

CSCI312 Big Data Management
Singapore 2021-4
Assignment 3
Published on 8 November 2021

Scope

The objectives of Assignment 3 implementation of HBase table, querying and manipulating data in HBase table, simple data processing with Pig, and data processing with Spark.

This assignment is due on **Saturday, 22 November 2020, 8:00pm** (sharp) Singaporean Time (ST).

This assignment is worth **20%** of the total evaluation in the subject.

Only electronic submission through Moodle at:

<https://moodle.uowplatform.edu.au/login/index.php>

will be accepted. All email submissions will be deleted and mark 0 ("zero") will be immediately granted for Assignment 3. A submission procedure is explained at the end of Assignment 3 specification.

A policy regarding late submissions is included in the subject outline.

Only one submission of Assignment 3 is allowed and only one submission per student is accepted.

A submission marked by Moodle as "late" is always treated as a late submission no matter how many seconds it is late.

A submission that contains an incorrect file attached is treated as a correct submission with all consequences coming from the evaluation of the file attached.

All files left on Moodle in a state "Draft (not submitted) " will not be evaluated.

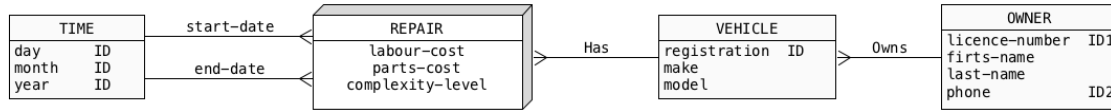
A submission of compressed files (zipped, gzipped, rared, tared, 7-zipped, lhzed, ... etc) is not allowed. The compressed files will not be evaluated.

The second assignment is an **individual assignment** and it is expected that all its tasks will be solved **individually without any cooperation** with the other students. However, it is allowed to declare in the submission comments that a particular component or task of this assignment has been implemented in cooperation with another student. In such a case evaluation of a task or component may be shared with another student. In all other cases plagiarism will result in a **FAIL** grade being recorded for entire assignment. If you have any doubts, questions, etc. please consult your lecturer or tutor during laboratory/tutorial classes or over e-mail.

Task 1 (5 marks)

Design and implementation of HBase table

Implement as a single HBase table a database that contains information described by the following conceptual schema.



- (1) Create HBase script `solution1.hb` with HBase shell commands that create HBase table and load sample data into the table. Load into the table information about at least two vehicles such that each one required two repairs. Do not forget about the owners of the vehicles.

When ready use HBase shell to process a script file `solution1.hb` and to save a report from processing in a file `solution1.rpt`.

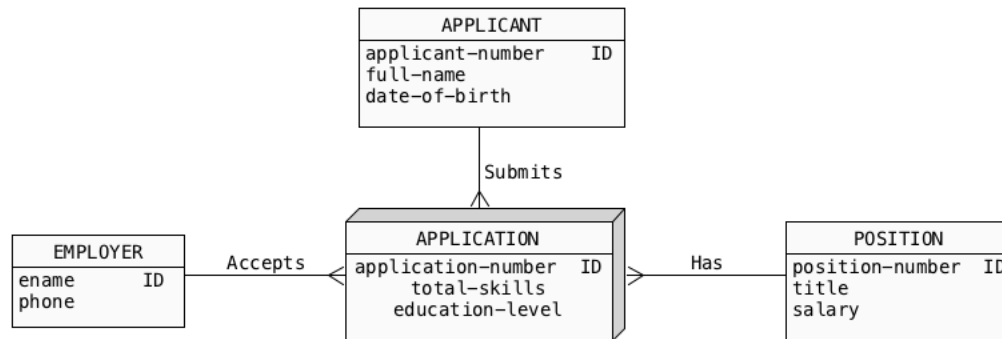
Deliverables

A file `solution1.rpt` that contains a report from processing of `solution1.hb` script with the statements that create HBase table and load sample data.

Task 2 (5 marks)

Querying and manipulating data in HBase table

Consider a conceptual schema given below. The schema represents a simple database domain where applicants apply for the positions offered by employers.



Download a file `task2.hb` with HBase shell commands and use HBase shell to process it. Processing of a script `task2.hb` creates HBase table `task2` and loads some data into it.

Use HBase shell to implement the following queries and data manipulations on the HBase table created in the previous step. Save the queries and data manipulations in a file `solution2.hb`.

- (1) Find all information about a position, that has a number 312, list one version per cell.
- (2) Find all information about an application for a position 312 performed by an applicant 007, list one version per cell.
- (3) Delete a column family EMPLOYER.
- (4) Add to a column family POSITION a column that contains information about the total number of applications. Next, update the values in the new column for all positions recorded in a database.
- (5) Increase the total number of versions in each cell of a column family APPLICANT.

When ready, start HBase shell and process a script file `solution2.hb` with Hbase command shell. When processing is completed copy the contents of Command window with a listing from processing of the script and paste the results into a file `solution2.rpt`. Save the file. When ready submit a file `solution2.rpt`.

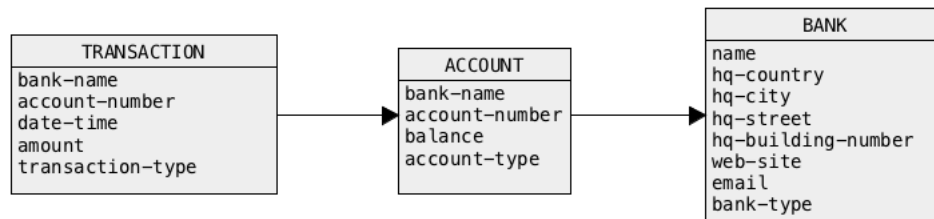
Deliverables

A file `solution2.rpt` with a listing from processing of a script file `solution2.hb`.

Task 3 (5 marks)

Data processing with Pig Latin

Consider the following logical schema of a relational database.



Download a file `task3.zip` published on Moodle together with a specification of Assignment 3 and unzip it. You should obtain a folder `TASK3` with the following files: `bank.tbl`, `account.tbl`, and `transaction.tbl`. The files contain data dumped from a relational database whose logical schema is given above.

Use a text editor to examine the contents of `*.tbl` files.

(1) Transfer the files into HDFS.

Create Pig Latin script `solution3.pig` that implements the following queries.

- (2) Find the account numbers (`account-number`) opened in any construction (`bank-type`) bank.
- (3) Find the names of banks (`bank-name`), that have no accounts opened in the banks.
- (4) Find the total number of accounts opened in each bank located in Japan (`hq-country`).
- (5) Find the account numbers (`account-number`) of all accounts that have been used by both deposit and withdrawal transactions (`transaction-type`).
- (6) Find the account numbers (`account-number`) of all accounts that have been used only by deposit transactions (`transaction-type`).

When ready, use `pig` command line interface to process a script `solution3.pig` and to save a report from processing in a file `solution3.rpt`.

Deliverables

A file `solution3.rpt` with a report from processing of Pig Latin script `solution3.pig`.

Task 4 (5 marks)

Data processing with Spark

Consider the following sales related information.

```
bolt 45  
bolt 5  
drill 1  
drill 1  
screw 1  
screw 2  
screw 3
```

Load the sales related information listed above into a text file `sales.txt` and later on load the file into HDFS.

An objective of this task is to *find the total sales per part* using three different techniques: Resilient Distributed Datasets, Datasets, and DataFrames with SQL.

Use Spark command line interface to implement the following tasks.

- (1) Load the contents of a file `sales.txt` located in HDFS into a Resilient Distributed Dataset (RDD) and use RDD to find the total sales per part.

When ready copy the contents of Terminal screen with a report from implementation of a task (1) and paste it into a file `solution4.rpt`.

- (2) Load the contents of a file `sales.txt` located in HDFS into a Dataset and use the Dataset to find the total sales per part.

When ready copy the contents of Terminal screen with a report from implementation of a task (2) and paste/append it at the end of a file `solution4.rpt`.

- (3) Load the contents of a file `sales.txt` located in HDFS into a DataFrame and use SQL to find the total sales per part.

When ready copy the contents of Terminal screen with a report from implementation of a task (3) and paste/append it at the end of a file `solution4.rpt`.

Deliverables

A file `solution4.rpt` with a report from of implementation of the tasks (1), (2), and (3) .

Submission of Assignment 3

Note, that you have only one submission. So, make it absolutely sure that you submit the correct files with the correct contents. No other submission is possible !

Submit the files **solution1.rpt**, **solution2.rpt**, **solution3.rpt**, and **solution4.rpt** through Moodle in the following way:

- (1) Access Moodle at **<http://moodle.uowplatform.edu.au/>**
- (2) To login use a **Login** link located in the right upper corner the Web page or in the middle of the bottom of the Web page
- (3) When logged select a site **ISIT312 (SP421) Big Data Management**
- (4) Scroll down to a section **SUBMISSIONS**
- (5) Click at **In this place you can submit the outcomes of your work on the tasks included in Assignment 3** link.
- (6) Click at a button **Add Submission**
- (7) Move a file **solution1.pdf** into an area **You can drag and drop files here to add them**. You can also use a link **Add...**
- (8) Repeat step (7) for the remaining files **solution1.rpt**, **solution2.rpt**, **solution3.rpt**, and **solution4.rpt**.
- (9) Click at a button **Save changes**
- (10) Click at a button **Submit assignment**
- (11) Click at the checkbox with a text attached: **By checking this box, I confirm that this submission is my own work, ...** in order to confirm authorship of your submission.
- (12) Click at a button **Continue**

End of specification