

Programming Assignment 4:

HBase

Imani Palmer

Last update: September 14, 2015

1 Overview

Welcome

Welcome to HBase Machine Practice. It is highly recommended that you practice the **Tutorial: Introduction to HBase** before beginning this assignment.

2 Requirements

This assignment is designed to work on and will be graded based on the **Hortonworks Sandbox 2.3** virtual machine. You need to have a working HortonWorks Sandbox machine either locally or on the Amazon Web Services.

All assignments are also designed based on **JDK 7** (included in the virtual machine).



Please refer to “Tutorial: Run HortonWorks Sandbox 2.3 Locally” or “Tutorial: Run HortonWorks Sandbox 2.3 on AWS” for more information.



For a quick review of how to use this virtual machine and MapReduce, take a look at “Tutorial: Introduction to Hadoop MapReduce.”

3 Setup

Step 1: Start the virtual machine, and then connect to it through the SSH.

Step 2: After successfully logging in, you should see a prompt similar to the following:

```
[root@sandbox ~]#
```

Step 3: Start HBase Service:

```
# bash ~/start-hbase.sh
```

Step 4: Download the assignments files:

```
# git clone https://github.com/xldrx/cloudapp-mp4.git
```

Step 5: Change the current folder to:

```
# cd cloudapp-mp4/
```

Step 6: Finish the exercises by editing the provided template file. All you need to do is complete the parts marked with **TODO**.

- Each exercise should be implemented in one file only. Multiple file implementation is not allowed.
- Only standard JDK 7 and Apache Hadoop 2.7.1 libraries may be used. In other words, **the code should be compiled and run on a vanilla HortonWorks Sandbox VM.**

More information about this exercise is provided in the next section.

Step 8: After you are done with the assignments, make a submission using the following command (it is not possible to make the submission through the Coursera website):

```
# bash submit.sh
```

This command will **restore** the changes to the **HBase**, then run some tests on your codes. After that, you will be asked for your **Submission Login** and **Submission Password**. Make sure these are an exact match with the information on the assignment page on Coursera; otherwise, your submission might not be graded. Remember **Submission Login** is different than your **Coursera ID**, also **Submission Password** is not the same as your **Coursera Password**.

Step 9: You may check the score of your submission on the Coursera Programming Assignments web page. Please note that it takes a while (sometimes a few hours) for your submission to be graded. If you are not completely satisfied with your grade, you have a chance to make more submissions until the deadline.

Exercise: Create and Populate an HBase Table

In this exercise, you are going to implement a simple HBase application. To make the implementation easier, we have provided a boilerplate for this exercise in the following file:

SuperTable.java

All you need to do is make the necessary changes in the file.

Your application creates a new table named “powers” in HBase and populate it with the following data:

Row key	personal		professional	
	hero	power	name	xp
row1	superman	strength	clark	100
row2	batman	money	bruce	50
row3	wolverine	healing	logan	75

For simplicity treat all data as **String** (i.e. don't use any numerical data type).

After that, output the results from scanning the **personal.hero** column of the table to the standard output (console). The output should be similar to this:

```
keyvalues={row1/personal:hero/1442104171204/Put/vlen=8/seqid=0}
```

This is just a sample. Actual values may be slightly different.

Remember that the output is case sensitive. Furthermore, your application should not remove the table and its values at the end of execution.