08/05/2024

Elizabeth Muir

World Database

SQL

# Tasks

1. **Count Cities in the 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.

A screenshot of a computer program

Description automatically generated

This query gets back the total amount of cities in the USA. We select the CountryCode and use COUNT to count for us where it matches ‘USA’ and put it into the column labelled ‘Amount of cities in 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 prioritising healthcare resources and interventions.

A screenshot of a computer

Description automatically generated

This query displays the country that has the highest life expectancy. We select the name of the country and their life expectancy column and then add in a sub query to display the top number of life expectancy from the country table.

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 of the New Year in style.

A screenshot of a computer

Description automatically generated

This query displays all the cities that have new in their name. We select everything from the city table and put in a condition of where the name has ‘new’ in any part of it.

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.

A screenshot of a computer screen

Description automatically generated

This query displays the first 10 cities in the world. As we are using TSQL we have to use the TOP keyword for us to limit the results compared to other versions will allow us to us LIMIT. I selected the name and the population from the city table and the top 10 cities.

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

A screenshot of a computer screen

Description automatically generated

This query shows cities in the database that has a population of 2 million and above. We first select the name and population from city table and put a where clause to ensure that we display the population that is 2 million and above.

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.

A screenshot of a computer

Description automatically generated

This query displays a list of cities that start with ‘Be’. I first select the name out of cities table and put a where clause where the name has ‘Be’ at the start of the name.

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.

A screenshot of a computer

Description automatically generated

This query shows cities and their population that ranges between 500,000 and 1 million. First, I select name and population from the city table. I then put a where clause to display the population between 500,000 and 1 million.

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.

A screenshot of a computer

Description automatically generated

This query

displays a sorted list of cities that are listed in ascending order. I first selected name from the city table and then order by the name in ascending order.

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.

A screenshot of a computer

Description automatically generated

This query displays the most populated city from the database. I first select name and population from the city table. I then put a where clause selecting population from the city table but using the MAX function to select the highest populated city.

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 lesson.

A screenshot of a computer

Description automatically generated

This query displays a sorted list of cities and the number of times they are displayed in the database. I select name and count so it can count how many times it appears in the database. I then name a column frequency appeared to display the result. I then order and group it by name in ascending 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.

A screenshot of a computer

Description automatically generated

This query displays the city with the lowest population. I first selected name and population from city table. I then want to make sure that we are getting the lowest, so we use MIN function to do that within our subquery.

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.

A screenshot of a computer

Description automatically generated

The query displays the country with the highest population. Again, I select the name and population from the country table and then use a where clause to select the MAX from the population column.

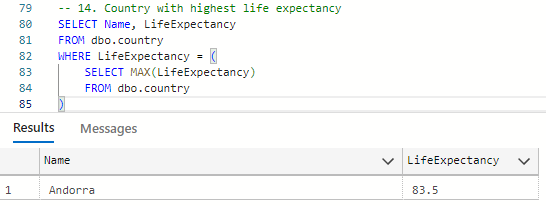
1. **Capital of Spain:** *Scenario* – A travel agency is organising tours across Europe and needs accurate information on capital cities. You’re tasked with identifying the capital of Spain from the database to ensure itinerary accuracy and provide travellers with essential destination information.

A screenshot of a computer

Description automatically generated

This query identifies the capital of Spain. I first select name from the city table and give it a column name of capital. I then join with the country table using the id off the city table and the capital off the country table. I then use a where clause to ensure we only want to see the capital ‘Spain’.

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.



For this query I am displaying the country with the highest life expectancy. I first select name and life expectancy from the country table. I use a where clause to use the MAX function to display the highest life expectancy.

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.

A screenshot of a computer

Description automatically generated

The query displays cities in the continent of Europe. I first select Name and display the column as Cities in Europe from the country table. I then use WHERE to ensure that we are selecting only European cities.

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.

A screenshot of a computer screen

Description automatically generated

This query displays the average population for each country. I had to use the AVG function to ensure that we got the correct number. I also select name so we can see what country has for population on average. I grouped by the name so that it’s displayed in ascending order

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.

A screenshot of a computer

Description automatically generated

This query is to display the capital cities of each country and their population. To do this we have to join country and city tables. I have selected the country name with the city name and population.

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.

A screenshot of a computer

Description automatically generated

The query displays the countries and their populations. To calculate we have to take the population and divide it to get our population density to display with each country.

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 analysing cities with high GDP per capita for investment opportunities. You’re tasked with identifying cities with about-average GDP per capita from the database to assist the firm in identifying potential investment destinations.

A screenshot of a computer

Description automatically generated

This query was a tricky one and this is what I produced. It does not work at the moment, but I am in the process of trying to understand what needs fixing to process a correct answer.

1. **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.

A screenshot of a computer

Description automatically generated

This query displays all the information from the city table. We are also just displaying entries between 31 and 40 that is on the table. We use OFFSET to tell us where to start after the number we have selected e.g. Here we stated 30, this will display records from 31 onwards. When then use Fetch to state we want the next 10 rows only.