RDBMS and SQL, October–November 2018

Assignment 2: SQL
Due: Midnight, 18 November, 2018

1 Problem Description

You will be provided with a MySQL database dump containing information about trains on Indian Railways. This database has three tables, `schedule`, `stations` and `terminus` that are described in more detail below.

You have to write (a sequence of) SQL queries to construct a table listing all pairs of stations (A,B), where A and B are from the list of terminus stations specified in the table `terminus` such that one can go from A to B with at most two changes of train.

A valid connection is one where the interval between the arrival time of the incoming train and the departure time of the outgoing train is at least 1 hour and at most 5 hours. All trains can be assumed to run daily.

Your output should be a table of the form:

```
(StationA, StationB, Train1, Change1, Train2, Change2, Train3)
```

- If there is a direct train, the columns from `Change1` onwards will be empty.
- If one change is required, `Train1` is the train originating from `StationA`, `Change1` is the station where one changes trains and `Train2` is the second train that reaches `Station B`. `Change2` and `Train3` will be empty.
- If two changes are required, `Train1` is the train originating from `StationA`, `Change1` is the station where one changes from `Train1` to `Train2`, `Change2` is the station where one changes from `Train2` to `Train3` and `Train3` reaches `StationB`.

In this table, `StationA`, `StationB`, `Change1` and `Change2` should all be in terms of the abbreviated station codes while `Train1`, `Train2` and `Train3` will be in terms of the train numbers. Needless to say, in any row of the table, no two stations and no two trains should be the same.

You should submit a zip file containing the following:

- The output table in the format described above.
- A document explaining how you generated the output. This should include the sequence of SQL commands that you executed, along with explanatory notes about what each command is supposed to do.

2 Tables

1. `schedule`

```
+---------+------------+
| Field   | Type       |
+---------+------------+
| train   | int(11)    |
| station | varchar(6) |
| arr     | time       |
| arrdate | int(11)    |
| dep     | time       |
| depdate | int(11)    |
+---------+------------+
```

This table describes each train, one stop at a time. A row in the table of the form

```
(N, S, T1, D1, T2, D2)
```

indicates that train `N` arrives at station `S1` at time `T1` on day `D1` and leaves at time `T2` on day `D2`.

For the originating station of a train, `T1` and `D1` will be blank, while for the final station, `T2` and `D2` will be blank.

Trains are identified by their number, stations by their station code. Since a train journey may last over multiple days, the arrival and departure times are accompanied by the *day number*. Each train starts on day 0, so all stops on the first day have day number 0, stops on the second day have day number 1 etc.

For instance, if train `8888` originates from `Chennai Central (MAS)` at `2345` and next stops at `Arakkonam (AJJ)` at `0045` for 10 minutes, the corresponding entries in the table `schedule` will be

```
(8888,MAS,NULL,NULL,2345,0)
(8888,AJJ,0045,1,0055,1)
```

2. `stations`

```
+---------+-------------+
| Field   | Type        |
+---------+-------------+
| station | varchar(5)  |
| name    | varchar(20) |
+---------+-------------+
```

Lists out the station name for each station code.

3. `terminus`

```
+---------+-------------+
| Field   | Type        |
+---------+-------------+
| station | varchar(5)  |
| name    | varchar(20) |
+---------+-------------+
```

Subset of stations from the table `station`.

3 Instructions to access the data

- Load the dump.

```
mysql -h access2 --user="username" --password="password" dbname -p < dump-file
```

For example,

```
mysql -h access2 --user="madhavan" --password="xxx" madhavandb < railways.dump
```

- Henceforth, invoke `mysql` as usual to load the database.

```
mysql -h access2 --user="username" --password="password" dbname
```