Big Data Systems - CS4545/CS6545
Hands-on 1
Due: February 2, 2021 at 11 am

In this hands-on you will work with relational algebra expression to build queries using Apache Calcite.

INSTRUCTIONS FOR SETTING UP THE CODE

Follow steps 0 through 6 below.

0. Launch a VM with Eclipse installed.

Option 1. If you are remotely connected to the FCS lab (see *RemoteDesktopToALabMachine.pdf*)
Launch the BigDataSystems Master VM. Then launch a Terminal.

Option 2. If you have installed the provided VM image in your local machine (see *HandsonVMSetup.pdf*)
Launch the database_admin VM locally. Then launch a Terminal.

1. Get the handson1_lab.zip file by the typing the command in the Terminal:

```
$ wget http://www.cs.unb.ca/~sray/teaching/bds/handson/handson1_lab.zip
```

2. Unzip handson1_lab.zip and there are a few top level elements: $ unzip handson1_lab.zip
The contents of the file include
 - org-apache-calcite-jdbc.properties
 - src/
 - lib/

3. Eclipse Java project.

Option 1. If you are remotely connected to the FCS lab
Launch the Eclipse IDE already included in the VM under:
 /home/bigdata/eclipse/java-mars/eclipse
The eclipse workspace is under /home/bigdata/eclipse/java-mars/eclipse/workspace

Create a new Eclipse Java project named *calcite* .

Option 2. If you have installed the provided VM image in your local machine.
Launch the Eclipse IDE from "Show Applications" menu.

The eclipse workspace is under /home/bigdata/eclipse-workspace
There is already an Eclipse Java project named *calcite* .

4. Overwrite your *calcite* Eclipse project's workspace content with the top level elements in step #2:
- org-apache-calcite-jdbc.properties
 - src/
 - lib/
Then overwrite <eclipse workspace>/calcite/bin/resources with the src/resources in step #2.

5. Include all the .jar files in /lib into your project (it is already done, if you installed the VM locally).

6. Refresh the Eclipse project. Enable option “Build Automatically” from Project menu item.

INSTRUCTIONS FOR RUNNING THE EXAMPLES

1. The example data files are under ./src/resources/company/. The tables are stored as .csv files. The schema of the tables are given below:

```
EMPLOYEE(EMPID:int,NAME:string,DEPTNO:int,GEN-
DER:string,CITY:string,AGE:int,SLACKER:boolean,MGRID:int,JOINEDAT:date,SALARY:float)
DEPT(DEPTNO:int,NAME:string)
CERTIFICATE(EMPID:int, COURSEID:int,COMPLETIONDATE:date)
COURSE(COURSEID:int,TITLE:string, CATEGORYID:int)
CCATEGORY(CATID:int, CATNAME:string)
```

The included example java program ***RelAlgebraDemo.java*** uses Calcite CSV adapter to treat the above .csv files as data tables and thus enables you to run queries on them directly without any database engine.

Look at methods `example0()` through `example7()` in ***src/RelAlgebraDemo.java*** for various examples of how to write relational algebra expression using Apache Calcite. You can run them by running ***RelAlgebraDemo*** as a Java application.

Please refer to lecture notes L03_CS4545_CS6545_QueryProcessing.pdf posted in D2L for more on these examples.

For a tutorial on Apache Calcite see https://calcite.apache.org/docs/algebra.html
For Apache Calcite Java APIs see https://calcite.apache.org/apidocs/org/apache/calcite/tools/RelBuilder.html

INSTRUCTIONS FOR THE TASK (TO BE SUBMITTED FOR HANDSON)

The data files are under ./src/resources/world/. The tables are stored as .csv files. The schema of the tables are given below:

```
capitals(CapitalCity:string,Country:string,Country
code:string,Latitude:double,Longitude:double)

countries(Country:string,Population:long,Area:int,Currency:string)

major_cities(City:string,Country:string,CityPopulation:int)
```

Look for the methods: `runQuery1()`, `runQuery2()` and `runQuery3()` in ***src/BDS_handson1.java*** and solve them. In particular, write the relational algebra expressions for following queries:

1. Query 1: Show the name, population and area of the 5 largest countries by area (in descending order).

2. Query 2: For each country that has a Megacity (i.e. city with population more than 10 million), show the name of country and the number of Megacities it has.

3. Query 3: For each country whose capital is a major city, show the name of the country, country code, capital and the population of its capital.

Tip: it may be useful to write down the relational algebra expression in a piece of paper before you code it using Calcite. This piece of paper need not be submitted for the hands-on.

NOTE

The files modified in the FCS lab virtual machines are not persisted. So, before you shutdown your virtual machine, you may want to save your modified file in your UNB account.

You can access your UNB user account's files from the Big Data Systems Master VM by accessing the folder /media/sf_FCS-HomeDir.
For instance, execute the commands below to copy a file from your user account to the VM:

```
cd /media/sf_FCS-HomeDir
cp <src file>  /home/bigdata/DataScience/workspace/.
```

SUBMISSION INSTRUCTIONS :

1) Submit the following:
    a. ***BDS_handson1.java*** file with your **solution**
    b. ***Soln.txt file*** with the **relational algebra expression** generated by Calcite based on your solution
2) Submit through Desire To Learn (D2L) course drop box Hands-on1
3) Hands-ons not submitted electronically via Desire To Learn or submitted after the due date will NOT be marked.