SIM GLOBAL EDUCATION

UOW AUSTRALIA

Family Name .....

First Name .....

SCIT

School of Computing and

Information Technology

ISIT312
Big Data Management

This paper is for students studying at the Singapore Institute of Management Pte Ltd.

S2-2021 FINAL EXAMINATION

Date: 07 June 2021

Time: 10.00 am – 1.40 pm SGT

Exam value: **40% of the subject assessment**

Marks available: **40 marks.**

DIRECTIONS TO CANDIDATES

(1) The answers to the questions included in the final examination must be hand written with a BLACK or DARK BLUE PEN on the WHITE PIECES of paper format. No pencil and no other colour of paper is allowed.

(2) When finished, take the pictures of the hand-written solutions, save the pictures in the files (`jpeg`, `jpg`, `gif`, `bmp`, `png`, `pdf` formats are all acceptable), and submit the files through Moodle. Using mobile phone cameras is all right. It is possible to take more than one picture per answer to assure the good readability of an answer. The marks will be deducted for submissions in the different formats. No more than 20 files can be submitted and no more than 200Mbytes can be submitted. Please, plan well your pictures.

(3) The files must have the names indicating a number of the respective questions in the final examination paper like `q1`, `q2`, ... and `q1-1`, `q1-2`, ... when more than one picture is used for an answer of a question. Marks will be deducted for the incorrect file names.

(4) All answers including the drawings must be hand written. No printed material will be evaluated.

(5) Marks will be deducted for the late submissions at a rate of 1 mark per 1 minute late.

Question 1 (8 marks)

Assume that a file `orderline.txt` contains information about the products included in the orders submitted by the customers. For example, the first five rows in a sample file with the information about the ordered products contain the following lines.

```
000001 01 C001 bolt 200
000001 02 C001 screw 20
000002 01 C002 bolt 100
000002 02 C002 nut 50
000003 01 C002 screw 10
...    ... ...  ...    ...
```

The first value in each line is an order number, for example `000001` in the first line. The next value in each line is a line number in on order, for example `01` in the first line. The next value in each line is a customer code, for example `C001` in the first line. The next value in each line is a name of an item ordered by a customer, for example `bolt` in the first line. Finally, the last value in each line is a total number of items ordered, for example `200` in the first line.

Assume, that a file `orderline.txt` has been loaded to HDFS. A location of the file is up to you.

Your task is to explain how to implement a MapReduce application, that finds the total number of ordered items per each item. For example, when applied to the first five lines your application should return the following lines.

```
bolt 300
screw 30
nut 50
```

You must specify the parameters (if any) of your application and the key-value data in the input and output of the Map and Reduce stages.

There is no need to write Java code, however, if you like it then it is all right to do so. The precise explanation in plain English or in a pseudocode will do.

Question 2 (10 marks)

An objective of this task is to create a conceptual schema of a sample data warehouse domain described below. Read and analyse the following specification of a data warehouse domain.

A management of a large and busy (maybe not too busy at the moment, but let us hope that the things will get better in the future) airport would like to create a data warehouse to store information about the arrivals and departures of the passengers.

It is expected that the planned data warehouse would contain historical information collected over a long period of time.

A data warehouse supposed to contain information about the passengers, arrivals and departures of flights, airport staff members, security staff members and gates used for departures and arrivals.

A passenger is described a name, nationality and a number of identification documents. An airport staff member is described an employee number, and full name. A departure/arrival gate is described by a gate number and capacity. A departure or arrival of a flight is described by a flight number, departure or arrival time and departure or arrival date. Additionally, each flight is described by a type of airplane used and destination airport for departures or origin airport for arrivals.

A data warehouse must be designed such it would be possible to easily implement the following classes of applications.

(1) *Find the total number of arriving passengers per year, per month, and per week, per arrival gate, per airport staff members involved in arrivals, per security staff members involved in arrivals, per airplane type, per flight number, per origin airport. For example, it should be possible to find the total number of arriving passengers per origin airport in 2019 and in 2020.*

(2) *Find the total number of departing passengers per year, per month, and per week, per arrival gate, per airport staff members involved in arrivals, per security staff members involved in arrivals, per airplane type, per flight number, per origin airport. For example, it should be possible to find the total number of departing passengers per origin airport in 2019 and in 2020.*

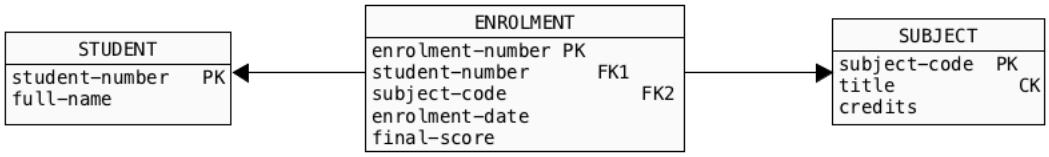
(3) *Find the aggregations of the measures obtained from the recorded activities of arriving and departing passenger like for example total amount of money spent at the shops located at the airport, total amount of money spent at the restaurants, total amount of tax returns, total time spent for waiting at a gate when departing or in an airplane when arriving.*

For example, it should be possible to find the total amount money spent at the shops per year and per departing flight number. Or, it should be possible to find an average time spent while waiting for departures per year, per flight number, per destination airport, or an average time spent waiting in a queue to customs per arriving passenger, etc.

To draw a conceptual schema, use a graphical notation explained to you in a presentation 11 Conceptual Data Warehouse Design.

Question 3 (6 marks)

Consider the following logical schema, that implements a two-dimensional data cube.



The data cube contains information about the enrolments of subjects performed by the students.

Assume, that the files `student.txt`, `subject.txt`, and `enrolment.txt` contain data consistent with a logical schema of two-dimensional data cube given above. Internal format of each file is a sequence of values separated with the commas (CSV format).

(1) Write a sequence of commands, that load the files into HDFS. A location for the files in HDFS is up to you.

(1 mark)

(2) Write HQL statements that create the Hive tabular views of the files `student.txt`, `subject.txt`, and `enrolment.txt` loaded into HDFS.

(1 mark)

(3) Write HQL statements to retrieve the following information from the data warehouse. Each correctly implemented statement is worth 1 mark.

(i) Find the total number of enrolments per student, per subject, and per both students and subject and the total number of enrolments. List the values of the attributes: `student-number` and `subject code` and the total number of enrolments.

(ii) For each subject and for each year list `year(enrolment-date), subject-code` and the `scores` in a subject in a year ordered in an ascending order of scores, and an average of all scores in a year.

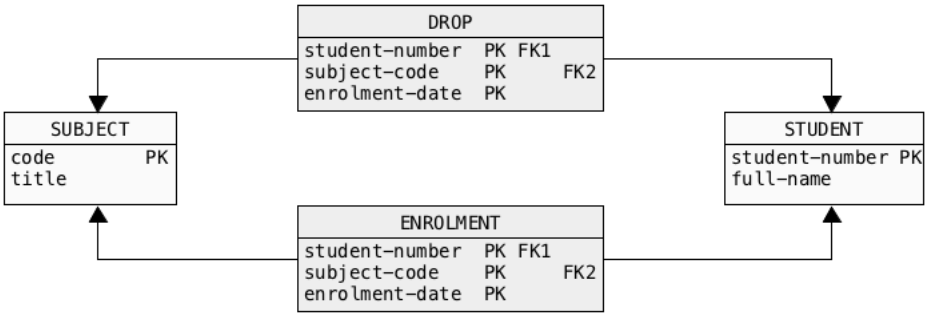
(iii) Find an average score in all subjects per year, and per both subject and year and per both student and year. You can use the row functions `year` to extract a year from a date. List the values of the attributes: `year(enrolment-date), student-number` and `subject-code` and an average score.

(iv) For each student and for each subject list a pair: `student-number` and `subject-code` together with an average score of all subjects enrolled by a student.

(4 marks)

Question 4 (7 marks)

Consider the following logical schema of a relational database, that implements a data cube with historical information related to the subjects enrolled and dropped by the students.



(1) Write HBase shell commands to create a single HBase table, that implements a logical schema given above.

Write HBase commands to load into the table information about at least two subjects, one student, two enrolments and one drop. Please remember, that the students are allowed to enrol and/or drop many subjects and a subject can be enrolled dropped by many students. Your Hbase table must be created in a way, that does not contribute to any data redundancies when information about students, subjects, enrolments and drops is entered into the table.

(3 marks)

(2) Write HBase shell commands, that implement the following queries and data manipulations on the HBase table created and loaded with data in the previous step. Each correctly implemented task is worth 1 mark.

(i) Find all information (student number and full name) about the students enrolled in a subject `ISIT312`.

(ii) Find all information (subject code and title) about a subject `ISIT312`.

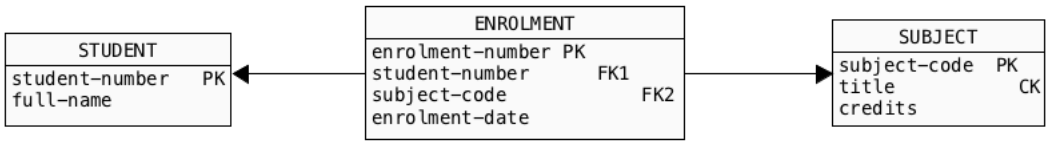
(iii) Add a column family `LECTURER` and allow for two versions in each cell of the new column family.

(iv) Assume that lecturers are described by an employee number and full name. Insert into the table information about a lecturer and about a subject taught by a lecturer. Assume, that a lecturer teaches one subject and each subject is taught by one lecturer.

(4 marks)

Question 5 (4 marks)

In this question we use the same logical schema of the two-dimensional data cube as in Question 3.



Assume, that the files `student.txt`, `subject.txt`, and `enrolment.txt` contain data consistent with a logical schema of two-dimensional data cube given above. Internal format of each file is a sequence of values separated with the commas (CSV format). Assume, that the files have been already loaded to HDFS.

Write Pig-Latin statements that implement the following queries. A correct implementation of each query is worth 1 mark.

(1) Find the full names of students who enrolled a subject with a code `ISIT312`.

(2) Find the student numbers of students of the customers who never enrolled a subject with a code `ISIT312`.

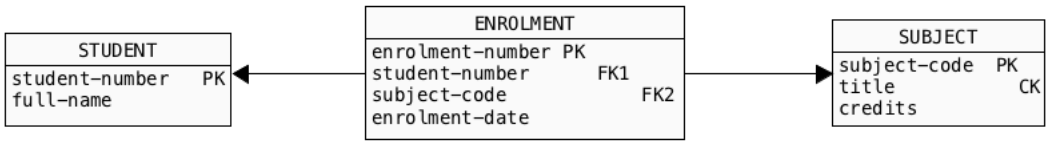
(3) Find the student numbers of students who enrolled both subjects with the codes `ISIT312` and `CSCI317`.

(4) Find the subject codes together with the total number of students enrolled in each subject.

(4 marks)

Question 6 (5 marks)

In this question we use the same logical schema of the two-dimensional data cube as in Question 3.



Assume, that the files `student.txt`, `subject.txt`, and `enrolment.txt` contain data consistent with a logical schema of two-dimensional data cube given above. Internal format of each file is a sequence of values separated with the commas (CSV format). Assume, that the files have been already loaded to HDFS.

Implement the following Spark-shell operations. A correct implementation of each operation is worth 1 mark.

(1) Create the DataFrames, that contain information about students, enrolments and subjects.

(2) Implement a query, that accesses the data frames created in the previous step and finds the total number of enrolments in a subject `ISIT312`.

(3) Implement a query, that accesses the data frames created in the previous step and for each student finds the total number of enrolments performed by a student.

(4) Register the DataFrames, that contains information about the students, enrolments and subjects as SQL temporary views.

(5) Use SQL views created in the previous step to find the titles of subjects together with the total number of students enrolled in each subject.

(5 marks)

End of Examination