### User:A3nm/SD202 2021 ex1

#### < User:A3nm

Welcome to the first lab of the SD202 class! The goal of this lab is to learn how to query data using an SQL database.

In this first exercise, we will query a single world table. The table contains information about the world's countries. The table has the following fields:

- name varchar (50): the English name of the country (primary key)
- continent varchar(60): the English name of the continent
- area decimal (10, 0): the area in square kilometers
- population decimal (11, 0): the population
- gdp decimal (14, 0): the GDP (we will not use it in the exercise)
- capital varchar (60): the English name of the capital of the country
- tld varchar(5): the Top Level Domain associated to the country (we will not use it)
- flag varchar (255): the URL to the flag (we will not use it)

Here is an excerpt of the first 5 rows of the table:

name	continent	area	population	gdp	capital	tld	flag
Afghanistan	Asia	652230	25500100	20364000000	Kabul	.af	//upload.wikimedia.org/wikipedia/commons/9/9a/Flag
Albania	Europe	28748	2821977	12044000000	Tirana	.al	//upload.wikimedia.org/wikipedia/commons/3/36/Flag
Algeria	Africa	2381741	38700000	207021000000	Algiers	.dz	//upload.wikimedia.org/wikipedia/commons/7/77/Flag
Andorra	Europe	468	76098	3222000000	Andorra la Vella	.ad	//upload.wikimedia.org/wikipedia/commons/1/19/Flag
Angola	Africa	1246700	19183590	116308000000	Luanda	.ao	//upload.wikimedia.org/wikipedia/commons/9/9d/Flag

**Warning:** contrary to what the SQL standard prescribes, table names are case-sensitive in the SQL engine used in this website. So the table must be called world, **not** World, otherwise the evaluation will fail (with the error: Table 'gisq.World' doesn't exist).

Using the world table, your task will be to write queries satisfying some requirements. Sometimes, an example query is provided for you submitting a query, you will get its result, and you can compare it to the expected result if the two do not match. **Warning:** the order columns) is taken into account when checking an answer.

Cheat mode

#### **Contents**

**Basics: Countries and continents** 

**Filtering names** 

**Population** 

**Boolean conditions** 

**String operations** 

Ordering

**Computation and aggregation** 

**Advanced queries** 

Acknowledgements

#### **Basics: Countries and continents**

#### 1.

Modify the query below to select the name of all countries in Europe. Hint: use the WHERE clause:

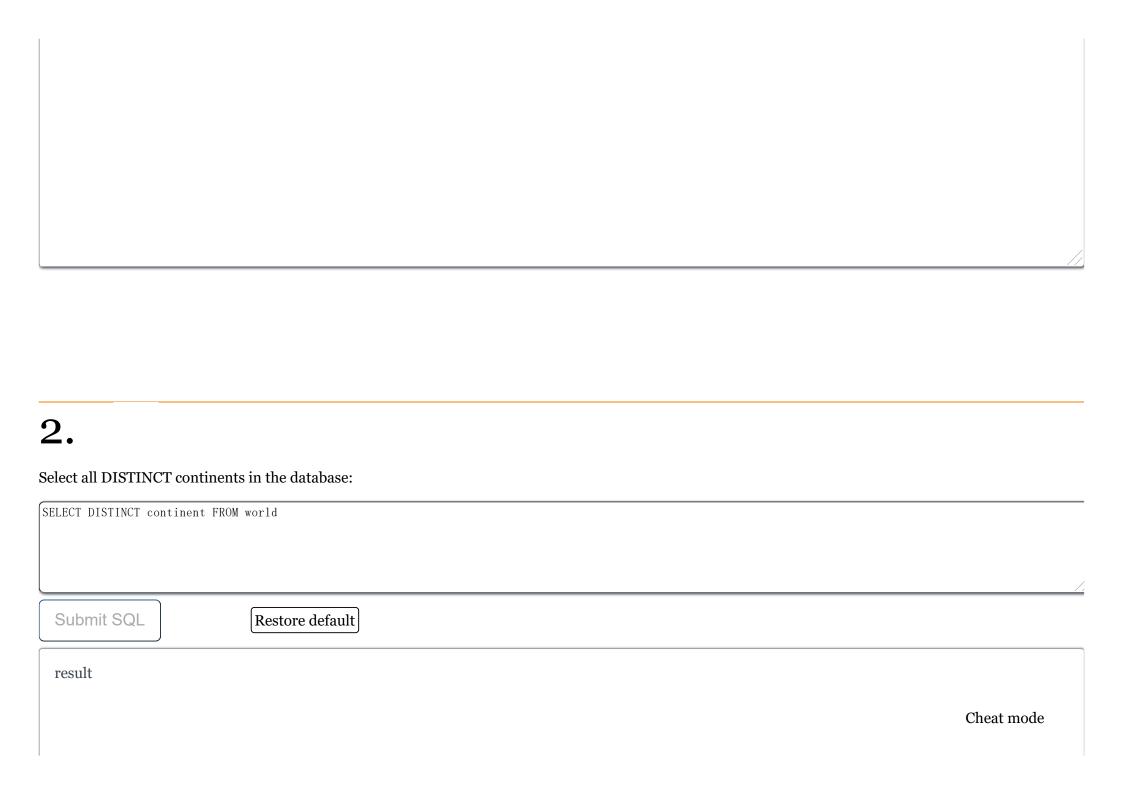
SELECT name FROM world WHERE continent = 'Europe'

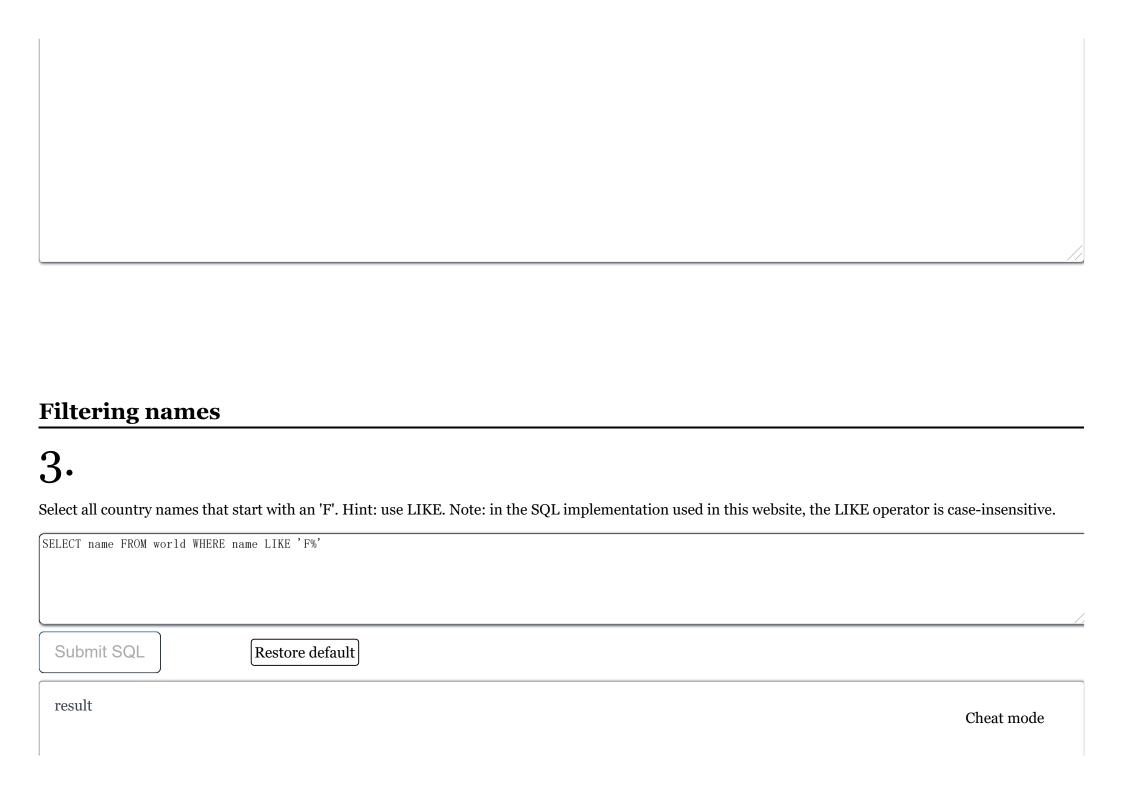
Submit SQL

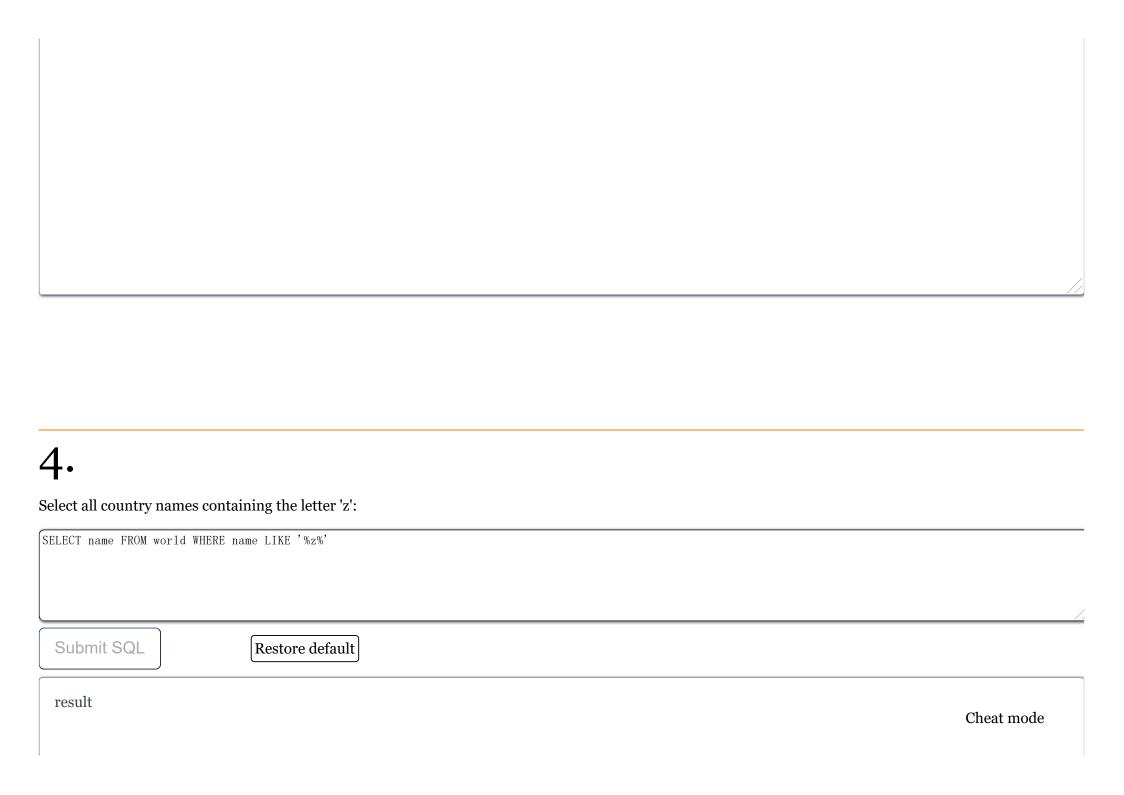
Restore default

result

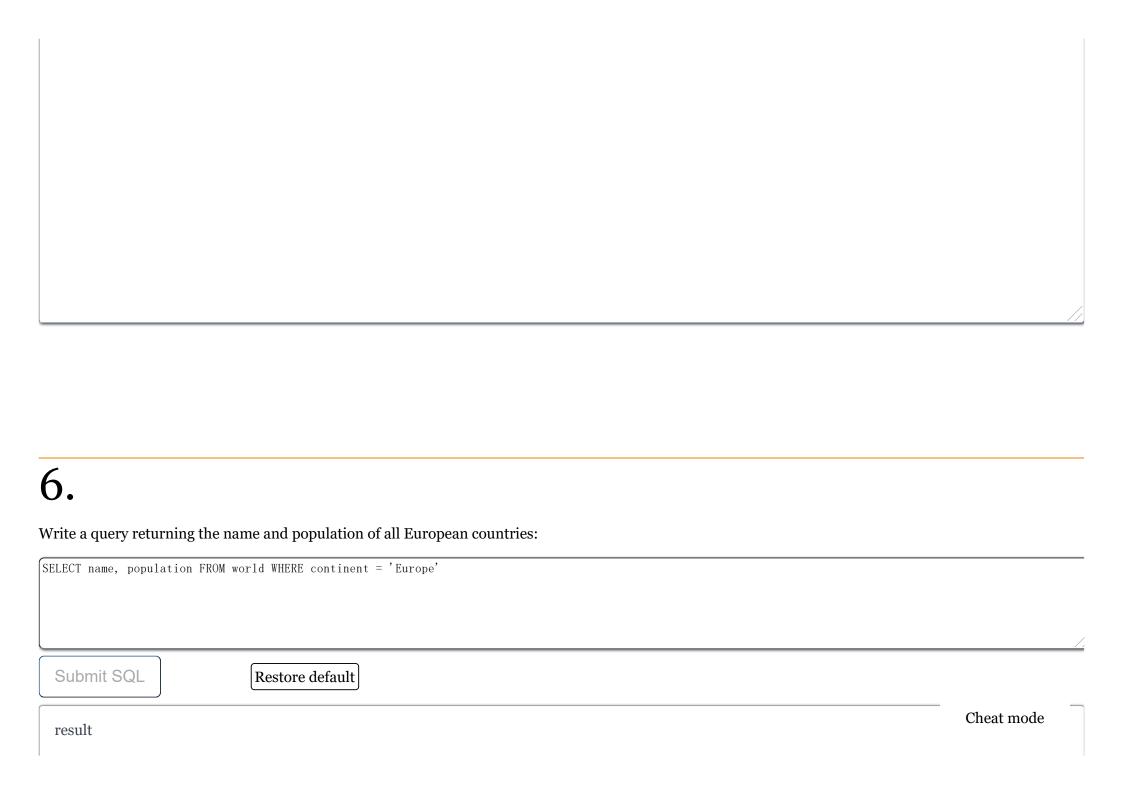
Cheat mode

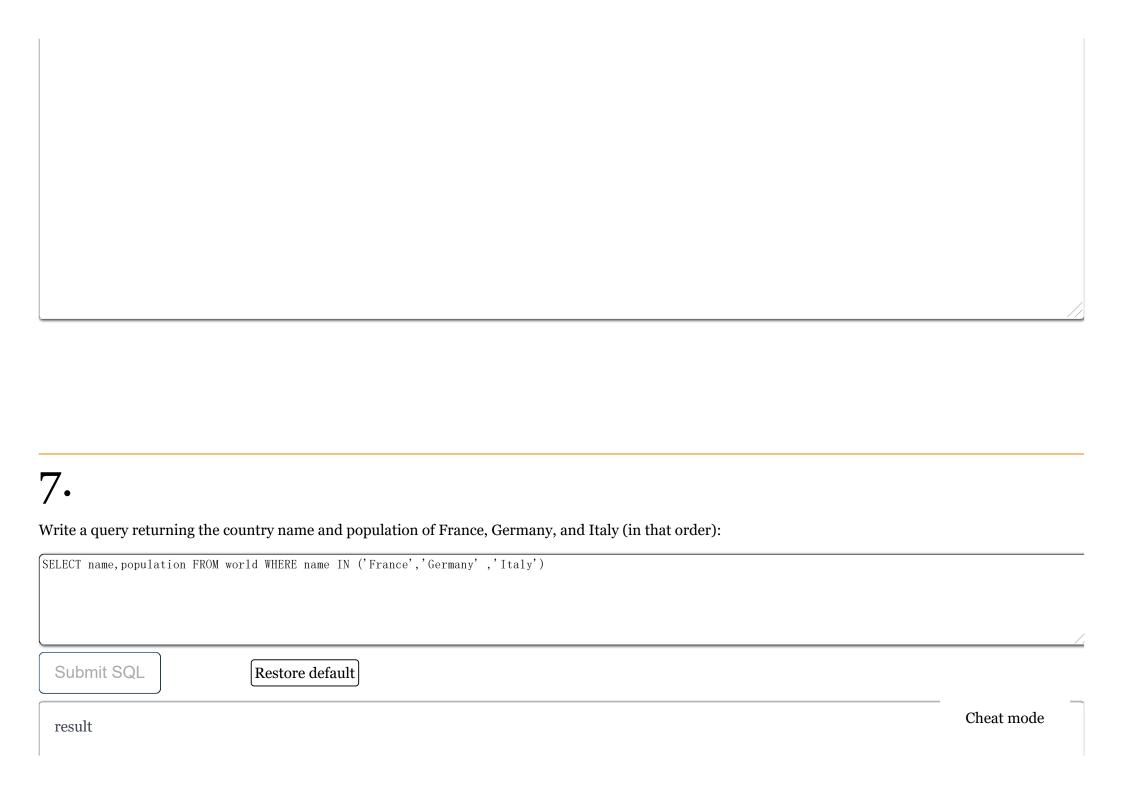


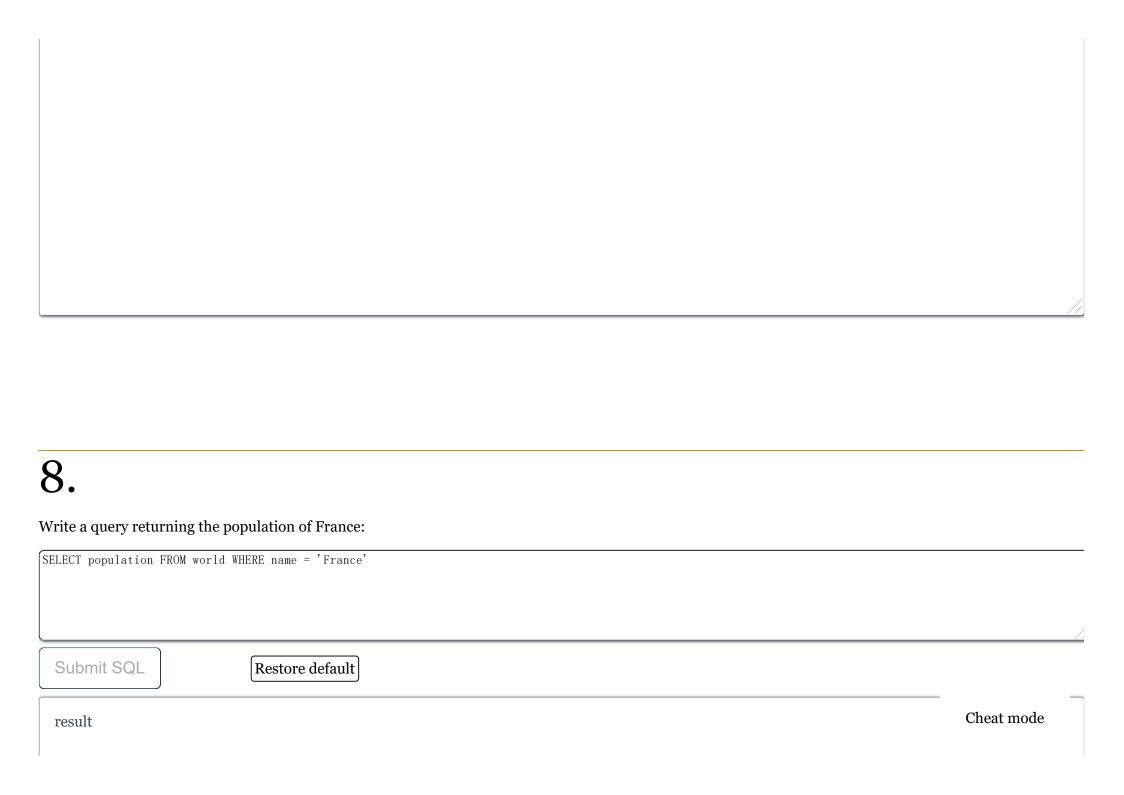


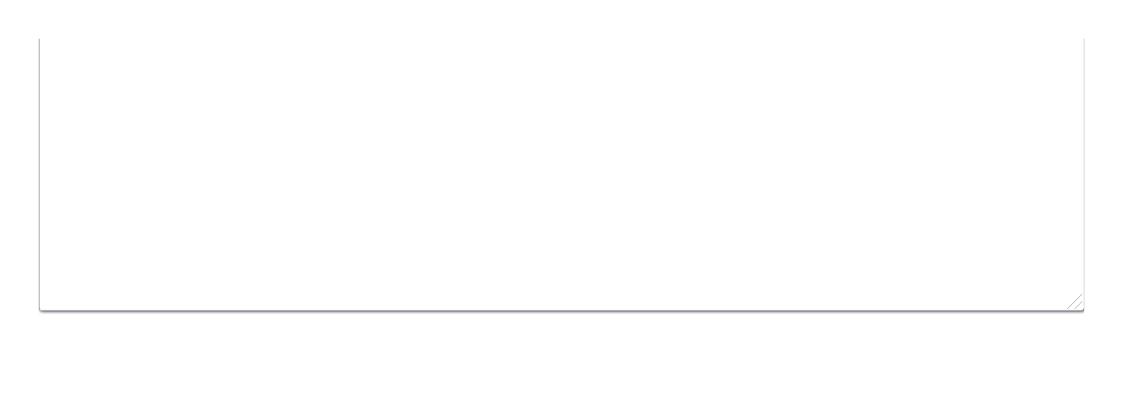


Population	
<b>5</b> • Select all country names having at most one million inhabitants:	
SELECT name FROM world WHERE population <= 1000000	
Submit SQL Restore default  result	Cheat mode









#### **Boolean conditions**

9.

Write a query selecting the name, population, and area of all countries that have at least 100 million inhabitants or have an area of at least 3 million square kilometers (or both). Order the results by name.

SELECT name, population, area FROM world WHERE population >= 100000000 OR area >=3000000 ORDER BY name

Submit SQL

Restore default

Cheat mode

result	

Write a query selecting the name, population, and area of all European countries which have population greater than 50 million or area greater than 500000 square kilometers (or both), ordered by name. Hint: be careful!

SELECT name, population, area FROM world WHERE continent = 'Europe' AND (population >= 50000000 OR area >= 500000) ORDER BY name Submit SQL Restore default result

### 11.

Write a query selecting the name, population, and area of all countries that are either large in population or large in size, but not bo countries that have at least 100 million inhabitants OR have an area of at least 3 million square kilometers, but not both. Order the results by name

SELECT name, population, area FROM world WHERE population >= 100000000 XOR area >= 3000000 ORDER BY name Submit SQL Restore default result

#### **String operations**

12. Cheat mode

Write a query selecting the country names that are the same as that country's capital, sorted by name.

SELECT name FROM world WHERE name = capital ORDER BY name Restore default Submit SQL result

13. Cheat mode

Write a query selecting the country names and capital names where both names have the same length. E.g., "Greece" and "Athens" both have length 6. Exclude the cases from the previous question, i.e., those where the country name is the same as the capital, and sort the results by country name. Hint: search the Internet for a function to compute the length of a string.

SELECT name, capital FROM world WHERE LENGTH(name)=LENGTH(capital) AND NOT name = capital ORDER BY name	
	/
Submit SQL Restore default	
result	
resuit	

Write a query returning the country name with the least population, along with its population.

SELECT name, populat	ion FROM world ORDER BY population LIMIT	1		
Submit SQL	Restore default			
result				
				,

Write a query returning the name and population of the 5 countries with the greatest population (ordered by descending population, i.e., from greatest to least population).

SELECT name, popul	ation FROM world ORDER BY population DESC LIM	IT 5	
Submit SQL	Restore default		
result			

Write a query returning the name, population, and area of the 42nd country in alphabetical order.

SELECT name, population, area FROM world ORDER BY name LIMIT 1 OFFSET 41 Submit SQL Restore default result

Write a query returning the name and population of the 10 most *populous* countries among the 20 countries with the greatest *area*. Hint: use a subquery, and remember to give an alias to its result!

ELECT name,	population FROM	(SELECT name, population	FROM world ORDER B	BY area DESC LIMIT	20) AS target 0	RDER BY population	n DESC LIMIT 10	
								,
Submit S	SQL	Restore default						
result								

#### **Computation and aggregation**

# 18.

Write a query returning the name and *population density* of the Asian countries. The *population density* is the population divided by the area. Order the results by name.

SELECT name, popula	ion/area AS 'population_density' FROM world WHERE continent = 'Asia' ORDER BY name	
Submit SQL	Restore default	/
result		
		//

Write a query returning the name of Asian countries with a new column dense containing "yes" for countries with density at least 100 inhabitants per square kilometer, and "no" otherwise. Sort the results to have first the dense countries, then the non-dense countries, and then order the countries in each group by name. Hint: use UNION.

SELECT name, 'yes' AS 'dense' FROM world WHERE continent = 'Asia' AND population/area >= 100 UNION SELECT name, 'no' AS 'dense' FROM world WHERE continent = 'Asia' AND population/area < 100 Submit SQL Restore default result

Write a query returning the number of countries in the database. Hint: use  $\texttt{COUNT}\,(*)$  .

SELECT COUNT(name) FROM world Submit SQL Restore default result

Write a query returning the total world population. Hint: use SUM.

SELECT SUM(population)	FROM world		
			/
Submit SQL	Restore default		
result			
			//

Write a query returning the continent names and, for each continent, the number of countries in that continent. Hint: use aggregation. In this question and the next ones, unless otherwise stated, order the results by continent.

SELECT continent,	COUNT(name) FROM world GROUP BY continent ORDER BY continent	
		/
Submit SQL	Restore default	
result		
		//

Write a query returning the continent names and, for each continent, the total population and total area of the countries in that continent.

SELECT continent,	SUM(population), SUM(area) FROM world GROUP BY continent ORDER BY con	tinent
Submit SQL	Restore default	
result		

Show the name and total population of continents having total population at least 100 million.

SELECT continent, SUM (population) FROM world GROUP BY continent HAVING SUM (population) > 100000000 ORDER BY continent Submit SQL Restore default result

Show the name of continents and the number of countries in that continent that have population at least 1 million.

SELECT continent, COUNT(name) FROM world WHERE population >= 1000000 GROUP BY continent Submit SQL Restore default result

Compute the average population of the world's countries. Unfortunately, the AVERAGE command is unavailable, but find a way to do it nevertheless.;)

SELECT SUM(population)/COUNT(name) AS average FROM world Submit SQL Restore default result

#### **Advanced queries**

## 27.

Compute, for every continent name, its population density (i.e., the continent's population divided by the continent's area), and the average of the population densities of its countries. Do you understand why the two values are different?

SELECT \* FROM (SELECT continent, SUM(population)/SUM(area) as density1 FROM world GROUP BY continent) as group1 LEFT JOIN (SELECT continent, group2.density2 as density3 FROM (SELECT continent, SUM(population/area)/COUNT(\*) AS density2 FROM world GROUP BY continent) as group2) as group3 ON group1.continent = group3.continent Submit SQL Restore default result

Return a table with a column alpha containing the first letter of country names (ordered alphabetically) and a column total with the total population of countries whose name starts with that letter. Hint: use SUBSTR as in the example.

SELECT SUBSTR(name, 1,	1) AS alpha, SUM(population) AS total FRO	OM world GROUP BY alpha	
Submit SQL	Restore default		/
result			
			//

Compute, for every continent name, its country having the largest area, and its country having the greatest population. Order the result by continent. Hint: this is a complicated task, use subqueries and/or join the world table with itself. Note: It is normal that Kazakhstan appears as a country in Europe.

SELECT continent, name AS area_name FROM world ORDER BY area DESC	
Submit SQL Restore default	
result	
	//

Compute, for every continent name, a count of how many countries in the continent are strictly more populous that the continent's largest country (by area). Order the results by continent, and do not omit results where the count is zero.

SELECT continent, SUM(group2.density2)/COUNT(group2.name) FROM (SELECT continent, name, population/area AS density2 FROM world) as group2 Group by continent Submit SQL Restore default result Cheat mode

#### Acknowledgements

This exercise is inspired by the exercises <u>SELECT\_from\_WORLD\_Tutorial</u>, <u>SELECT\_within\_SELECT\_Tutorial</u>, and <u>SUM\_and\_COUNT</u>. If you reach the end of this exercise and still have time, you can complete the questions in these three tutorials that are not the same as the ones covered here.

Thanks to Andrew Cumming for his great work in maintaining SQLZoo!

Retrieved from "http:///sqlzoo.net/w/index.php?title=User:A3nm/SD202\_2021\_ex1&oldid=39854"

This page was last edited on 24 September 2021, at 10:02.