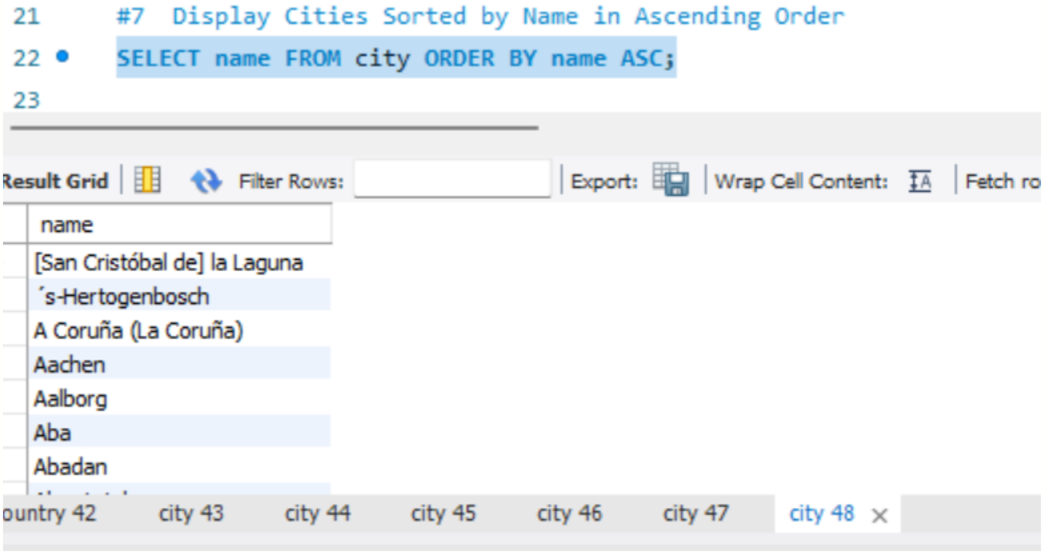
```
#6. Cities with Population Between 500,000-1,000,000:
SELECT name, Population FROM city WHERE Population BETWEEN 500000 AND 1000000;
```

name	Population
Amsterdam	731200
Rotterdam	593321
Oran	609823
Dubai	669181
Rosario	907718
Lomas de Zamora	622013
Quilmes	559249

8. **Display Cities Sorted by Name in Ascending Order:** *Scenario:* A geography teacher is preparing a lesson on alphabetical order using city names. You're tasked with providing a sorted list of cities from the database in ascending order by name to support the lesson plan.

   STATEMENT:

   SELECT name FROM city ORDER BY name ASC;



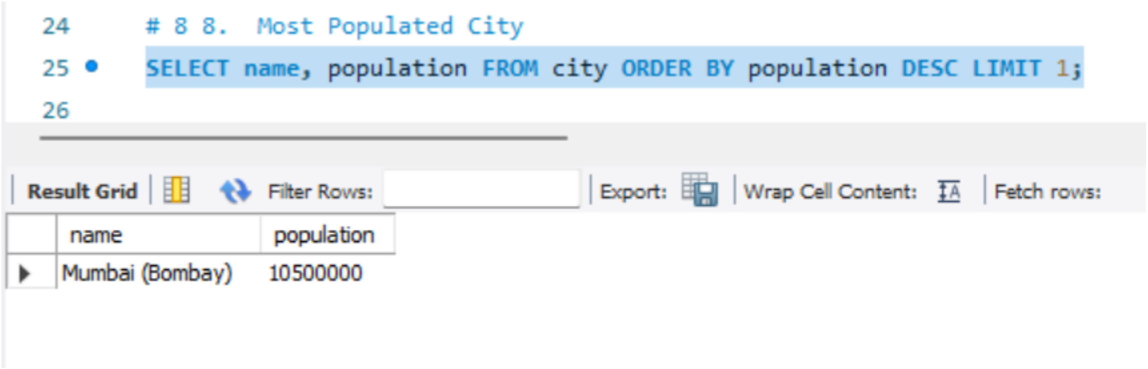
```
#7  Display Cities Sorted by Name in Ascending Order
SELECT name FROM city ORDER BY name ASC;
```

name
[San Cristóbal de] la Laguna
´s-Hertogenbosch
A Coruña (La Coruña)
Aachen
Aalborg
Aba
Abadan

9. **Most Populated City:** *Scenario:* A real estate investment firm is interested in cities with significant population densities for potential development projects. You're tasked with identifying the most populated city from the database to guide their investment decisions and strategic planning.

STATEMENT:

SELECT name, population FROM city ORDER BY population DESC LIMIT 1;



```
# 8 8.  Most Populated City
SELECT name, population FROM city ORDER BY population DESC LIMIT 1;
```

name	population
Mumbai (Bombay)	10500000

10. City Name Frequency Analysis: Supporting Geography Education

Scenario: In a geography class, students are learning about the distribution of city names around the world. The teacher, in preparation for a lesson on city name frequencies, wants to provide students with a list of unique city names sorted alphabetically, along with their respective counts of occurrences in the database. You're tasked with this sorted list to support the geography teacher's l

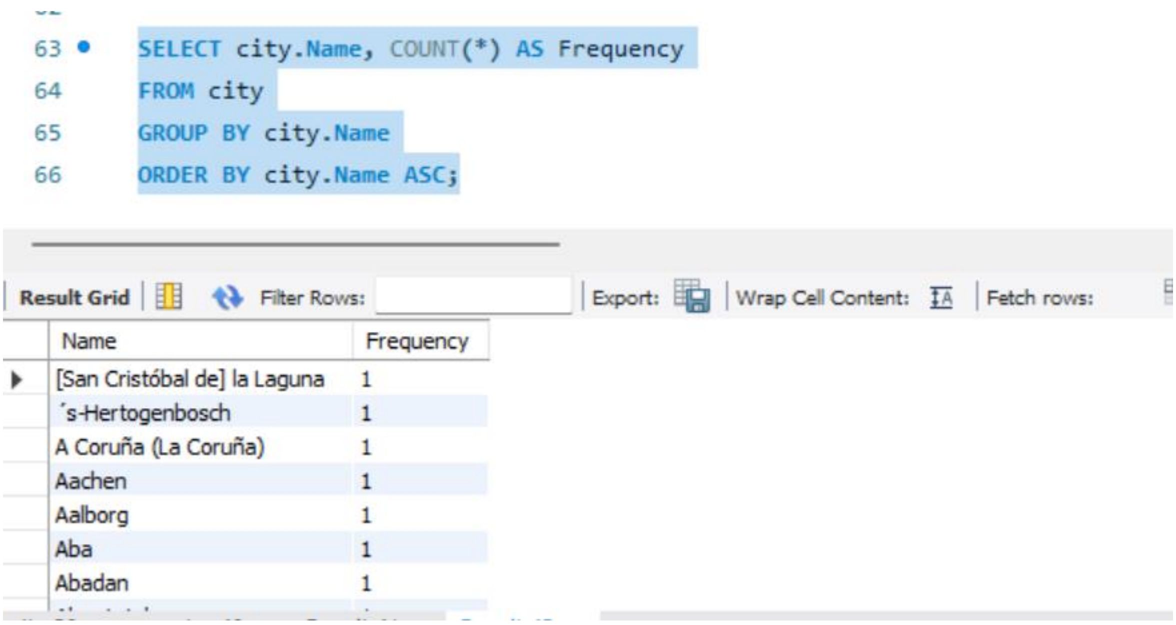
STATEMENT:

SELECT city.Name, COUNT(*) AS Frequency

FROM city

GROUP BY city.Name

ORDER BY city.Name ASC;



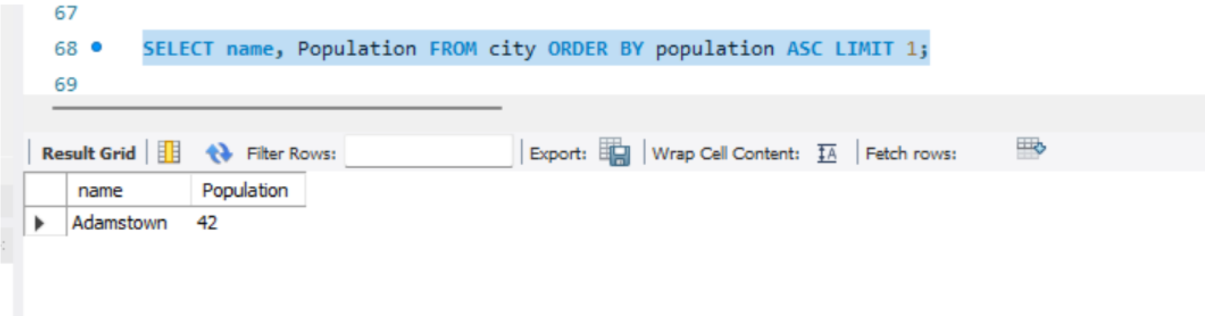
```
SELECT city.Name, COUNT(*) AS Frequency
FROM city
GROUP BY city.Name
ORDER BY city.Name ASC;
```

Name	Frequency
[San Cristóbal de] la Laguna	1
´s-Hertogenbosch	1
A Coruña (La Coruña)	1
Aachen	1
Aalborg	1
Aba	1
Abadan	1

11. **City with the Lowest Population:** *Scenario:* A census bureau is conducting an analysis of urban population distribution. You're tasked with identifying the city with the lowest population from the database to provide a comprehensive overview of demographic trends.

STATEMENT:

SELECT name, Population FROM city ORDER BY population ASC LIMIT 1;



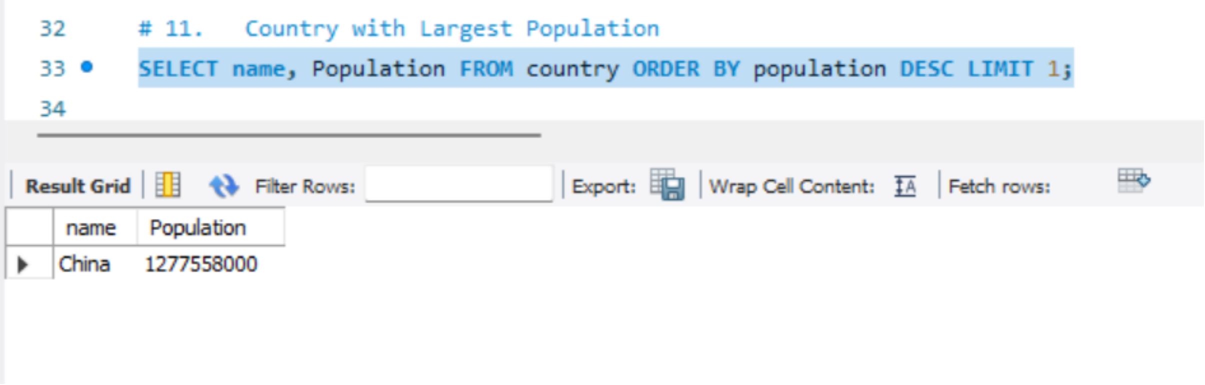
```
SELECT name, Population FROM city ORDER BY population ASC LIMIT 1;
```

name	Population
Adamstown	42

12. **Country with Largest Population:** *Scenario:* A global economic research institute requires data on countries with the largest populations for a comprehensive analysis. You're tasked with identifying the country with the

highest population from the database to provide valuable insights into demographic trends.

STATEMENT:
SELECT name, Population FROM country ORDER BY population DESC LIMIT 1;



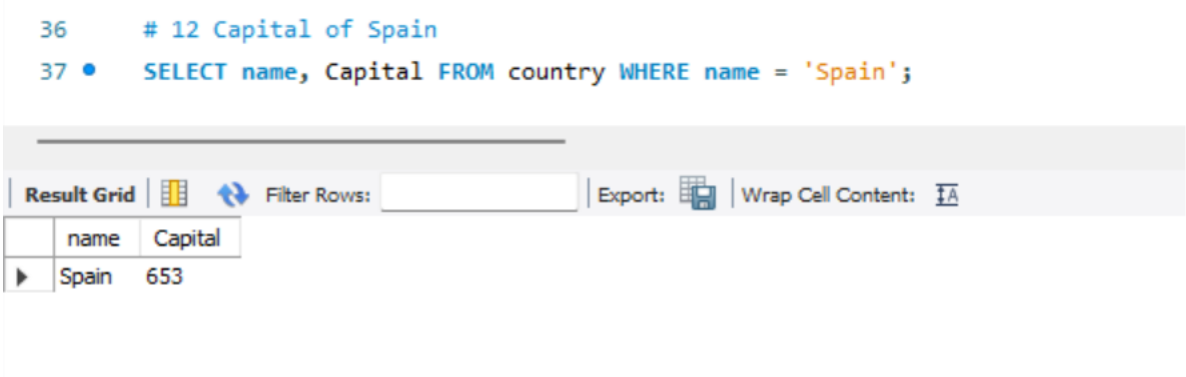
```
# 11.   Country with Largest Population
SELECT name, Population FROM country ORDER BY population DESC LIMIT 1;
```

name	Population
China	1277558000

13. **Capital of Spain:** *Scenario:* A travel agency is organizing tours across Europe and needs accurate information on capital cities. You're tasked with identifying the capital of Spain from the database to ensure itinerary accuracy and provide travellers with essential destination information.

Statement:

SELECT name, Capital FROM country WHERE name = 'Spain';

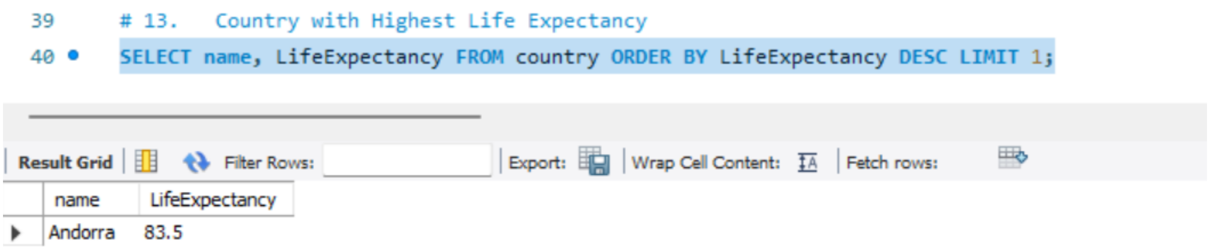


```
# 12 Capital of Spain
SELECT name, Capital FROM country WHERE name = 'Spain';
```

name	Capital
Spain	653

14. **Country with Highest Life Expectancy:** *Scenario:* A healthcare foundation is conducting research on global health indicators. You're tasked with identifying the country with the highest life expectancy from the database to inform their efforts in improving healthcare systems and policies.

STATEMENT:
SELECT name, LifeExpectancy FROM country ORDER BY LifeExpectancy DESC LIMIT 1;



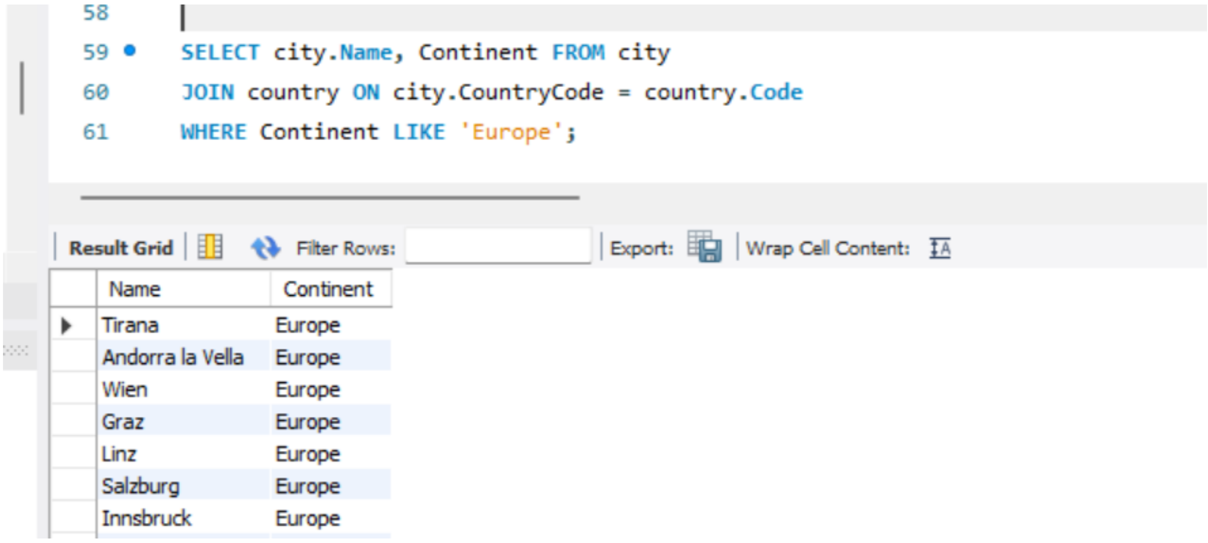
```
# 13.   Country with Highest Life Expectancy
SELECT name, LifeExpectancy FROM country ORDER BY LifeExpectancy DESC LIMIT 1;
```

name	LifeExpectancy
Andorra	83.5

15. **Cities in Europe:** *Scenario:* A European cultural exchange program is seeking to connect students with cities across the continent. You're tasked with compiling a list of cities located in Europe from the database to facilitate program planning and student engagement.

STATEMENT:

SELECT city.Name, Continent FROM city
JOIN country ON city.CountryCode = country.Code
WHERE Continent LIKE 'Europe';

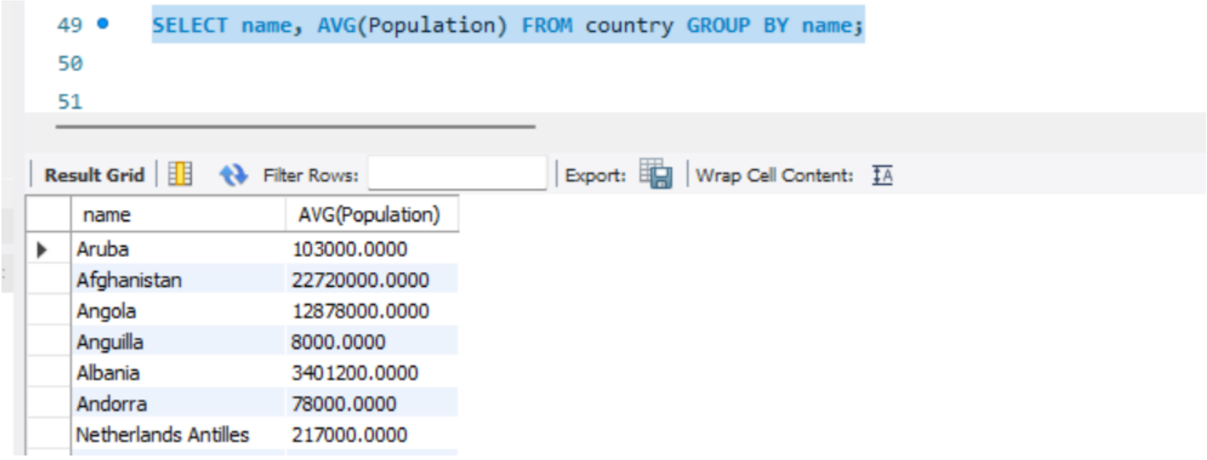


```sql
SELECT city.Name, Continent FROM city
JOIN country ON city.CountryCode = country.Code
WHERE Continent LIKE 'Europe';
```

Name	Continent
Tirana	Europe
Andorra la Vella	Europe
Wien	Europe
Graz	Europe
Linz	Europe
Salzburg	Europe
Innsbruck	Europe

16. **Average Population by Country:** *Scenario:* A demographic research team is conducting a comparative analysis of population distributions across countries. You're tasked with calculating the average population for each country from the database to provide valuable insights into global population trends.

    STATEMENT:
    SELECT name, AVG(Population) FROM country GROUP BY name;



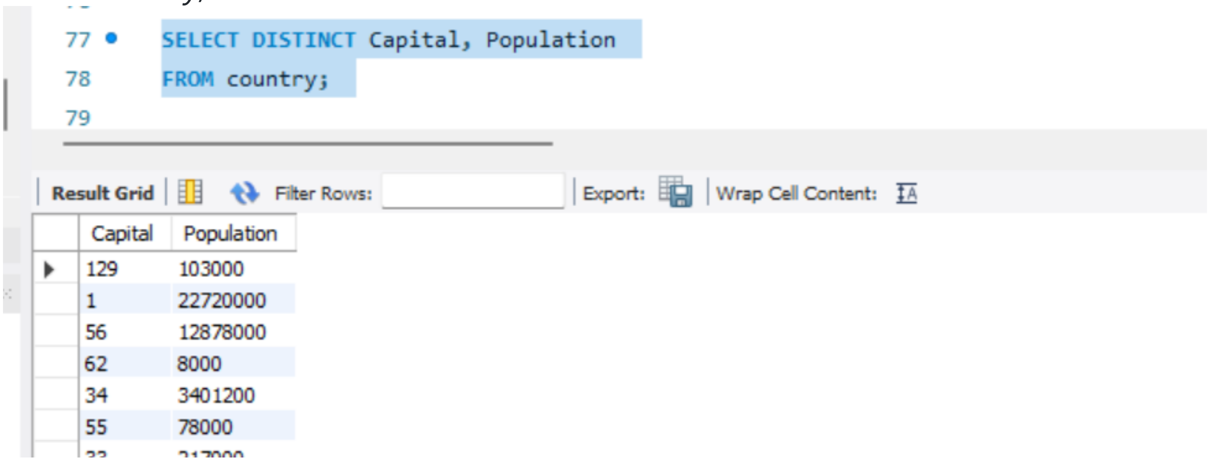
```sql
SELECT name, AVG(Population) FROM country GROUP BY name;
```

name	AVG(Population)
Aruba	103000.0000
Afghanistan	22720000.0000
Angola	12878000.0000
Anguilla	8000.0000
Albania	3401200.0000
Andorra	78000.0000
Netherlands Antilles	217000.0000

17. **Capital Cities Population Comparison:** *Scenario:* A statistical analysis firm is examining population distributions between capital cities worldwide. You're tasked with comparing the populations of capital cities from different countries to identify trends and patterns in urban demographics.

    STATEMENT:

SELECT DISTINCT Capital, Population
FROM country;



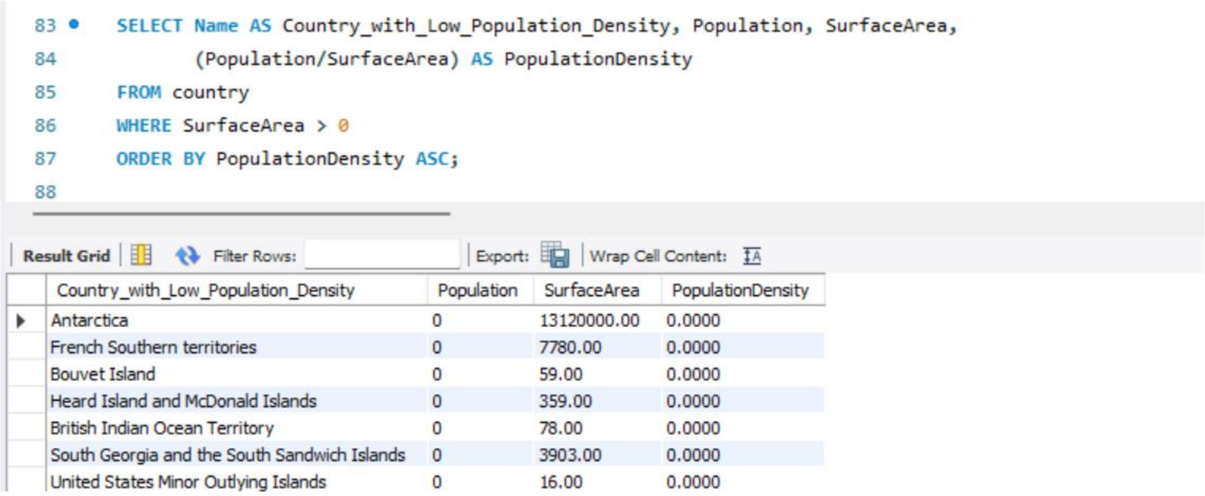
```
SELECT DISTINCT Capital, Population
FROM country;
```

Capital	Population
129	103000
1	22720000
56	12878000
62	8000
34	3401200
55	78000

18. **Countries with Low Population Density:** *Scenario:* An agricultural research institute is studying countries with low population densities for potential agricultural development projects. You're tasked with identifying countries with sparse populations from the database to support the institute's research efforts.

STATEMENT

SELECT Name AS Country_with_Low_Population_Density, Population, SurfaceArea,
(Population/SurfaceArea) AS PopulationDensity
FROM country
WHERE SurfaceArea > 0
ORDER BY PopulationDensity ASC;



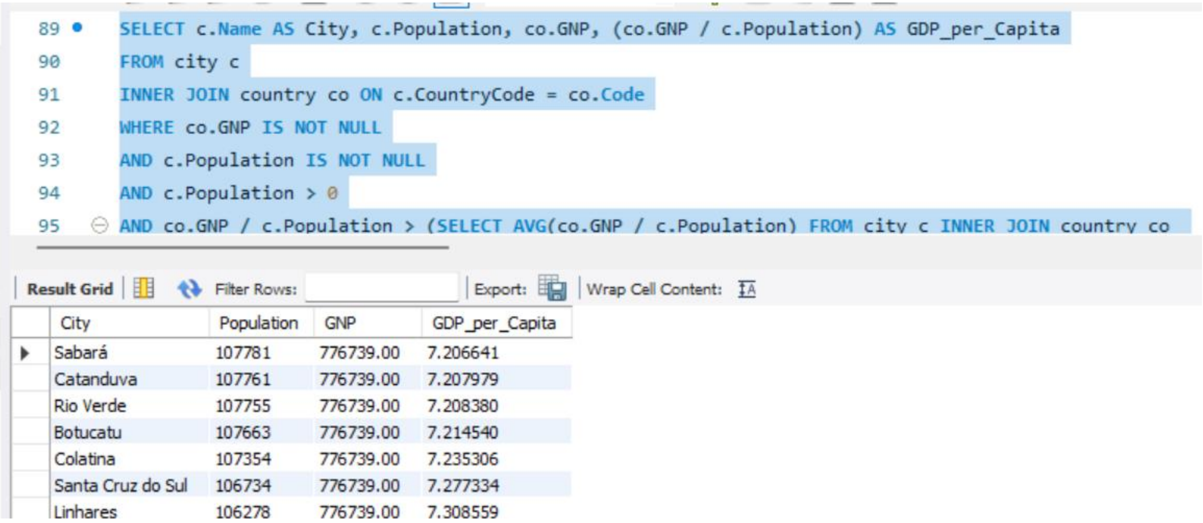
```
SELECT Name AS Country_with_Low_Population_Density, Population, SurfaceArea,
        (Population/SurfaceArea) AS PopulationDensity
FROM country
WHERE SurfaceArea > 0
ORDER BY PopulationDensity ASC;
```

Country_with_Low_Population_Density	Population	SurfaceArea	PopulationDensity
Antarctica	0	13120000.00	0.0000
French Southern territories	0	7780.00	0.0000
Bouvet Island	0	59.00	0.0000
Heard Island and McDonald Islands	0	359.00	0.0000
British Indian Ocean Territory	0	78.00	0.0000
South Georgia and the South Sandwich Islands	0	3903.00	0.0000
United States Minor Outlying Islands	0	16.00	0.0000

19. **Cities with High GDP per Capita:** *Scenario:* An economic consulting firm is analyzing cities with high GDP per capita for investment opportunities. You're tasked with identifying cities with above-average GDP per capita from the database to assist the firm in identifying potential investment destinations.

    STATEMENT:

    SELECT c.Name AS City, c.Population, co.GNP, (co.GNP / c.Population) AS GDP_per_Capita
    FROM city c
    INNER JOIN country co ON c.CountryCode = co.Code
    WHERE co.GNP IS NOT NULL
    AND c.Population IS NOT NULL
    AND c.Population > 0
    AND co.GNP / c.Population > (SELECT AVG(co.GNP / c.Population) FROM city c INNER JOIN country co ON c.CountryCode = co.Code WHERE co.GNP IS NOT NULL AND c.Population IS NOT NULL AND c.Population > 0);



```
SELECT c.Name AS City, c.Population, co.GNP, (co.GNP / c.Population) AS GDP_per_Capita
FROM city c
INNER JOIN country co ON c.CountryCode = co.Code
WHERE co.GNP IS NOT NULL
AND c.Population IS NOT NULL
AND c.Population > 0
AND co.GNP / c.Population > (SELECT AVG(co.GNP / c.Population) FROM city c INNER JOIN country co
```

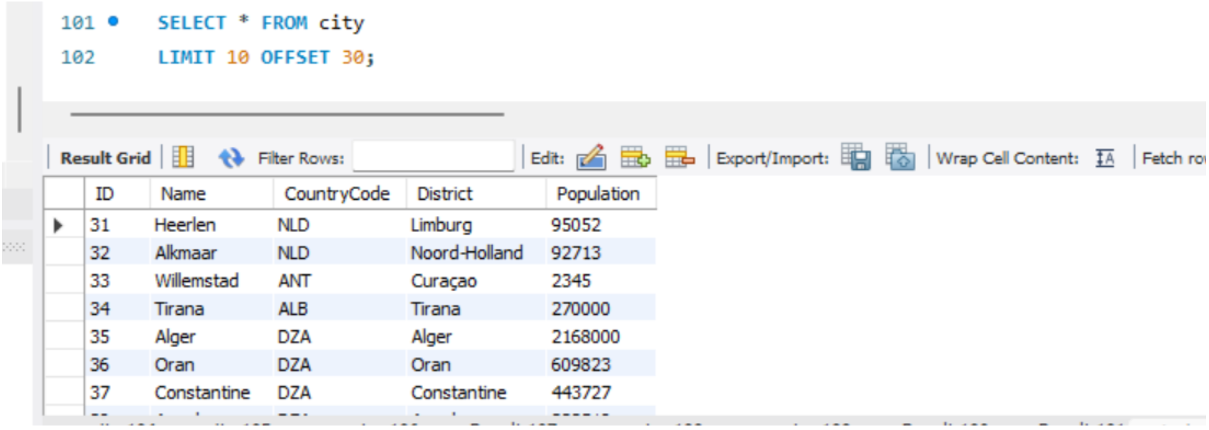
City	Population	GNP	GDP_per_Capita
Sabará	107781	776739.00	7.206641
Catanduva	107761	776739.00	7.207979
Rio Verde	107755	776739.00	7.208380
Botucatu	107663	776739.00	7.214540
Colatina	107354	776739.00	7.235306
Santa Cruz do Sul	106734	776739.00	7.277334
Linhares	106278	776739.00	7.308559

20. **Display Columns with Limit (Rows 31-40):** *Scenario:* A market research firm requires detailed information on cities beyond the top rankings for a comprehensive analysis. You're tasked with providing data on cities ranked between 31st and 40th by population to ensure a thorough understanding of urban demographics.
21.

    STATEMENT:

*SELECT * FROM city*

LIMIT 10 OFFSET 30;



```sql
SELECT * FROM city
LIMIT 10 OFFSET 30;
```

ID	Name	CountryCode	District	Population
31	Heerlen	NLD	Limburg	95052
32	Alkmaar	NLD	Noord-Holland	92713
33	Willemstad	ANT	Curaçao	2345
34	Tirana	ALB	Tirana	270000
35	Alger	DZA	Alger	2168000
36	Oran	DZA	Oran	609823
37	Constantine	DZA	Constantine	443727