Contents

MySQL Questions

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

Question A (MySQLQA.txt)

Show the *personname* and a column entitled *Countries_Visited* that shows the number of individual countries visited by that person.

The results should be sorted alphabetically by *personname*.

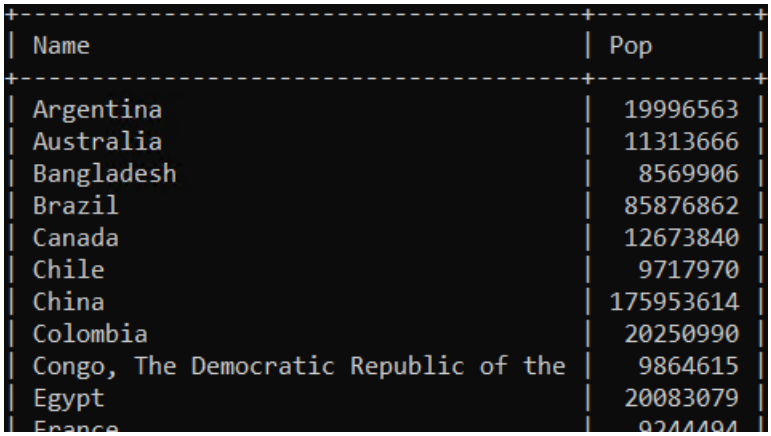
```
+---------------------+-------------------+
| personname          | Countries_Visited |
+---------------------+-------------------+
| Byrne, Alan         |                 5 |
| Dr. Sean Murphy     |                 3 |
| Mannion, Mr. Michael |                3 |
| Mr. Tom Kelleher    |                 6 |
| MS. Sara Gallagher  |                 3 |
+---------------------+-------------------+
5 rows in set (0.00 sec)
```

Figure 1 Example of output required for this question

Question B (MySQLQB.txt)

Show the *Name* and a column entitled *Pop* that shows the total population of cities in that country, only for countries whose cities population is greater than the average cities population of all countries.

The results should be sorted alphabetically by *Name*.



Name	Pop
Argentina	19996563
Australia	11313666
Bangladesh	8569906
Brazil	85876862
Canada	12673840
Chile	9717970
China	175953614
Colombia	20250990
Congo, The Democratic Republic of the	9864615
Egypt	20083079
France	9244494

Figure 2 Example of output required for this question.

Question C (MySQLQC.txt)

Show the *Name*, *Continent* and *LifeExpectancy* of non-European countries whose *LifeExpectancy* is greater than the average European *LifeExpectancy*.

Name	Continent	LifeExpectancy
Kuwait	Asia	76.1
Taiwan	Asia	76.4
Cyprus	Asia	76.7
Jordan	Asia	77.4
Israel	Asia	78.6
Hong Kong	Asia	79.5
Singapore	Asia	80.1
Japan	Asia	80.7
Macao	Asia	81.6
Jamaica	North America	75.2
Virgin Islands, British	North America	75.4
Panama	North America	75.5
Puerto Rico	North America	75.6
Costa Rica	North America	75.8
Anguilla	North America	76.1
Cuba	North America	76.2
Bermuda	North America	76.9
Guadeloupe	North America	77.0
United States	North America	77.1
Saint Pierre and Miquelon	North America	77.6
Montserrat	North America	78.0
Virgin Islands, U.S.	North America	78.1
Martinique	North America	78.3
Aruba	North America	78.4
Cayman Islands	North America	78.9
Canada	North America	79.4
Libyan Arab Jamahiriya	Africa	75.5
Saint Helena	Africa	76.8

Figure 3 Example of output required for this question.

Question D (MySQLQD.txt)

Show the *Name*, *District*, *Population* (which should be formatted with commas), and a column entitled *CityType* that contains:

- *Small* if the city's population is between 0 and 49,999
- *Big* if the city's population is between 50,000 and 99,999
- *Huge* if the city's population is between 100,000 and 499,999
- *Mega* if the city's population is greater than 499,999.

The results should be sorted alphabetically by *CityType*, and within that alphabetically by *Name*.

Name	District	Population	CityType
Águas Lindas de Goiás	Goiás	89,200	Big
Angra dos Reis	Rio de Janeiro	96,864	Big
Araguari	Minas Gerais	98,399	Big
Bacabal	Maranhão	93,121	Big
Barra do Piraí	Rio de Janeiro	89,388	Big
Bento Gonçalves	Rio Grande do Sul	89,254	Big
Birigui	São Paulo	94,685	Big
Cametá	Pará	92,779	Big
Campo Largo	Paraná	91,203	Big
Conselheiro Lafaiete	Minas Gerais	97,507	Big
Coronel Fabriciano	Minas Gerais	95,933	Big
Corumbá	Mato Grosso do Sul	90,111	Big
Crato	Ceará	98,965	Big
Eunápolis	Bahia	96,610	Big
Guaíba	Rio Grande do Sul	92,224	Big
Ituiutaba	Minas Gerais	90,507	Big
Jacobina	Bahia	96,131	Big
Ji-Paraná	Rondônia	93,346	Big
Ourinhos	São Paulo	96,291	Big
Palhoça	Santa Catarina	89,465	Big
Parnamirim	Rio Grande do Norte	96,210	Big
Passos	Minas Gerais	98,570	Big
Patos	Paraíba	90,519	Big
Paulo Afonso	Bahia	97,291	Big
Pinhais	Paraná	98,198	Big
Poá	São Paulo	89,236	Big
Salto	São Paulo	96,348	Big
Santana do Livramento	Rio Grande do Sul	91,779	Big
São José de Ribamar	Maranhão	98,318	Big
São Lourenço da Mata	Pernambuco	91,999	Big
Sertãozinho	São Paulo	98,140	Big
Tatuí	São Paulo	93,897	Big
Toledo	Paraná	99,387	Big
Votorantim	São Paulo	91,777	Big
Abaetetuba	Pará	111,258	Huge
Alagoinhas	Bahia	126,820	Huge

Figure 4 Example of output required for this question.

Question E (MySQLQE.txt)

Show the *name*, *indepyear* and a column entitled *GovernmentForm* that contains the following:

- If the country was never independent – "n/a"
- If the country became independent less than 10 years ago, the word "New" should be prepended to the existing *GovernmentForm*.
- If the country became independent between 10 and 49 years ago, the word "Modern" should be prepended to the existing *GovernmentForm*.
- If the country became independent between 50 and 100 years ago, the word "Early" should be prepended to the existing *GovernmentForm*.
- If the country became independent more than 100 years ago, the word "Old" should be prepended prepended to the existing *GovernmentForm*.

- In addition, if the population of the country is more than 100 million, the word "Large" should be prepended to the existing *GovernmentForm*.

name	indepyear	GovernmentForm
Algeria	1962	Early Republic
Angola	1975	Modern Republic
Benin	1960	Early Republic
Botswana	1966	Early Republic
British Indian Ocean Territory	NULL	n/a
Burkina Faso	1960	Early Republic
Burundi	1962	Early Republic
Cameroon	1960	Early Republic
Cape Verde	1975	Modern Republic
Central African Republic	1960	Early Republic
Chad	1960	Early Republic
Comoros	1975	Modern Republic
Congo	1960	Early Republic
Congo, The Democratic Republic of the	1960	Early Republic
Côte d?Ivoire	1960	Early Republic
Djibouti	1977	Modern Republic
Egypt	1922	Old Republic
Equatorial Guinea	1968	Early Republic
Eritrea	1993	Modern Republic
Ethiopia	-1000	Old Republic
Gabon	1960	Early Republic
Gambia	1965	Early Republic
Ghana	1957	Early Republic
Guinea	1958	Early Republic
Guinea-Bissau	1974	Early Republic
Kenya	1963	Early Republic
Lesotho	1966	Early Constitutional Monarchy
Liberia	1847	Old Republic
Libyan Arab Jamahiriya	1951	Early Socialistic State
Madagascar	1960	Early Federal Republic
Malawi	1964	Early Republic
Mali	1960	Early Republic
Mauritania	1960	Early Republic
Mauritius	1968	Early Republic
Mayotte	NULL	n/a
Morocco	1956	Early Constitutional Monarchy
Mozambique	1975	Modern Republic
Namibia	1990	Modern Republic
Niger	1960	Early Republic
Nigeria	1960	Early Large Federal Republic

Figure 5 Example of output required for this question.

Question F (MySQLQF.txt)

Show the *ID*, *name* and a column entitled *latitude* which contains:

- The city's latitude followed by (N) if the latitude > 0
- The city's latitude followed by (S), without the minus symbol, if the latitude <= 0

and a column entitled *longitude* which contains:

- The city's longitude followed by (E) if the longitude > 0
- The city's longitude followed by (W) if the longitude <= 0.

Only cities whose latitude and longitude are not NULL should be shown in ascending *ID* order.

```
+------+----------------+------------+-------------+
| ID   | Name           | latitude   | longitude   |
+------+----------------+------------+-------------+
|   23 | Dordrecht      | 51.8133(N) | 4.6901(E)   |
|  130 | Sydney         | 33.8688(S) | 151.2093(E) |
|  443 | Guaíba         | 30.1091(S) | 51.3238(W)  |
| 1023 | Tanjung Pinang | 0.0000(S)  | 104.4665(E) |
| 1535 | Nagoya         | 35.1814(N) | 136.9064(E) |
| 1678 | Tama           | 35.6370(N) | 139.4463(E) |
| 2133 | Suzhou         | 31.2990(N) | 120.5853(E) |
| 3245 | Zürich         | 49.1805(N) | 2.1032(E)   |
+------+----------------+------------+-------------+
8 rows in set (0.00 sec)
```

Figure 6 Example of output required for this question.

Neo4j Questions

Write only the exact Neo4j/Cypher command for each question into the appropriate file.

Question A (Neo4jQA.txt)

Show the name (as *Name*) and eid (as *EID*) of each Employee who is a MEMBER_OF any team working on projects that the "Software" Department exclusively OVERSEES.

If a project is overseen by another Department, it should be ignored.

If employees are working on multiple projects, they should only be returned once.

Results should be returned in alphabetical *EID* order.

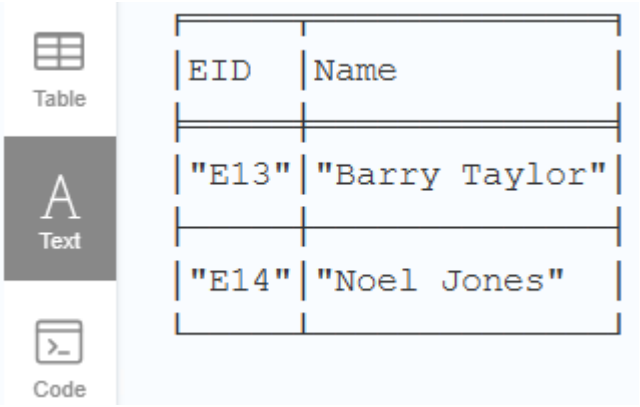


Figure 7 Example of output required for this question.

Question B (Neo4jQB.txt)

Show the Names (as *Senior_Manager*) and EIDs (as *EID*) of Employees who are IN_CHARGE_OF Departments in which the “Red” Team works.

Results should be returned in alphabetical *EID* order.

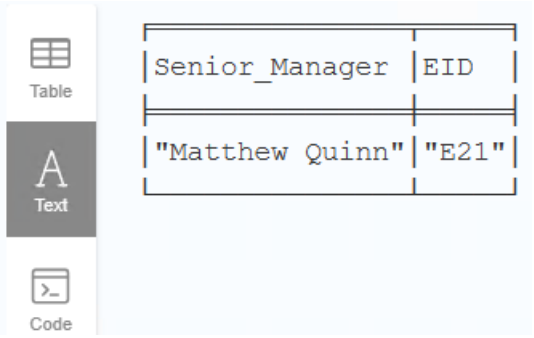


Figure 8 Example of output required for this question.

Question C (Neo4jQC.txt)

Show the name of each team (as *Team*), and the number of Employees that are a MEMBER_OF that team (as *Members*).

Results should be returned in alphabetical *Team* order.

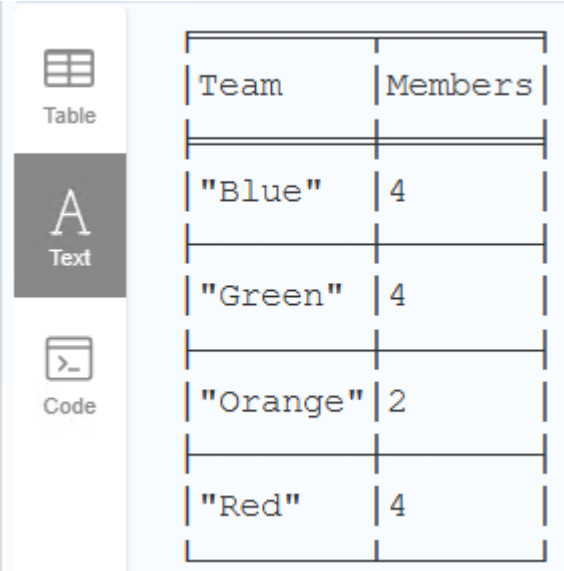


Figure 9 Example of output required for this question.

Question D (Neo4jQD.txt)

Show the Employee's eid (as *EID*), his/her name (as *Name*), and his/her salary (as *Salary*), only for Employees who are not also Managers, and whose salary is greater than the average salary for Employees who are not also Managers.

Results should be returned in alphabetical *EID* order.

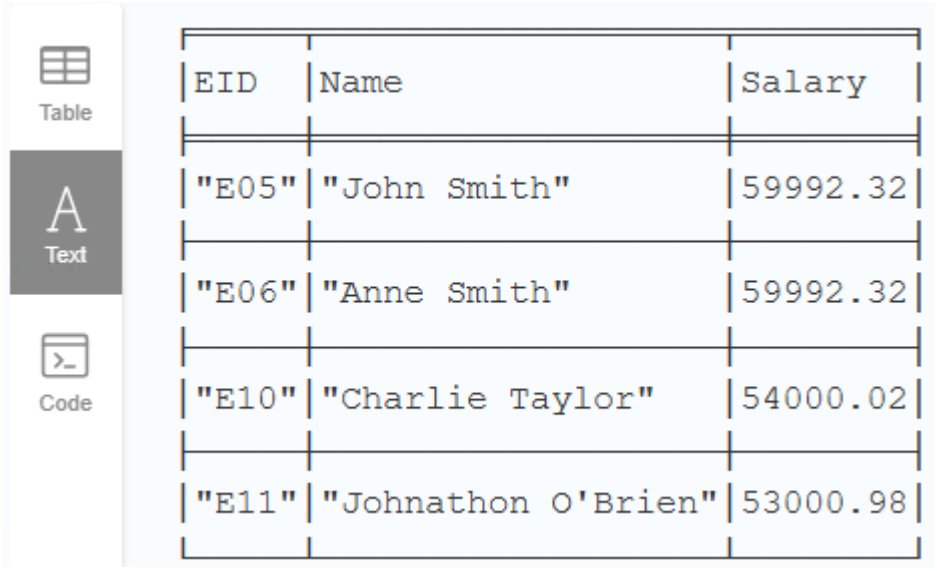


Figure 10 Example of output required for this question.

Question E (Neo4jQE.txt)

Show the name of the LEADER_OF the “Blue” team (as *Leader*), and the employees who are a MEMBER_OF the “Blue” team (as *Members*).

The *Members* should be sorted alphabetically by name.

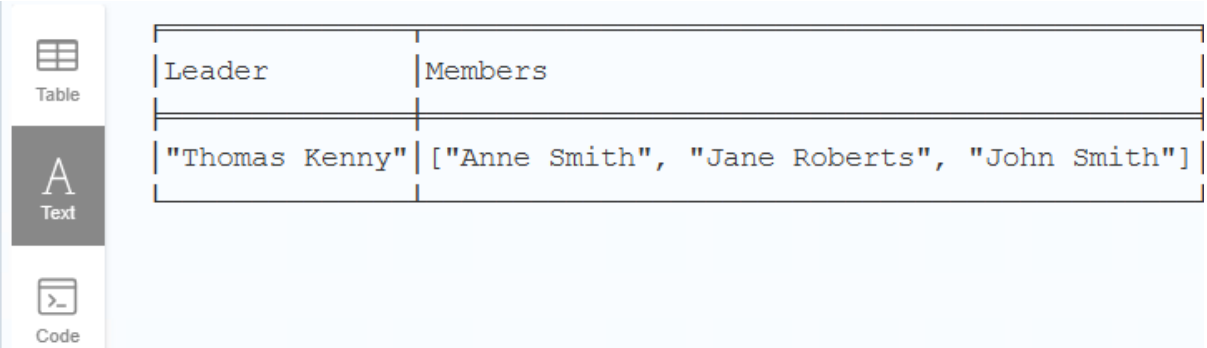


Figure 11 Example of output required for this question.

Question F (Neo4jQF.txt)

Show a column entitled *EmployeesWithSalary* that contains the number of Employees (not Managers) that have a salary.

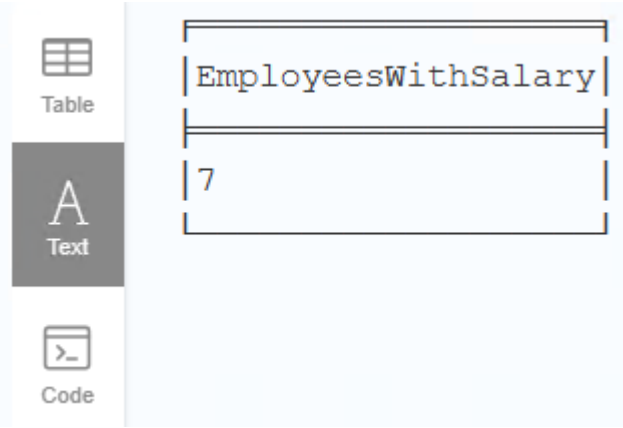


Figure 12 Example of output required for this question.