Contents

MySQL Questions

Import the *world* database from *world.sql* to MySQL and write queries to satisfy the following.

Write only the exact MySQL command for each question into the appropriate file.

Question C (MySQLQC.txt)

Show the *Name* and first 10 characters of GovernmentForm (as *Government*) of countries where “German” is an official language.

The results should be sorted alphabetically by Name.

```
+---------------+------------+
| Name          | Government |
+---------------+------------+
| Austria       | Federal Re |
| Belgium       | Constituti |
| Germany       | Federal Re |
| Liechtenstein | Constituti |
| Luxembourg    | Constituti |
| Switzerland   | Federation |
+---------------+------------+
6 rows in set (0.02 sec)
```

Figure 1 Example of output required for this question

Question D (MySQLQD.txt)

Show the *PersonID*, *Personname*, *age* and a column entitled *Visa* which has the following values:

- Visa-1
  If the person is under 20

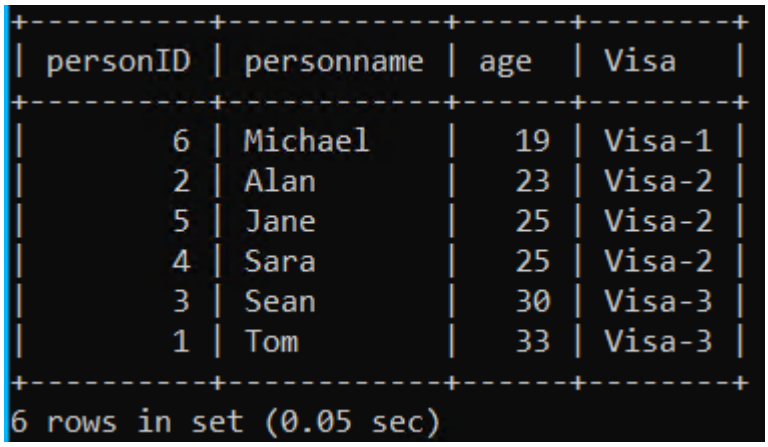
- Visa-2
  If the person is between 20 and 29 inclusive

- Visa-3
  If the person is between 30 and 39 inclusive

- Visa-4
  If the person is between 40 and 69 inclusive

- Visa-5
  If the person is 70 or older.

The results should be sorted alphabetically by *Visa*, and within that alphabetically by *Personname*.



personID	personname	age	Visa
6	Michael	19	Visa-1
2	Alan	23	Visa-2
5	Jane	25	Visa-2
4	Sara	25	Visa-2
3	Sean	30	Visa-3
1	Tom	33	Visa-3

6 rows in set (0.05 sec)

Figure 2 Example of output required for this question.

Question G (MySQLQG.txt)

For all cities visited by “Sara” show:

- The name of the city
- The Arrival Date in the city
- The name of the country the city is in

The results should be sorted alphabetically by city name.

```
+--------------+-------------+----------------+
| name         | dateArrived | name           |
+--------------+-------------+----------------+
| Jaunpur      | 1998-06-20  | India          |
| Saint Helier | 1999-05-20  | United Kingdom |
| Zürich       | 1999-01-20  | Switzerland    |
+--------------+-------------+----------------+
3 rows in set (0.02 sec)
```

Figure 3 Example of output required for this question.

Question H (MySQLQH.txt)

Show the *Continent*, and the *Name* and *Population* of the country with the biggest population in each continent.

NOTE: Only include countries where the population is greater than 0.

The results should be sorted from largest to smallest population, and within that alphabetically by Continent.

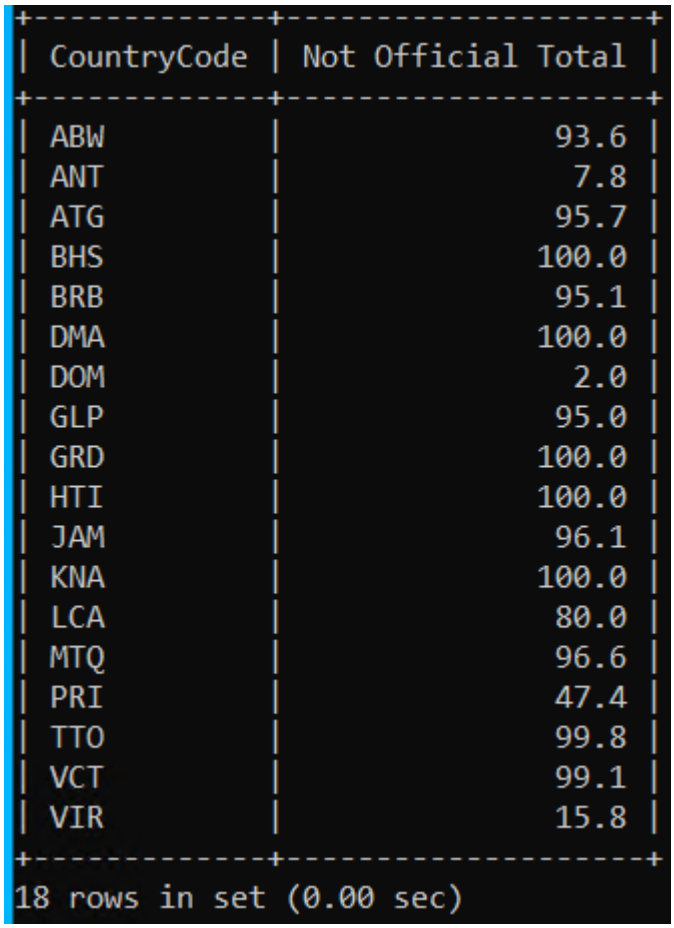
```
+---------------+--------------------+------------+
| Continent     | Name               | Population |
+---------------+--------------------+------------+
| Asia          | China              | 1277558000 |
| North America | United States      |  278357000 |
| South America | Brazil             |  170115000 |
| Europe        | Russian Federation |  146934000 |
| Africa        | Nigeria            |  111506000 |
| Oceania       | Australia          |   18886000 |
+---------------+--------------------+------------+
6 rows in set (0.11 sec)
```

Figure 4 Example of output required for this question.

Question K (MySQLQK.txt)

Show the *CountryCode* and the percentage of people who speak the non-official languages (as *Not Official Total*) of all countries in the "Caribbean".

The results should be sorted alphabetically by *CountryCode*.



CountryCode	Not Official Total
ABW	93.6
ANT	7.8
ATG	95.7
BHS	100.0
BRB	95.1
DMA	100.0
DOM	2.0
GLP	95.0
GRD	100.0
HTI	100.0
JAM	96.1
KNA	100.0
LCA	80.0
MTQ	96.6
PRI	47.4
TTO	99.8
VCT	99.1
VIR	15.8

18 rows in set (0.00 sec)

Figure 5 Example of output required for this question.

Question L (MySQLQL.txt)

Show *Name*, *Population* and *PersonName* of all cities visited by people, where the city population is greater than the maximum population of “Polynesia”.

The results should be sorted alphabetically by name.

```
+------------+------------+------------+
| name       | population | personname |
+------------+------------+------------+
| Muntinlupa |     379310 | Tom        |
| Nagoya     |    2154376 | Michael    |
| Sydney     |    3276207 | Tom        |
| Zürich     |     336800 | Sara       |
+------------+------------+------------+
4 rows in set (0.00 sec)
```

Figure 6 Example of output required for this question.

Neo4j Questions

Import *personDB.txt* into Neo4j as follows:

```
cd C:\Users\appDB2022\Documents\neo4j-community-4.4.3-windows\neo4j-community-4.4.3\bin

type path_to_personDB.txt | cypher-shell.bat -u neo4j -p neo4j --format plain
```

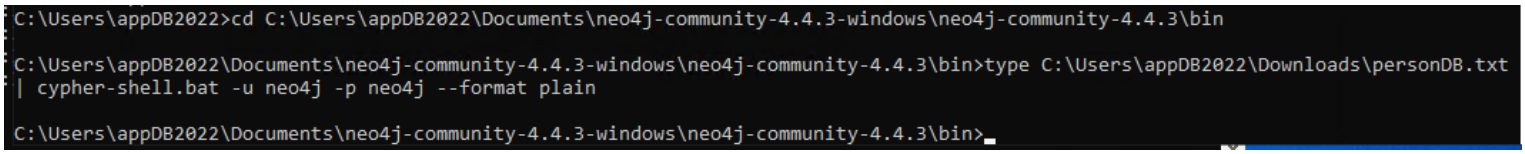


Figure 7 Import Neo4j database

Write only the exact MongoDB command for each question into the appropriate file.

Question C (Neo4jQC.txt)

Return the names of hobbies (as *Hobby*) and the number of people who have that hobby (as *People*).

The results should be sorted in increasing People order and within that by Hobby.

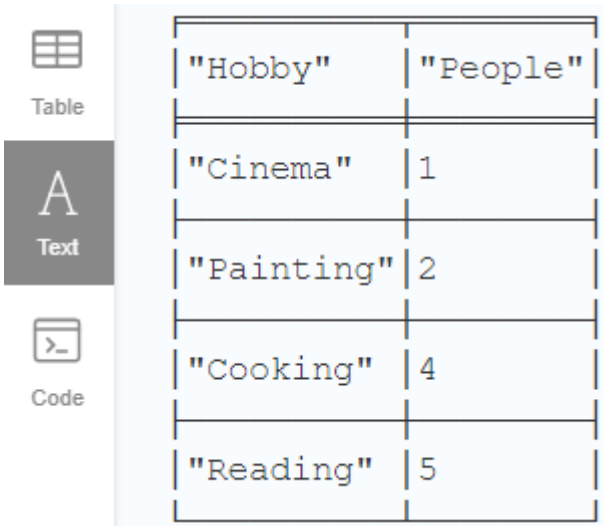


Figure 8 Example of output required for this question.

Question D (Neo4jQD.txt)

Return the number of people who have a salary (as *People_on_a_Salary*).

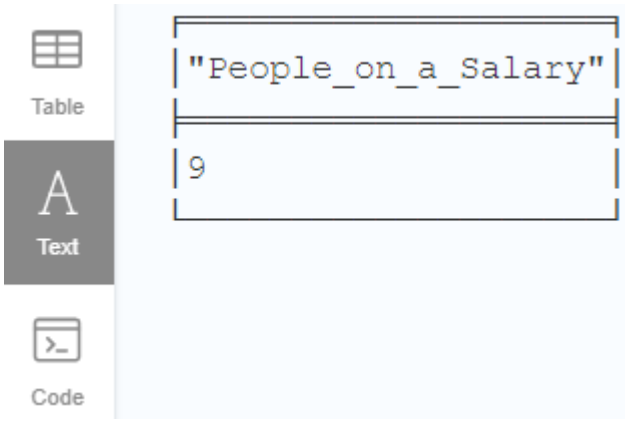


Figure 9 Example of output required for this question.

Question G (Neo4jQG.txt)

Return the names (as *Names*) and ages (as *Ages*) of people who have the same Hobby as *Barbara Smith*'s maternal grandmother.

Results should be sorted alphabetically by *name*, and within that from youngest to oldest.

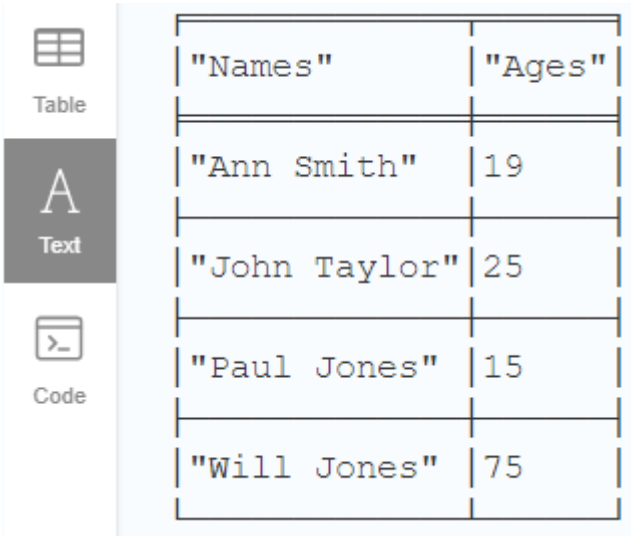


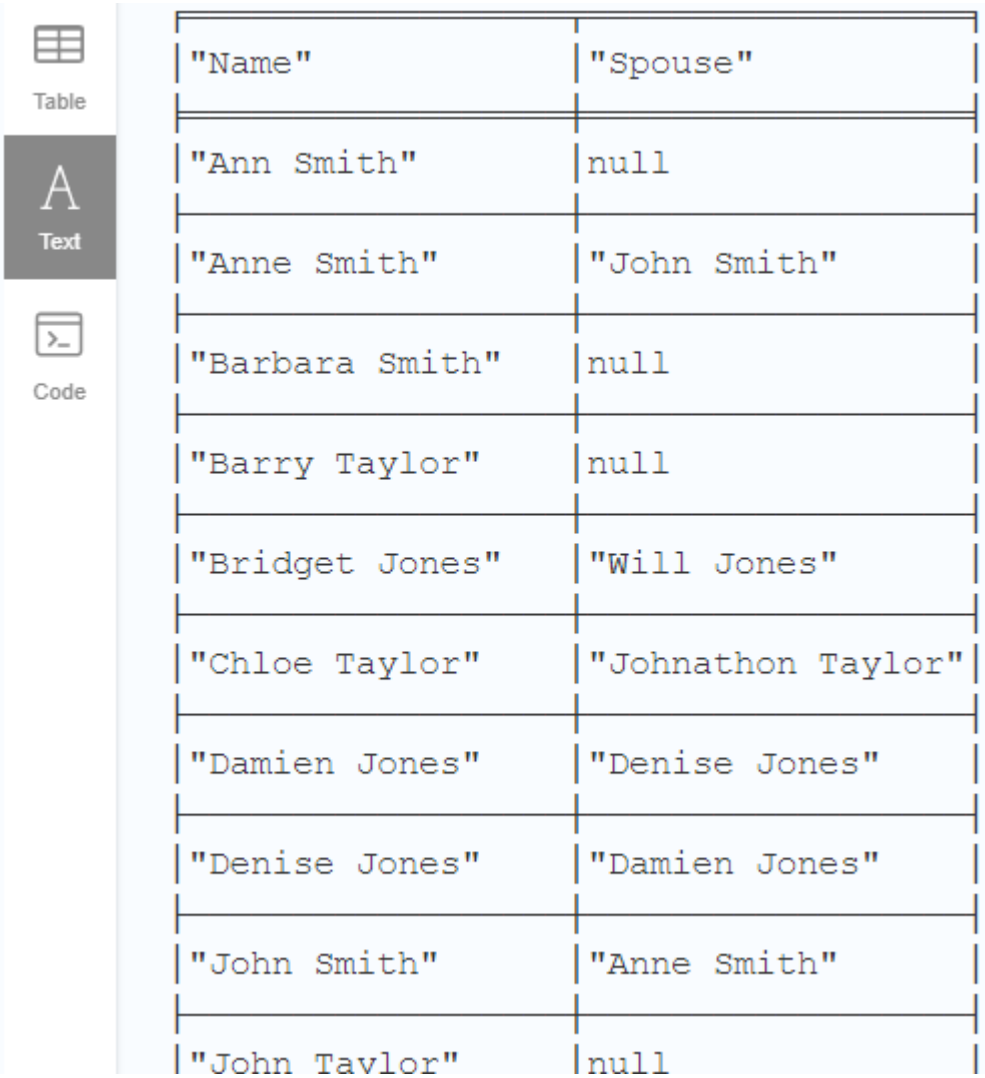
Figure 10 Example of output required for this question.

Question H (Neo4jQH.txt)

Return the name of each person (as *Name*) and the person they have a MARRIED_TO relationship with (as *Spouse*).

If someone does not have a MARRIED_TO relationship their *Spouse* should be *null*.

Results should be returned in alphabetical *Name* order, followed by alphabetical *Spouse* order.



"Name"	"Spouse"
"Ann Smith"	null
"Anne Smith"	"John Smith"
"Barbara Smith"	null
"Barry Taylor"	null
"Bridget Jones"	"Will Jones"
"Chloe Taylor"	"Johnathon Taylor"
"Damien Jones"	"Denise Jones"
"Denise Jones"	"Damien Jones"
"John Smith"	"Anne Smith"
"John Taylor"	null

Figure 11 Example of output required for this question.

Question K(Neo4jQK.txt)

Return the number of people under 20 (as *Under_20s*) who have the hobby *Reading*.

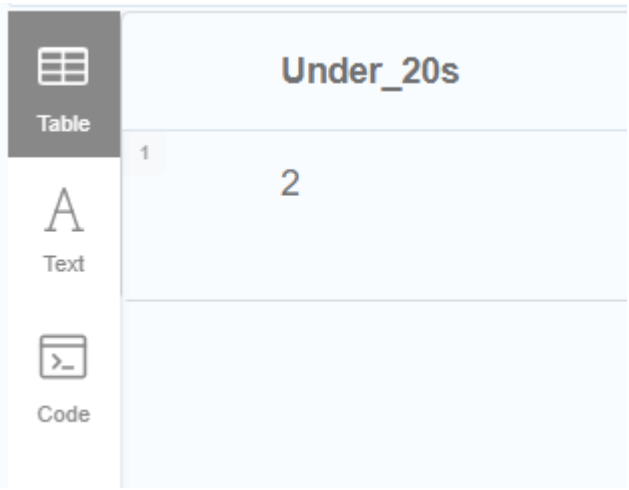


Figure 12 Example of output required for this question.

Question L (Neo4jQL.txt)

Return the list of salaries of people who are less than 50,000 (as *Salaries_LT_50k*). The salaries should be rounded up or down to the nearest whole number.

E.g. 100.5 becomes 101, 100.4 becomes 100.

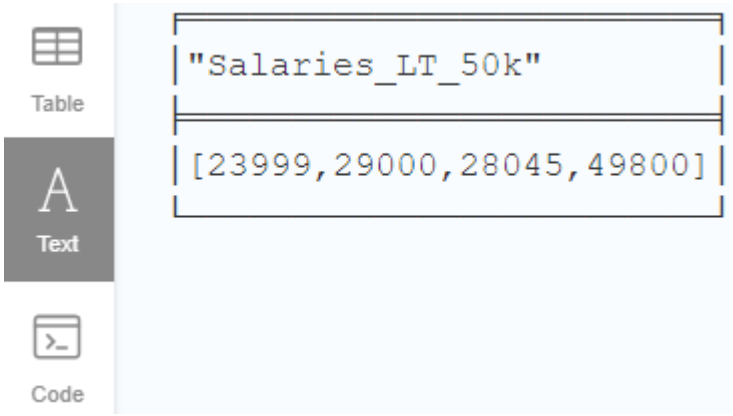


Figure 13 Example of output required for this question.