* Technologies used: SQLite
* Create database.
* Import all 4 csv files into new tables.

------------ Replace '' empty strings with NULL ----------

UPDATE continent\_map

SET

country\_code = CASE country\_code WHEN '' THEN NULL ELSE country\_code END,

continent\_code = CASE continent\_code WHEN '' THEN NULL ELSE continent\_code END;

1. Data Integrity Checking & Cleanup

* Alphabetically list all of the country codes in the continent\_map table that appear more than once. Display any values where country\_code is null as country\_code = "FOO" and make this row appear first in the list, even though it should alphabetically sort to the middle. Provide the results of this query as your answer.
* For all countries that have multiple rows in the continent\_map table, delete all multiple records leaving only the 1 record per country. The record that you keep should be the first one when sorted by the continent\_code alphabetically ascending. Provide the query/ies and explanation of step(s) that you follow to delete these records.

----------- Select Statement To Pull Up Duplicate Country Codes, FOO on top ----------

SELECT COALESCE(country\_code, 'FOO')

FROM continent\_map

GROUP BY country\_code

HAVING COUNT(\*) > 1

ORDER BY country\_code;

A screenshot of a computer

Description automatically generated with medium confidence

------------ Create a temporary table with a new column ID as a row\_number on the table after order by contry\_code, continent\_code ------------

CREATE TABLE t1 AS

SELECT ROW\_NUMBER() OVER (ORDER BY country\_code, continent\_code ASC) AS ID, country\_code, continent\_code

FROM continent\_map;

CREATE TABLE t2 AS

SELECT MIN(ID) AS ID

FROM t1

GROUP BY country\_code;

------------ Delete the rows that dont have a min ID number after group by country\_code ------------

Delete From t1 where ID NOT IN(select ID from t2) ;

------------ Reset continent\_map table ------------

Delete From continent\_map;

------------ Refill continent\_map from temp\_table ------------

insert into continent\_map

select country\_code, continent\_code from t1;

------------ drop temporary tables ------------

DROP TABLE t1;

DROP TABLE t2;

2. List the countries ranked 10-12 in each continent by the percent of year-over-year growth descending from 2011 to 2012.

The percent of growth should be calculated as: ((2012 gdp - 2011 gdp) / 2011 gdp)

The list should include the columns:

* rank
* continent\_name
* country\_code
* country\_name
* growth\_percent

------------ in order to do this one I first created a VIEW in my data base which joined the per\_capita table to all other tables appropriately. the join looks like this -----------

CREATE VIEW gdp\_join AS

SELECT

per\_capita.country\_code AS country\_code,

countries.country\_name AS country\_name,

continents.continent\_code AS continent\_code,

continents.continent\_name AS continent\_name,

per\_capita.year AS year,

per\_capita.gdp\_per\_capita AS gdp\_per\_capita

FROM

continent\_map

JOIN countries ON continent\_map.country\_code = countries.country\_code

JOIN continents ON continent\_map.continent\_code = continents.continent\_code

JOIN per\_capita ON continent\_map.country\_code = per\_capita.country\_code;

---------- Using this view as my dataset to refer to, I'll answer the GDP question with this query ----------

CREATE TABLE GDP\_Growth\_Rank AS

SELECT

t1.continent\_name,

t1.country\_code,

t1.country\_name,

ROUND(((t2.gdp\_2012 - t1.gdp\_2011) / t1.gdp\_2011) \* 100, 2) || '%' AS growth\_percent,

(

SELECT COUNT(\*) + 1

FROM (

SELECT

t1\_inner.country\_code,

((t2\_inner.gdp\_2012 - t1\_inner.gdp\_2011) / t1\_inner.gdp\_2011) AS growth

FROM

(

SELECT

continent\_name,

country\_code,

country\_name,

gdp\_per\_capita AS gdp\_2011

FROM

gdp\_join

WHERE

year = 2011

) t1\_inner

INNER JOIN

(

SELECT DISTINCT

country\_code,

gdp\_per\_capita AS gdp\_2012

FROM

gdp\_join

WHERE

year = 2012

) t2\_inner ON t1\_inner.country\_code = t2\_inner.country\_code

WHERE

t1\_inner.continent\_name = t1.continent\_name

ORDER BY

growth DESC

) AS ranks

WHERE ranks.growth > ((t2.gdp\_2012 - t1.gdp\_2011) / t1.gdp\_2011)

) AS drank

FROM

(

SELECT

continent\_name,

country\_code,

country\_name,

gdp\_per\_capita AS gdp\_2011

FROM

gdp\_join

WHERE

year = 2011

) t1

INNER JOIN

(

SELECT DISTINCT

country\_code,

gdp\_per\_capita AS gdp\_2012

FROM

gdp\_join

WHERE

year = 2012

) t2 ON t1.country\_code = t2.country\_code;

----------- for result ----------

SELECT \* FROM gdp\_growth\_rank

where drank > 9 and drank < 13

ORDER BY continent\_name;

A screenshot of a computer screen

Description automatically generated with medium confidence

3. For the year 2012, create a 3 column, 1 row report showing the percent share of gdp\_per\_capita for the following regions:

(i) Asia, (ii) Europe, (iii) the Rest of the World. Your result should look something like

| **Asia** | **Europe** | **Rest of World** |
| --- | --- | --- |
| 25.0% | 25.0% | 50.0% |

SELECT

ROUND(((SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012 AND continent\_name = 'Asia') / (SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012)) \* 100, 1) || '%' AS 'Asia',

ROUND(((SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012 AND continent\_name = 'Europe') / (SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012)) \* 100, 1) || '%' AS 'Europe',

ROUND(((SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012 AND continent\_name != 'Asia' AND continent\_name != 'Europe') / (SELECT SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2012)) \* 100, 1) || '%' AS 'Rest of World';

A picture containing text, screenshot, line, font

Description automatically generated

4.

a. What is the count of countries and sum of their related gdp\_per\_capita values for the year 2007 where the string 'an' (case insensitive) appears anywhere in the country name?

SELECT COUNT(\*), SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2007 AND country\_name LIKE '%an%';

A screenshot of a computer code

Description automatically generated with low confidence

b. Repeat question 4a, but this time make the query case sensitive.

SELECT COUNT(\*), SUM(gdp\_per\_capita)

FROM gdp\_join

WHERE year = 2007 AND NOT (country\_name LIKE '%an%' COLLATE NOCASE);

A screenshot of a computer

Description automatically generated with medium confidence

5. Find the sum of gpd\_per\_capita by year and the count of countries for each year that have non-null gdp\_per\_capita where (i) the year is before 2012 and (ii) the country has a null gdp\_per\_capita in 2012. Your result should have the columns:

* year
* country\_count
* total

SELECT year, COUNT(DISTINCT country\_name) AS country\_count, SUM(gdp\_per\_capita) AS total

FROM gdp\_join

WHERE year < 2012 AND country\_code IN (

SELECT country\_code

FROM gdp\_join

WHERE year = 2012 AND gdp\_per\_capita IS NULL

)

AND gdp\_per\_capita IS NOT NULL

GROUP BY year;

A screenshot of a computer

Description automatically generated with medium confidence

6. All in a single query, execute all the steps below and provide the results as your final answer:

a. creates a single list of all per\_capita records for year 2009 that includes columns:

* continent\_name
* country\_code
* country\_name
* gdp\_per\_capita

b. order this list by:

* continent\_name ascending
* characters 2 through 4 (inclusive) of the country\_name descending

SELECT continent\_name, country\_code, country\_name, gdp\_per\_capita

FROM gdp\_join

where year = 2009

order by continent\_name asc, substring(country\_name,2,3) desc

A screenshot of a computer

Description automatically generated

c. creates a running total of gdp\_per\_capita by continent\_name

d. return only the first record from the ordered list for which each continent's running total of gdp\_per\_capita meets or exceeds $70,000.00 with the following columns:

* continent\_name
* country\_code
* country\_name
* gdp\_per\_capita
* running\_total

A screenshot of a computer

Description automatically generated with medium confidence

7. Find the country with the highest average gdp\_per\_capita for each continent for all years. Now compare your list to the following data set. Please describe all mistakes that you can find with the data set below. Include any code that you use to help detect these mistakes.

| **rank** | **continent\_name** | **country\_code** | **country\_name** | **avg\_gdp\_per\_capita** |
| --- | --- | --- | --- | --- |
| 1 | Africa | SYC | Seychelles | $11,348.66 |
| 1 | Asia | KWT | Kuwait | $43,192.49 |
| 1 | Europe | MCO | Monaco | $152,936.10 |
| 1 | North America | BMU | Bermuda | $83,788.48 |
| 1 | Oceania | AUS | Australia | $47,070.39 |
| 1 | South America | CHL | Chile | $10,781.71 |

WITH average\_gdp AS (

SELECT continent\_name, country\_code, country\_name, AVG(gdp\_per\_capita) AS avg\_gdp\_per\_capita

FROM gdp\_join

GROUP BY continent\_name, country\_name)

SELECT continent\_name, country\_code, country\_name, avg\_gdp\_per\_capita

FROM average\_gdp

WHERE (continent\_name, avg\_gdp\_per\_capita) IN (

SELECT continent\_name, MAX(avg\_gdp\_per\_capita)

FROM average\_gdp

GROUP BY continent\_name)

ORDER BY continent\_name ASC;

A screenshot of a computer code

Description automatically generated with medium confidence