Spark & MySQL

INF 551

Wensheng Wu

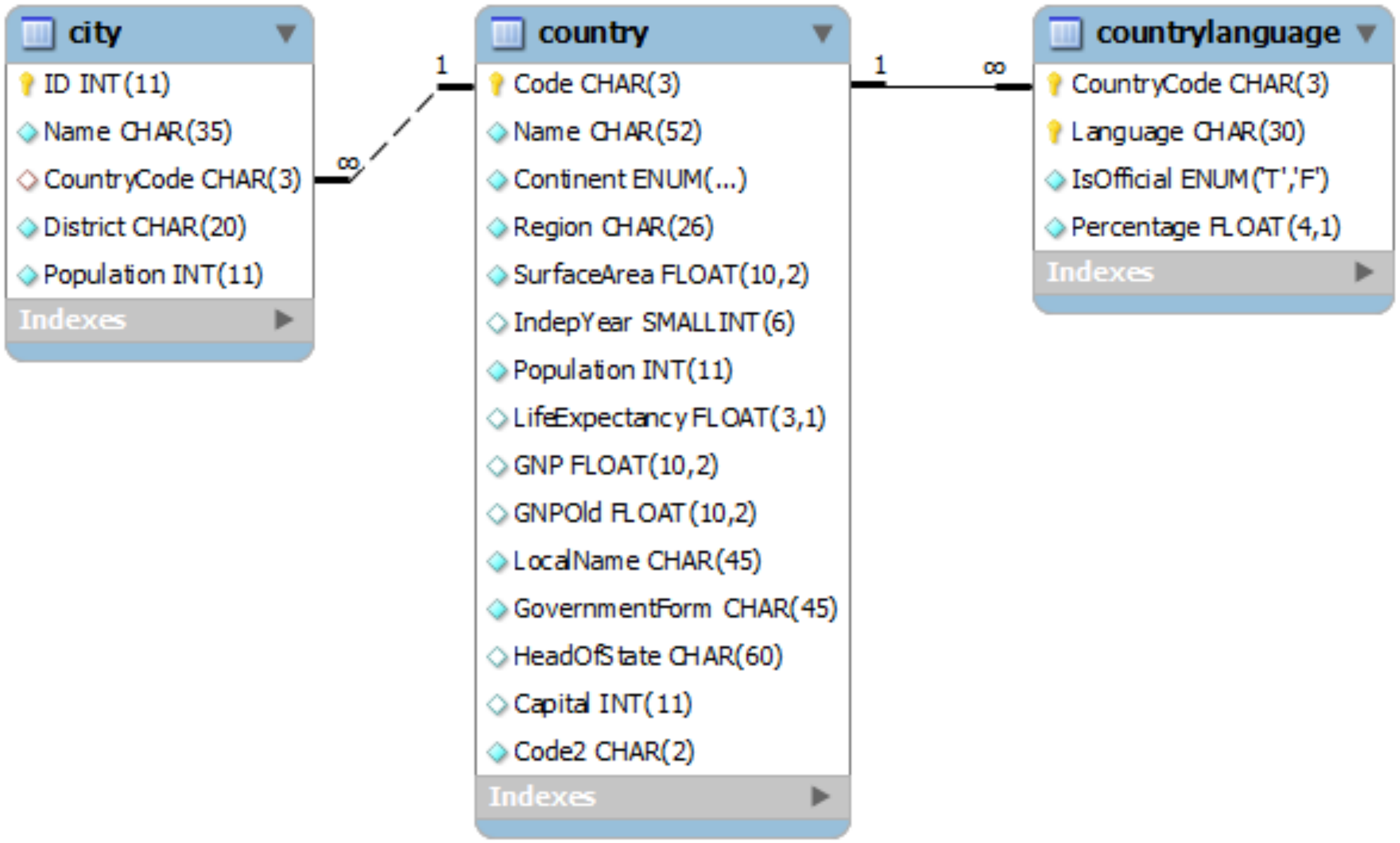
Installing world database in MySQL

- wget http://downloads.mysql.com/docs/world.sql.gz

- gunzip world.sql.gz

- mysql -u root -p < world.sql

Schema of world database



Grant access to world database to user inf551

- grant all privileges on world.* to inf551@localhost;

- Remember that you have created inf551 user earlier
  - If not, log in as MySQL root; then execute:
    - create user 'inf551' identified by 'inf551';

Download MySQL jdbc driver package

- wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.41.tar.gz

- tar xvf mysql-connector-java-5.1.41.tar.gz

- cd mysql-connector-java-5.1.41

- cp mysql-connector-java-5.1.41-bin.jar ~/spark-2.1.0-bin-hadoop2.7

Replace this with the your spark installation directory

Run pyspark

- bin/pyspark --driver-class-path mysql-connector-java-5.1.41-bin.jar

Loading table as a data frame

- country = spark.read.format("jdbc").option("url", "jdbc:mysql://localhost:3306/world").option("dbtable", "country").option("user", "inf551").option("password", "inf551").load()

- country.show()

Data frame operations

- country.select("name").show()
  - Similar to "select name from country"

- country.filter(country['GNP'] > 800)
  - Similar to "select * from country where gnp > 800"

Data frame operations

- country.groupBy("continent").count().show()

```
+-------------+-----+
|    continent|count|
+-------------+-----+
|       Europe|   46|
|       Africa|   58|
|North America|   37|
|   Antarctica|    5|
|South America|   14|
|      Oceania|   28|
|         Asia|   51|
+-------------+-----+
```

Run SQL queries on data frame

- country.createOrReplaceTempView("country")
- sqlDF = spark.sql("SELECT * FROM country")
- sqlDF.show(10)
  - Show first 10

Example with where clause

- sqlDF = spark.sql("SELECT name FROM country where gnp > 1000000")

- sqlDF.show()

```
+--------------+
|          name|
+--------------+
|       Germany|
|        France|
|United Kingdom|
|         Italy|
|         Japan|
| United States|
+--------------+
```

Example with group-by

- sqlDF = spark.sql("SELECT continent, sum(gnp) FROM country group by continent having sum(population) > 1000000 order by continent")

- sqlDF.show()

```
+-------------+---------+
|    continent| sum(gnp)|
+-------------+---------+
|       Africa| 580375.0|
|         Asia|7655392.0|
|       Europe|9498865.0|
|North America|9688627.2|
|      Oceania| 419774.7|
|South America|1511874.0|
+-------------+---------+
```

Loading more tables

- city = spark.read.format("jdbc").option("url", "jdbc:mysql://localhost:3306/world").option("dbtable", "city").option("user", "inf551").option("password", "inf551").load()

- countrylanguage = spark.read.format("jdbc").option("url", "jdbc:mysql://localhost:3306/world").option("dbtable", "countrylanguage").option("user", "inf551").option("password", "inf551").load()

Create more SQL temp views

- city.createOrReplaceTempView("city")

- countrylanguage.createOrReplaceTempView("
  countrylanguage")

Join example

- sqlDF = spark.sql("SELECT city.name, country.name, continent from city, country where city.countrycode = country.code and continent like '%America%'")

- sqlDF.show(10)

Convert dataframe to RDD

- city.rdd
  - Rdd is a list of rows
  - Each row is a (named) tuple

- city.rdd.take(1)
  - [Row(ID=1, Name=u'Kabul', CountryCode=u'AFG', District=u'Kabol', Population=1780000)]

Examples of RDD operations

- city.rdd.map(lambda r: r.ID).take(5)
  - [1, 2, 3, 4, 5]

- city.rdd.filter(lambda r: r.CountryCode == 'USA').map(lambda x: x.Name).take(5)
  - [u'New York', u'Los Angeles', u'Chicago', u'Houston', u'Philadelphia']

Encoding error

- If you see error like this:
  - UnicodeEncodeError: 'ascii' codec can't encode character u'\xf4' in position 1

- Add this to the beginning of your script:
  - import sys
  - reload(sys)
  - sys.setdefaultencoding('utf-8')

Using spark-submit

- bin/spark-submit --driver-class-path mysql-connector-java-5.1.41-bin.jar q1.py
  - Need to specify driver class path too

Resources

- Spark SQL, DataFrames, and Datasets Guide
  - http://spark.apache.org/docs/latest/sql-programming-guide.html#datasets-and-dataframes

- Example MySQL databases:
  - https://dev.mysql.com/doc/index-other.html