**Part 1.**

Query 1:

SELECT

country\_name,

survey\_acronym,

survey\_year

FROM df

WHERE region\_code ='LAC' and survey\_coverage = 'national'

GROUP BY country\_name, survey\_acronym, survey\_year;

|country\_name │ survey\_acronym │ survey\_year │

│ varchar │ varchar double │

├─────────────────────┼──────────

│ Bolivia │ EH │ 2017.0 │

│ Brazil │ PNAD │ 1983.0 │

│ Brazil │ PNADC-E1 │ 2016.0 │

│ Brazil │ PNADC-E1 │ 2017.0 │

│ Brazil │ PNADC-E1 │ 2018.0 │

382 rows

Query 2:

SELECT

country\_name,

AVG(reporting\_pce) AS min\_avg\_reporting\_pce

FROM df

WHERE region\_code ='SSA'

GROUP BY country\_name

HAVING AVG(reporting\_pce) <= ALL

(SELECT avg\_reporting\_pce FROM

(SELECT

country\_name,

AVG(reporting\_pce) AS avg\_reporting\_pce

FROM df

WHERE region\_code ='SSA'

GROUP BY country\_name)

WHERE avg\_reporting\_pce IS NOT NULL);

Output:

┌────────────────────────────┐

│ country\_name │ min\_avg\_reporting\_pce │

│ varchar │ double │

├────────── ─┼─ ──────────────┤

│ Congo, Dem. Rep.│ 305.26031975 │

└──────────────────┴─────────┘

**Part 2.**

Columns “region\_code”, “survey\_coverage” and “reporting\_gdp

” have NULLs

1. For column “region\_code” we use imputation based on other columns, because each value in column “country\_code” corresponds to one value in column “region\_code”:

Every country is located in one region, in rows with NULL in “region\_code” we can take “country\_code” (it doesn’t have NULL), and take it region\_code from another row with ssame country.

But after that we see, that Gabon and Guyana are still without region\_code (because all rows with this countries have NULLs in region\_code). But we can use Google and find, that this countries located in SSA and LAC respectively, and input it manually.

1. For column “survey\_coverage” we use impattiom based on most common value in this column. We see, that value “national” meets mush more often than others.
2. For column “reporting\_gdp” we use imputation based on the same column. Let’s take country\_code from rows with NULL in this columns and calculate mean value of reporting\_gdp for this country. Some countries have no value at all, for them use mean value of they regions.

After these imputation in 1st query we got the result 20 rows more (because we filled in NULLs in columns “region\_code” and “survey\_coverage”).

Result of 2nd query didn’t change.

**Part 3.**

1. Functional dependence:

country\_code → country\_name

country\_code →region\_code

|  |
| --- |
| Main |
| **record\_id** |
| **country\_code** |
| **reporting\_year** |
| **reporting\_level** |
| **survey\_coverage** |
| **survey\_year** |
| **…** |
| **reporting\_gdp** |
| **…** |

|  |
| --- |
| Countries |
| country\_code |
| country\_name |

|  |
| --- |
| Regions |
| country\_code |
| region\_code |

1. Queries
   1. SELECT

C.country\_name,

survey\_acronym,

survey\_year

FROM df, regions AS R, countries AS C

WHERE df.country\_code = R.country\_code AND

df.country\_code = C.country\_code AND

R.region\_code ='LAC' AND

survey\_coverage = 'national'

GROUP BY

country\_name,

survey\_acronym,

survey\_year

ORDER BY C.country\_name, survey\_year;

* 1. SELECT

C.country\_name,

AVG(reporting\_pce) AS min\_avg\_reporting\_pce

FROM df, regions AS R, countries AS C

WHERE df.country\_code = R.country\_code AND

R.region\_code ='SSA' AND

df.country\_code = C.country\_code

GROUP BY C.country\_name

HAVING AVG(reporting\_pce) <= ALL

(SELECT avg\_reporting\_pce FROM

(SELECT

df.country\_code,

AVG(reporting\_pce) AS avg\_reporting\_pce

FROM df, regions AS R

WHERE df.country\_code = R.country\_code AND

R.region\_code ='SSA'

GROUP BY df.country\_code)

WHERE avg\_reporting\_pce IS NOT NULL)