

**WORLD dATABASE tASKs**

**EXPLORING the World DB using SQL**

A rainbow colored planet with numbers

Description automatically generated

**Evangeline Mapstone**

**10/05/2024**

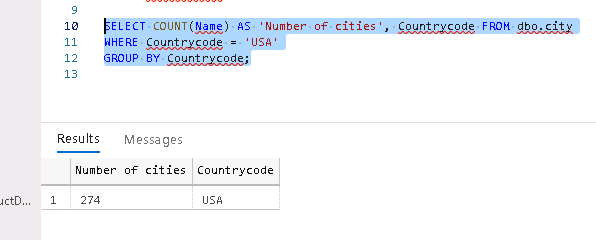
**Tasks**

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 COUNT(Name) AS 'Number of cities', Countrycode FROM dbo.city

WHERE Countrycode = 'USA'

GROUP BY Countrycode;

This query returns the number of cities in the USA from the ‘City’ table, by using GROUP BY to filter the results where ‘Countrycode’ = ‘USA’.

1. **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 dbo.country

WHERE LifeExpectancy= (SELECT MAX(LifeExpectancy) as LifeExpectancy

From dbo.country)

This query will return the country which has the highest life expectancy. The subquery narrows down the search to find this specific data.

A screenshot of a computer

Description automatically generated

1. **"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.

SELECT Name, District, CountryCode FROM dbo.city

WHERE Name LIKE '% NEW %' OR Name LIKE 'New %' OR Name LIKE '% Name'

A screenshot of a computer

Description automatically generatedOrder BY Name

This query returns the names of all the cities from around the world that include the word ‘New’. Using ‘WHERE Name LIKE ‘NEW%’’, specifically returns cities that start with the word ‘New’ thus keeping in line with the New Year theme. District and Countrycode are included to show travellers which part of the world these cities are located in. I have only included the cities where the word ‘New’ is distinctive and not when it is part of a city name i.e. ‘Newcastle’. This could be included if required.

1. **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.

SELECT TOP (10) Name, Population FROM dbo.city

Order BY Population DESC;

A screenshot of a computer

Description automatically generated

This query returns the top 10 most populated Cities in the world. The DESC keyword orders the cities starting with the ones that are the most populous.

1. **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.

SELECT Name, Population FROM dbo.city

WHERE Population > 2000000

ORDER BY Population

A screenshot of a computer

Description automatically generatedThis query returns all the names of the cities which populations exceed 2 million. The list is ordered by Population.

1. **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.

Select Name From dbo.city

WHERE Name Like 'Be\_%'

Order by Name

A screenshot of a computer

Description automatically generatedThis query is needed to return the list of all the cities which names have the prefix ‘Be’. The WHERE clause and LIKE operator are applied to the Name column to narrow down this search.

1. **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.

SELECT Name, Population

FROM Dbo.City

WHERE Population BETWEEN 500000 AND 1000000

ORDER BY Population

A screenshot of a computer

Description automatically generated

This query returns the names of all the cities where the population is between 500,000 and 1 million. The BETWEEN operator is inclusive; it displays the cities that have a population of 500,000 and 1 million as well as the cities in between.

A screenshot of a computer

Description automatically generated

1. **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.

SELECT Name

FROM dbo.City

ORDER BY Name

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedThis query returns the names of City names and lists them in alphabetical order.

A screenshot of a computer

Description automatically generated

1. **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.

SELECT Name, Population FROM dbo.city

Where Population =

(SELECT MAX(Population) As Population

FROM dbo.city)

This query returns the name of the most populated city, the subquery includes the MAX aggregate function to narrow down the search to find the most populated city only.

A screenshot of a computer

Description automatically generated

1. **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

SELECT DISTINCT Name, COUNT(\*) AS 'Distribution of City Names'

FROM dbo.city

Group By Name

ORDER BY Name;

A screenshot of a computer

Description automatically generatedThis query returns the City names and the number of their occurrences across the world. The COUNT(\*) function is used to find the distribution of city names. Then GROUP BY Name. The ORDER BY Name lists the cities in alphabetic order.

1. **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.

SELECT Name, Population FROM dbo.city

Where Population =

(SELECT MIN(Population) As Population

FROM dbo.city)

A close-up of a box

Description automatically generatedThis query returns the name of the city with the lowest population. It includes the subquery that was used to find the highest populated city for task 9 however this time I use the MIN aggregate function to find the lowest populated city only.

1. **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.

SELECT Name, Population FROM dbo.country

Where Population =

(SELECT MAX(Population) As Population

FROM dbo.country)

A screenshot of a computer

Description automatically generated

This query returns the name of the country with the highest population. The Subquery useful for when trying to identify which countries have the largest or smallest populations. It can also be used when trying to find the highest/lowest Life Expectancy or Surface Area.

1. **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.

SELECT dbo.country.Name AS 'Country Name', dbo.city.name AS 'Capital City'

FROM dbo.country

JOIN dbo.city

ON dbo.city.ID = dbo.country.Capital

WHERE dbo.country.Name = 'Spain'

A screenshot of a computer

Description automatically generated

This query returns the capital city of Spain using the JOIN clause to extract the Country from the dbo.country table and the City from the dbo.city table. The logical relationship between the tables is where the ‘ID’ column in the dbo.city table refers to the ‘Capital’ column in the dbo.Country table.

1. **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.

SELECT Name, LifeExpectancy

FROM dbo.Country

ORDER BY LifeExpectancy DESC;

A screenshot of a computer

Description automatically generatedThis query returns the names of all the countries and their Life Expectancies from highest to lowest life expectancy by DESCENDING order.

1. **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.

SELECT dbo.country.Continent, dbo.city.Name

FROM dbo.country

JOIN dbo.city

A screenshot of a computer

Description automatically generatedON dbo.city.Countrycode = dbo.country.Code

WHERE Continent = 'Europe'

This query returns the list of all the cities in Europe. By using the JOIN clause, the data is queried and accessed from the dbo.city table and the dbo.country table based on the logical relationship between these tables. The relationship being ‘Countrycode’ in the City table refers to ‘Code’ in the Country table.

1. **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.

SELECT AVG(dbo.city.Population) AS 'Average City Population', dbo.country.NAME

AS 'Country'

A screenshot of a data

Description automatically generatedFROM dbo.city

JOIN dbo.country

ON dbo.city.Countrycode = dbo.country.Code

GROUP BY dbo.country.Name

ORDER BY dbo.country.Name

This query returns the average city populations across all the countries in the world. The first step is to run a query to find the average population in all the cities. Then include this query in a JOIN query between the City and Country. To for this query the logical relationship is where the ‘Countrycode’ column in the City table refers to the ‘Code’ column in the Country table.

1. **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.

SELECT dbo.city.Population AS 'Population', dbo.city.name AS 'Capital City',

dbo.country.NAME AS 'Country'

FROM dbo.city

JOIN dbo.country

ON dbo.city.ID = dbo.country.Capital

ORDER BY 'Population' DESC, 'Country'

A screenshot of a computer

Description automatically generated

This query returns the population of all the capital cities and lists them in order from the most to least populated and then order by the countries.

1. **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.

Select Name, Population /SurfaceArea AS 'Population Density'

FROM dbo.country

Order BY 'Population Density', Name

This query returns the Names and Population Density of each country by dividing the population total of each country with its surface area. The results are ordered by Population Density starting with the lowest populated countries first.

A screenshot of a computer

Description automatically generated

**Bonus Challenges on the next page**

BONUS TASKS: Challenge yourself: These are optional tasks. Feel free to skip.

1. **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.

SELECT dbo.City.Name AS 'City', dbo.Country.GNP/dbo.Country.population as 'GNP/Capita'

FROM dbo.Country

JOIN dbo.City

ON dbo.Country.Code = dbo.City.CountryCode

WHERE dbo.Country.GNP/dbo.Country.Population >

(SELECT AVG(GNP/Population)

A screenshot of a computer

Description automatically generatedFROM dbo.Country

WHERE Population <> 0)

and dbo.Country.Population <> 0

1. A screenshot of a computer

   Description automatically generated**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.

SELECT Name, Population

FROM dbo.City

ORDER BY Population DESC OFFSET 30 ROWS

FETCH NEXT 10 ROWS only;

To complete this task run a query that will return the names of all the cities and their population from the City table by DESCENDING population order so the list starts from the city that is ranked the highest in population. Then add OFFSET 30 Rows and FETCH NEXT 10 rows only to skip the top 30 cites ranked by population and show the cities that are ranked 31st to 40th Population.