School of Computing & Information Technology
University of Wollongong

Session: **4, September 2021**
Lecturer: **Janusz R. Getta**

CSCI312 Big Data Management
Singapore 2021-4
Assignment 2

Published on 18 October 2021

Scope

The objectives of Assignment 2 include conceptual modelling of a data warehouse, implementation of 0NF tables in HQL, implementation of external tables in HQL, and querying a data cube.

This assignment is due on **Monday, 8 November 2021, 8:00pm** (sharp) Singaporean Time (ST).

This assignment is worth **30%** 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 2. A submission procedure is explained at the end of Assignment 2 specification.

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

Only one submission of Assignment 2 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.

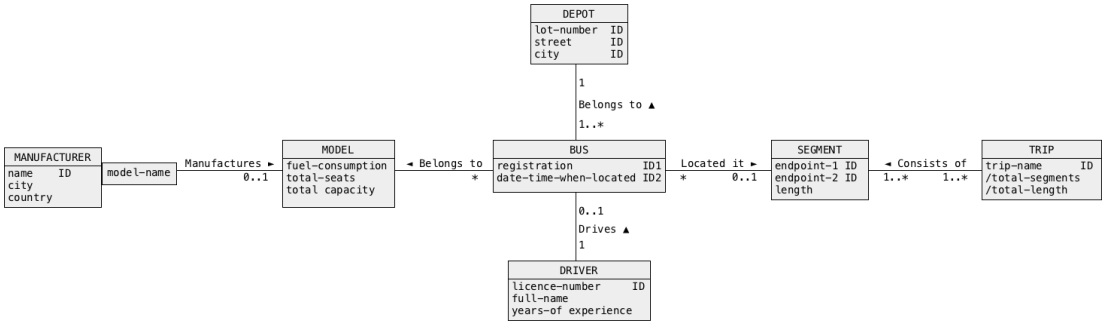
All files left on Moodle in a state **"**`Draft(not submitted)`**"** will not be evaluated.

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 (6 marks)
Intuitive design of a data cube from a conceptual schema of an operational database

A bus transportation company maintains an operational database, that contains information about the current locations of the busses owned by the company. A current location of a bus is determined by a trip segment a bus passes through, in a moment. A trip consists of a sequence of trip segments. A bus can traverse a trip in both directions. It is important to note, that the operational database contains only "point-in-time" information. Each time a bus moves to the next segment all information about its past locations (at the previous segments) is removed from a database. The remaining contents of an operational database is consistent with a conceptual schema given below.



The company would like to implement a data warehouse that can be used to implement the following applications.

(i) find the total number of kilometers travelled by each bus per year, per month per day.

(ii) find the total number of trips performed per bus, per driver, per year, per month, per day.

(iii) find the total number of drivers per trip.

(iv) find the total number travels per a trip segment, per trip, per bus, per year, per month per day.

(v) find an average duration of bus travel per trip segment, per trip, per year, per month per day.

(vi) find the total fuel consumption per trip segment, per trip, per bus model, per manufacturer, per year, per month, per day.

(vii) find the total number of trips per bus, per depot, per city

(viii) find the total number of passengers per segment, per trip, per year, per

month per day.

(ix) find the largest number of passengers per bus, per trip,

(x) find an average number of passengers per trip.

(1) Use a short explanation of a database domain and a conceptual schema given above, to find a data cube, that should be implemented by the bus company to create a data warehouse. In your specification of a data cube, list the facts, the measures, the names of dimensions and the hierarchies.

(2) Pick any three dimensions from a data cube found in the previous step and at least 4 values in each dimension and one measure to draw a sample three-dimensional data cube in a perspective view similar to a view included in a presentation 09 Data Warehouse Concepts, slide 6.

Deliverables

A file `solution1.pdf` that contains

(1) a specification of data cube as a list of facts, measures, dimensions, and hierarchies obtained as a result of task (1),

(2) a perspective drawing of three-dimensional data cube as a result of task (2).

Task 2 (7 marks)
Conceptual modelling of a data warehouse

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 large network of vehicle repair/maintenance facilities would like to create a data warehouse to store information about the facilities located in the different cities, the vehicles maintained at the facilities, and the employees working at the facilities. It is expected that the following information will be stored in the data warehouse.

A vehicle repair/maintenance facility is described by its location (country, city, building number), email address and phone number.

The owners bring their vehicle to the facilities for repairs and/or maintenances. An owner is described by a full name and a unique phone number. A facility keeps the dates when a vehicle was brought for repairs/maintenances and when it was collected by an owner.

A vehicle repair/maintenance facility has a number of employees. An employee is either a mechanic or an admin person. An employee has a unique employee number, first name, last name, and date of birth. The repairs/maintenances are performed by the mechanics. An Implementation of each repair/maintenance requires an involvement of one mechanic. A time spent by a mechanic on a repair/maintenance and all spare parts used are recorded for each repair/maintenance.

The data warehouse must contain information about the total number days spent on repair /maintenance of each vehicle and the total amount of money paid by an owner of a vehicle for repair/maintenance.

A vehicle is described by a registration number, manufacturer, model, and a year when it was manufactured. The owners use credit cards to pay for repairs/maintenances. A credit card number and a name of bank that issued a card is recorded in a database.

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

A management of the vehicle repair/maintenance facilities would like to get from a data warehouse information about

- *the total number of repairs/maintenances per facility, per year, per month per day, per city and per mechanic,*
- *total costs of repairs/maintenance per facility, per year, per month per day, per city and per mechanic,*
- *total number of mechanics involved per repair/maintenance, per year, per month per day, per facility, per city,*
- *average time spend on repairs/maintenances per year, month, day,*
- *total number of parts used for repars/maintenances per year, month, day, per facility, per city.*

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

To create a conceptual schema of a sample data warehouse domain, follow the steps listed below.

Step 1 Find a fact entity, find the measures describing a fact entity.
Step 2 Find the dimensions.
Step 3 Find the hierarchies over the dimensions.
Step 4 Find the descriptions (attributes) of all entity types.
Step 5 Draw a conceptual schema.

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

To draw your diagram, you can use UMLet diagram drawing tool and apply a "Conceptual modelling" notation, Selection of a drawing notation is available in the right upper corner of the main menu of UMLet diagram drawing tool. UMLet 14.3 software is can be downloaded from the subject's Moodle Web site in a section WEB LINKS. A neat hand drawing is still all right.

Deliverables

A file `solution2.pdf` with a drawing of a conceptual schema of a sample data warehouse domain.

Task 3 (5 marks)
Implementation of a table with a complex column type (0NF table) in Hive

Assume that we have a collection of semi-structured data that contains information about the students and their final evaluations (in a scale from 0 to 99) of the enrolled subjects. The first value in each row is a unique student number. Next, we have a list of pairs, that consist of subject code and the final evaluation in a subject.

```
1111,CSIT111:01,CSIT121:23,CSIT101:50,CSIT235:99,ISIT312:02
1112,CSIT101:56,CSIT111:78,CSIT115:10,ISIT312:05
1113,CSIT121:76,CSIT235:87:ISIT312:49
1114,CSIT111:50,ISIT312:45
1115,ISIT115;67,CSCI235:45,CSIT127:56
...
```

(1) Implement HQL script `solution3.hql` that creates an internal table to store information about the student numbers and the evaluations of the subjects. An internal table must be nested (it must be in 0NF).

(2) Include into the script `INSERT` statements that load sample data into the table. You must insert at least 5 records, that have a structure consistent with the records listed above.

(3) Include into the script `SELECT` statements that lists the contents of the table. Assume that no more 5 subjects have been participated by each student..

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

If the processing of the file returns the errors then you must eliminate the errors! Processing of your script must return NO ERRORS! **A solution with errors is worth no marks!**

Deliverables
A file `solution3.rpt` with a report from processing of HQL script `solution3.hql`. The report **MUST NOT include any errors**, and the report **must list all SQL statements processed**.

Task 4 (6 marks)

Implementation of a data warehouse as a collection of external tables in Hive

Download and unzip a file `task4.zip`. You should obtain a folder `task4` with the following files: `author.tbl`, `item.tbl`, `dbcreate.sql`.

Use an editor to examine the contents of `*.tbl` files. The files contain synthetic data extracted from the relational tables. A file `dbcreate.sql` contains `CREATE TABLE` statements used to create the relational tables with the synthetic data.

Transfer the files `author.tbl`, `item.tbl` into HDFS.

(1) Implement HQL script `solution4.hql` to create the external tables, that provide a tabular view of synthetic data included in the files `author.tbl`, `item.tbl`. The external tables must overlap on the files transferred to HDFS in the previous step. It is recommended to use `CREATE TABLE` statements included in a file `dbcreate.sql` to create a file `solution4.sql`.

(2) Include into `solution4.hql` script `SELECT` statements, that list the total number of rows in each table, the total number of rows in both tables, and the first 3 rows from each each one of the external tables implemented in the previous step.

When ready, use a command line interface `beeline` to process a script `solution4.hql` and to save a report from processing in a file `solution4.rpt`.

If the processing of the file returns the errors then you must eliminate the errors! Processing of your script must return NO ERRORS! **A solution with errors is worth no marks!**

Deliverables

A file `solution4.rpt` with a report from processing of HQL script `solution4.hql`. The report **MUST NOT include any errors**, and the report **must list all SQL statements processed**.

Task 5 (6 marks)
Querying a data cube

Download a file `task5.zip` and unzip the file. You should obtain a folder `task5` with the following files: `dbcreate.hql`, `dbdrop.hql`, `partsupp.tbl`, `lineitem.tbl`, and `orders.tbl`.

A file `orders.tbl` contains information about the orders submitted by the customers. A file `lineitem.tbl` contains information about the items included in the orders. A file `partsupp.tbl` contains information about the items and suppliers of items included in the orders.

Open Terminal window and use `cd` command to navigate to a folder with the just unzipped files. Start Hive Server 2 in the terminal window (remember to start Hadoop first). When ready process a script file `dbcreate.hql` to create the internal relational tables and to load data into the tables. You can use either `beeline` or SQL Developer. A script `dbdrop.hql` can be used to drop the tables.

The relational tables `PARTSUPP`, `LINEITEM`, `ORDERS` implement a simple two-dimensional data cube. The relational tables `PARTSUPP` and `ORDERS` implement the dimensions of parts supplied by suppliers and orders. A relational table `LINEITEM` implements a fact entity of a data cube.

(1) Implement the following query using `GROUP BY` clause with `CUBE` operator.

For the order clerks (`O_CLERK`) `Clerk#000000522`, `Clerk#000000154`, find the total number of ordered parts per customer (`O_CUSTKEY`), per supplier (`L_SUPPKEY`), per customer and supplier (`O_CUSTKEY`, `L_SUPPKEY`), and the total number of ordered parts.

(2) Implement the following query using `GROUP BY` clause with `ROLLUP` operator.

For the parts with the keys (`L_PARTKEY`) `7`, `8`, `9` find the largest discount applied (`L_DISCOUNT`) per part key (`L_PARTKEY`) and per part key and supplier key (`L_PARTKEY, L_SUPPKEY`) and the largest discount applied at all.

(3) Implement the following query using `GROUP BY` clause with `GROUPING SETS` operator.

Find the smallest price (`L_EXTENDEDPRICE`) per order year (`O_ORDERDATE`), and order clerk (`O_CLERK`).

Implement the following SQL queries as `SELECT` statements using window partitioning technique.

(4) For each part list its key (`PS_PARTKEY`), all its available quantities (`PS_AVAILQTY`), the smallest available quantity, and the average available quantity. Consider only the parts with the keys `5` and `15`.

(5) For each part list its key (`PS_PARTKEY`) and all its available quantities (`PS_AVAILQTY`) sorted in descending order and a rank (position number in an ascending order) of each quantity. Consider only the parts with the keys `10` and `20`. Use an analytic function `ROW_NUMBER()`.

(6) For each part list its key (`PS_PARTKEY`), its available quantity, and an average available quantity (`PS_AVAILQTY`) of the current quantity and all previous quantities in the ascending order of available quantities. Consider only the parts with the keys `15` and `25`. Use `ROWS UNBOUNDED PRECEEDING` sub-clause within `PARTITION BY` clause.

When ready, save your `SELECT` statements in a file `solution5.hql`. Then, process a script file `solution5.hql` and save the results in a report `solution5.rpt`.

If the processing of the file returns the errors then you must eliminate the errors! Processing of your script must return NO ERRORS! **A solution with errors is worth no marks!**

Deliverables

A file `solution5.rpt` with a report from processing of HQL script `solution5.hql`. The report **MUST NOT include any errors**, and the report **must list all SQL statements processed**.

Submission of Assignment 2

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.pdf**, **solution2.pdf**, **solution3.rpt**, **solution4.rpt**, and **solution5.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 2** 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 **solution2.pdf**, **solution3.rpt**, **solution4.rpt**, and **solution5.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