Course Project:

FindMyBus Database & Web Reporting System

Team Project Phase 2 (12 Marks)

Due: *Week 12 (22:00hrs, Sunday April 2, 2017) to eLearn ONLY*

Hours Expected: *8-10 hours per person*

Preparation

Database Setup

You are given the following:

a. The Logical Design of FindMyBus database in the appendix of this document.

b. Data Setup.zip that contains:

   I. create.txt: script that creates the tables required.

   II. data folder: all data files required for loading into the database

c. Please refer to MySQL_LoadData_handout.pdf for the details of loading data from .txt file into MySQL tables.

Note: You **ARE NOT** allowed to change the values of any sample data by inserting new records or deleting existing records from the sample data. It is mandatory for ALL to use the Database setup provided on eLearn.

Deliverables

1. DDL Scripts

i. **SQL Load script** that loads sample data into the tables created by creation script given.

2. DML Scripts

i. Derive the relevant **DML SQL statements** to support the questions. A **SINGLE SQL statement** is expected for each question**.**

ii. Sample outputs shown are only for your reference on the format of the result. Do not assume that the output shown is correct.

iii. The workloads of teams with different number of members are different.

- For those teams with **4 members**, please complete **ALL 4 DML (a, b, c & d) questions**.
- For those teams with only **3 members**, please complete **3 (a, b & c) DML questions only.**

a. For all service routes, list the service number, the route number, the first stop the route serves along with its location description, and the last stop the route serves along with its location description.

Partial Sample output is as shown below:

Service Number	Route Number	First Stop	First Stop Description	Last Stop	Last Stop Description
…	..	…	...	…	…
63	1	82009	Eunos Int	82009	Eunos Int
..	..	…	...	…	…
74	1	64009	Hougang Ctrl Int	11379	Buona Vista Ter
74	2	11379	Buona Vista Ter	64009	Hougang Ctrl Int
…	..	…	...	…	…

b. List all stop codes, with their location description, address. For each stop code, list the number of routes that the stop serves and display the Terminus status (Yes, if the stop is Terminus, No if the stop is Non-terminus).

Note that if the stop is serving the same service and same route more than once (eg: stop number 84501 serves Service number 60, route number 1 twice), the route count is increased by one instead of two. If the stop is serving the same service but for two different routes, the route count is increased by two. (eg: stop number 17009 serves Service number 7, route numbers 1 and 2).

List the output in the descending order of the number of routes that the stop serves followed by the stop code in ascending order.

Partial Sample output is as shown below:

Stop Code	Location Description	Address	Terminus?	Number of routes served
…	..	…	...	…
01211	Opp Blk 461	Victoria St	No	4
…	..	…	...	…
03239	Shenton Way Ter	Palmer Road	Yes	4
..	..	…	...	…
84501	Christ Ch	Bedok Reservoir Rd	No	1
…	..	…	...	…

c. List all drivers' names along with their license number and date certified. Given a specific period (say 2016-09-21 09:30:00 to 2016-09-21 12:30:00), for each of the drivers listed, list the non-cancelled Trips that he is assigned to (if any) such that the Trip Date of those trips fall in the specified period. For each of the trips, list the bus plate number, service number, route number.

List the output in the ascending order of driver's name, followed by bus plate number.

Partial Sample output is as shown below:

Driver Name	License Number	Date certified	Bus plate	Service Number	Route Number
Abhimanyu Gupta	4759965	2011-07-11	null	null	null
..					
Huo Long	8253440	2012-03-13	SHE8833M	153	1
..	..				

d. List the service number, route number and remarks of route that have the most number of cancelled trips. Also, display the total number of cancelled trips of this service and route. Note that if multiple routes share the same number of (maximum) cancelled trips, all the routes have to be returned.

   The output is expected in the following format:

Service Number	Route Number	Remark	Total Cancelled Trips
…	…	…	…

MySQL supports control functions like **IF** and **IFNULL** that may come in handy.

The **IF** function returns a value based on a condition.

IF(expr, if_true_expression, if_false_expression)

returns "if_true_experssion" if the "expr" evaluates to TRUE, otherwise returns "if_false_expression". The function may be used to return a numeric or string.

As an example, given the following statement IF(column1 IS NULL, 'N/A', column1) the output would be N/A if column1 has NULL value; otherwise the output would be column1 value

The **IFNULL** function accepts two arguments and returns the first argument if the argument is not null; otherwise it returns the second argument.

IFNULL(expr1, expr2)

returns expr1 if expr1 is not NULL, otherwise it returns expr2.

As an example, IFNULL(column1, column2) returns the value in column1 if it is not NULL, and the value of column2 otherwise.

3. Reporting System

Assume that you are going to develop a reporting system for the administrator staff of FindMyBus.

i. The workloads of teams with different number of members are different.
   - For those teams with **4 members**, please complete **ALL 4 (a, b, c & d) Reporting questions**.
   - For those teams with only **3 members**, please complete **3 (a, b & c) Reporting questions only.**

There is no restriction on the number of query statements written to get the desired output.

The design of the output of the system is left to your choice. Just make it user friendly.

a. FindMyBus Service Information

Develop a function for the administrator to enter a Service into a text field.

The function should return and display the following information:

i. For each route that the service operates, list the stop number, stop location description, the order of the stop in the route.

ii. List the details in the ascending order of route, followed by order of the stop

The format suggested:

Route 1

Stop number	**Stop location description**	**Stop order**
StopCode XXX	XXX Location Description	1
StopCode YYY	YYY Location Description	2
…		

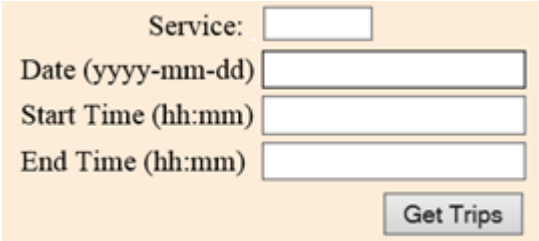
Route 2

Stop number	**Stop location description**	**Stop order**
StopCode AAA	AAA Location Description	1
StopCode BBB	BBB Location Description	2
…		

b. FindMyBus Trip Information

Develop a function for the administrator to enter a service, a date (yyyy-mm-dd), and time period (start time and end time). List all trips scheduled for the service during the search period (inclusive of start and end time)

Assume that the user will provide all input values.



The function should return and display the following information:

i. The trip time, route number, driver's name scheduled to run the trip, bus plate number scheduled to be used in the trip, cancelled status – Cancelled (if the trip was cancelled) and " " (empty string if the trip was not cancelled).

ii. List the details in the ascending order of trip time followed by route number.

The format suggested:

Trip Date	**Trip Time**	**Route number**	**Driver Name**	**Bus plate number**	**Status**
YYYY-MM-DD	hh:mm:00	..	..	..	
..	..	..	..	..	Cancelled
…	..	..	..	..	

c. FindMyBus Stop Search Function

Develop a function for the administrator to enter a stop number. Partial stop number could be entered and the function should return **both the exact and partial** match results.

For example, for an input of 4010, the result would list details of stop numbers 40101 and 40109.

The function should return and display the following:

i. For every Stop number(s) that matches, list stop number, stop address, total count of the services served by the stop.

The format suggested:

Stop Results

Stop number	**Stop address**	**Service numbers served**
Stop X1	xxx	3
Stop X2	xxx	8

d. Driver Function

Develop a function for the administrator to enter a specific number, where the number represents the day of the week, so as to list the drivers who are working on the selected day.

1	2	3	4	5	6	7
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday

The function should return and display the following:

a. Driver staff ID, NRIC, driver name, and license number.
b. List the details in the ascending driver name, followed by staff ID.

The format suggested:

Staff ID	**NRIC**	**Driver Name**	**License Number**
65234	S8475389E	Abbe Tan	5759467
66173	S5097439B	Abhimanyu Gupta	4759965
..	..	..	..
..	..	..	..

Submission

1. Submission is to be done using eLearn
2. The deliverable should be a single zip file GX-TY.zip, where X is the section (1 – 2) and Y is your team number (01-12). For instance, if you belong to G2 and Team 01, your file should be named G2-T01.zip
3. The GX-TY.zip file should contain only the following 3 folders:
   - Folder 1 called DDL that contains a file named ddl.sql.
     - This sql file contains the load data statements such that it enables us to load your data with minimum effort using "execute all" command on MySQLWorkBench
   - Folder 2 called DML that contains a file named dml.sql.
     - This file contains SQL statements to all questions that your team is required to complete.
     - The comment tag followed by the question number should precede every SQL statement. Eg: #a: Select * from Table1;
   - Folder 3 called PHP that contains all files related to reporting system.
     - These files contains php solutions to all questions that your team is required to complete.

Grading Scheme

Project Phase 2	**Marks**
DDL Scripts: SQL Statements for data loading	**1**
DML Scripts: SQL Statements for the required questions	**5**
PHP reporting system: PHP solutions for the required questions	**6**
Grades for individual DML & Reporting questions might vary slightly based on the number of members in the team and the complexity of the query ○ Marks will NOT be awarded for misinterpreting the questions. ○ Clarification should be made online via the eLearn Discussion Forum.	
You will be penalized if your deliverables are not according to our instructions as stated in the above Submission section.	

Project Logical Design