Project 3 report

Jiang Miao Fu Zhan

Purpose

In this project a query execution engine is designed. The functions of exercise 1 are actually methods of a Relation class. In a real database, we will have multiple instances of Relation, each representing one table. We modify exercise 1 to create the class and also add open(), getNext() and close() methods to it. And use the class to create and populate city and country tables. The data are obtained from the MySQL World Database sample. Finally use the class to find all cities whose population is more than 40% of the population of their entire country.
method open(), method getNext() and method close() please refer to the Relation part.

Comparator

The comparator class judge the condition of the join. It compare these records from the two relations whether meet specific condition. There are three conditions: equal(EQ), less than(LT), and greater than(GT).coefficient multiple default value is 1. In this project we want to find all cities whose population is more than 40% of the population of their entire country, at this time coefficient multiple is 0.4. And if attributes A is equals to attributes B, the condition will return true and then each tuple at this particular attributes do join (join operation refer to DBjoin).

Connection

The connection class define the connection of each relation and return the record of each relation. The record represents data by a key row(atrributes) and multiple value rows(value tuples). This class has two crucial methods select and union. Method 'select' Selects method to get required data from the relation, remove other useless data. Method 'Union' Unions this record with other records required. Moreover, method 'getonekey' can get the specific key form the relation by using its index number. Method 'getonevalue' can get the specific value form the relation by using its index number.

DBjoin

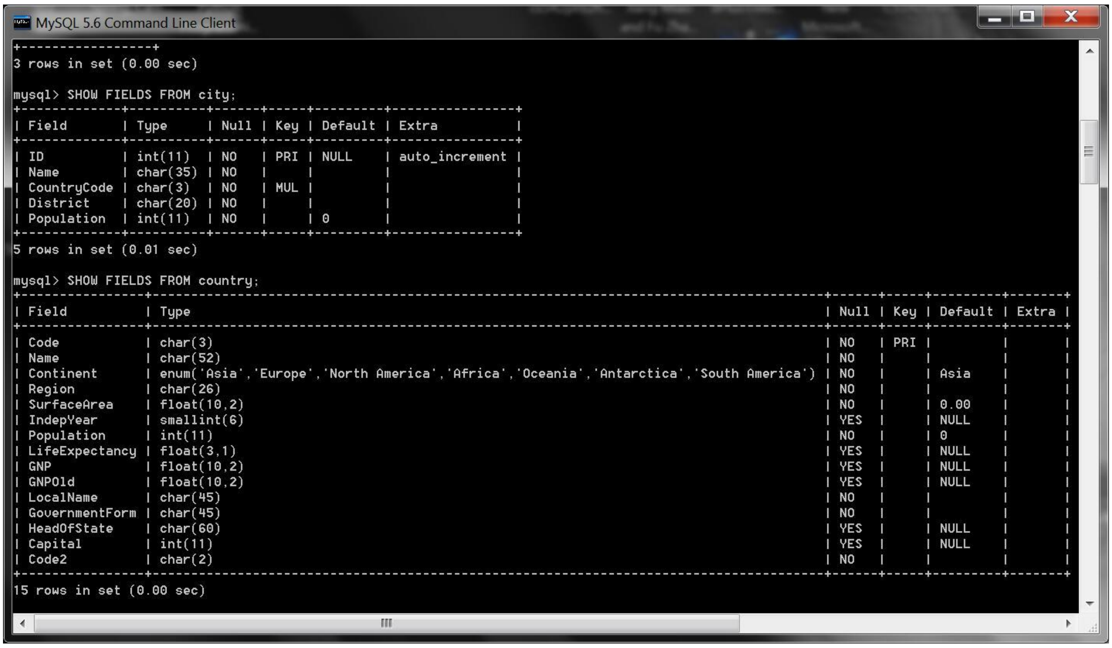
The DBjoin class aims to represent the join operation. When taking a join operation multiple conditions and two relation objects should be considered. The join condition method can act on more than one relationship. For example, in this project we need add two relationship:
(1) city.Coutrycode= coutry.code
(2) city.population> 40%*country.population.
And another important method is ‘connection implement’. It uses the left relation as the standard, check the condition and find all the required value in the right relation which satisfy the join conditions.

Relation

As described above, there are three main method in class relation.
Method open():Define a buffer block to operate, get the contents of db file line by line,the first line of each file is the key attribute .
Method getNext() : Get the next useful attribute record in the db file.
Method close() : close all the connection from the buffer which meant the original record will not be changed any more.

Test

In this project we need to find all cities whose population is more than 40% of the population of their entire country. Copy data to city.csv and country.csv(need read data from these files). To verify the result, original world.sql file was downloaded and can be used as a expecting result by the same query. Figure 1 shows we can get some query results in in database world by mysql. Figure 2 shows expecting result of this project. Figure 3 shows actual results obtained by the query execution engine. These results are the same, which means all cities whose population is more than 40% of the population of their entire country can be find correctly by this query engine.

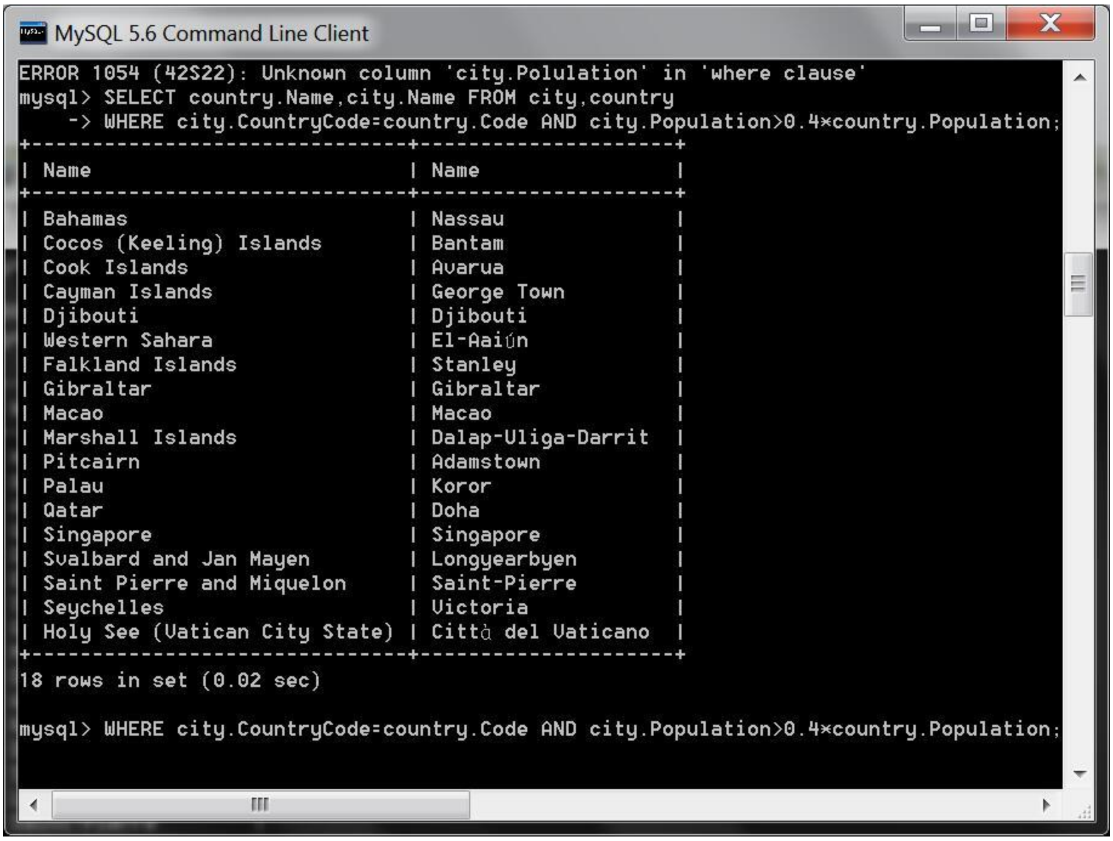


```
+-----------------+
3 rows in set (0.00 sec)

mysql> SHOW FIELDS FROM city;
+-------------+----------+------+-----+---------+----------------+
| Field       | Type     | Null | Key | Default | Extra          |
+-------------+----------+------+-----+---------+----------------+
| ID          | int(11)  | NO   | PRI | NULL    | auto_increment |
| Name        | char(35) | NO   |     |         |                |
| CountryCode | char(3)  | NO   | MUL |         |                |
| District    | char(20) | NO   |     |         |                |
| Population  | int(11)  | NO   |     | 0       |                |
+-------------+----------+------+-----+---------+----------------+
5 rows in set (0.01 sec)

mysql> SHOW FIELDS FROM country;
+----------------+---------------------------------------------------------------------------------------+------+-----+---------+-------+
| Field          | Type                                                                                  | Null | Key | Default | Extra |
+----------------+---------------------------------------------------------------------------------------+------+-----+---------+-------+
| Code           | char(3)                                                                               | NO   | PRI |         |       |
| Name           | char(52)                                                                              | NO   |     |         |       |
| Continent      | enum('Asia','Europe','North America','Africa','Oceania','Antarctica','South America') | NO   |     | Asia    |       |
| Region         | char(26)                                                                              | NO   |     |         |       |
| SurfaceArea    | float(10,2)                                                                           | NO   |     | 0.00    |       |
| IndepYear      | smallint(6)                                                                           | YES  |     | NULL    |       |
| Population     | int(11)                                                                               | NO   |     | 0       |       |
| LifeExpectancy | float(3,1)                                                                            | YES  |     | NULL    |       |
| GNP            | float(10,2)                                                                           | YES  |     | NULL    |       |
| GNPOld         | float(10,2)                                                                           | YES  |     | NULL    |       |
| LocalName      | char(45)                                                                              | NO   |     |         |       |
| GovernmentForm | char(45)                                                                              | NO   |     |         |       |
| HeadOfState    | char(60)                                                                              | YES  |     | NULL    |       |
| Capital        | int(11)                                                                               | YES  |     | NULL    |       |
| Code2          | char(2)                                                                               | NO   |     |         |       |
+----------------+---------------------------------------------------------------------------------------+------+-----+---------+-------+
15 rows in set (0.00 sec)
```

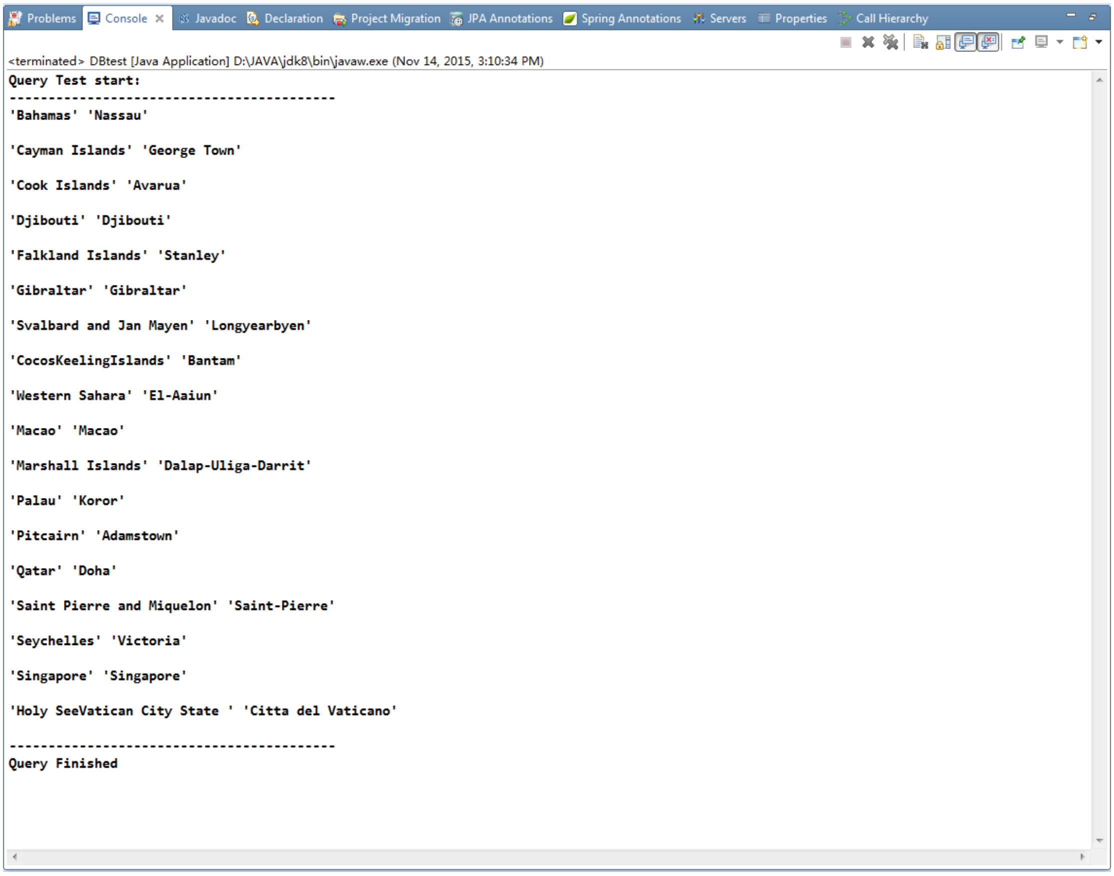
Figure 1



```
ERROR 1054 (42S22): Unknown column 'city.Polulation' in 'where clause'
mysql> SELECT country.Name,city.Name FROM city,country
    -> WHERE city.CountryCode=country.Code AND city.Population>0.4*country.Population;
+-------------------------------+---------------------+
| Name                          | Name                |
+-------------------------------+---------------------+
| Bahamas                       | Nassau              |
| Cocos (Keeling) Islands       | Bantam              |
| Cook Islands                  | Avarua              |
| Cayman Islands                | George Town         |
| Djibouti                      | Djibouti            |
| Western Sahara                | El-Aaiún            |
| Falkland Islands              | Stanley             |
| Gibraltar                     | Gibraltar           |
| Macao                         | Macao               |
| Marshall Islands              | Dalap-Uliga-Darrit  |
| Pitcairn                      | Adamstown           |
| Palau                         | Koror               |
| Qatar                         | Doha                |
| Singapore                     | Singapore           |
| Svalbard and Jan Mayen        | Longyearbyen        |
| Saint Pierre and Miquelon     | Saint-Pierre        |
| Seychelles                    | Victoria            |
| Holy See (Vatican City State) | Città del Vaticano  |
+-------------------------------+---------------------+
18 rows in set (0.02 sec)

mysql> WHERE city.CountryCode=country.Code AND city.Population>0.4*country.Population;
```

Figure 2



```
<terminated> DBtest [Java Application] D:\JAVA\jdk8\bin\javaw.exe (Nov 14, 2015, 3:10:34 PM)
Query Test start:
------------------------------------------
'Bahamas' 'Nassau'

'Cayman Islands' 'George Town'

'Cook Islands' 'Avarua'

'Djibouti' 'Djibouti'

'Falkland Islands' 'Stanley'

'Gibraltar' 'Gibraltar'

'Svalbard and Jan Mayen' 'Longyearbyen'

'CocosKeelingIslands' 'Bantam'

'Western Sahara' 'El-Aaiun'

'Macao' 'Macao'

'Marshall Islands' 'Dalap-Uliga-Darrit'

'Palau' 'Koror'

'Pitcairn' 'Adamstown'

'Qatar' 'Doha'

'Saint Pierre and Miquelon' 'Saint-Pierre'

'Seychelles' 'Victoria'

'Singapore' 'Singapore'

'Holy SeeVatican City State ' 'Citta del Vaticano'

------------------------------------------
Query Finished
```

Figure 3