homework

mle

May 9, 2016

Contents

1 SQL Questions

First create a database called fringe$_{\text{shows}}$

```
#terminal
psql
create database fringe_shows;
\q
```

Populate the data using the script - fringeshows.sql

```
#terminal
psql -d fringe_shows -f fringeshows.sql
```

Using the SQL Database file given to you as the source of data to answer the questions. Copy the SQL command you have used to get the answer, and paste it below the question, along with the result they gave.

2

2.1 Revision of concepts that we've learnt in SQL today

1. Select the names of all users.

   ```
   SELECT * FROM users;
   ```

2. Select the names of all shows that cost less than £15.

   ```
   select name from shows where price < 15 ;
   ```

3. Insert a user with the name "Val Gibson" into the users table.

   ```
   insert into users (name) values ('Val Gibson');
   ```

4. Select the id of the user with your name.

   ```
   select id from users where name = 'Hamish Edmondson';
   ```

5. Insert a record that Val Gibson wants to attend the show "Two girls, one cup of comedy".

```
INSERT INTO "shows_users" (show_id, user_id) VALUES ((SELECT id from shows where name =
```

1. Updates the name of the "Val Gibson" user to be "Valerie Gibson".

```
UPDATE users SET name='Valerie Gibson' WHERE (name='Val Gibson');
```

1. Deletes the user with the name 'Valerie Gibson'.

```
DELETE from users where name='Valerie Gibson';
```

1.

```
delete id from shows where id = (select id from shows_users where show_id is NULL);
```

2.2   Section 2

This section involves more complex queries. You will need to go and find out about aggregate funcions in SQL to answer some of the next questions.

1. Select the names and prices of all shows, ordered by price in ascending order.

1. Select the average price of all shows.

1. Select the price of the least expensive show.

select name, show where price from shows where price IN(selct max price from shows)

1. Select the sum of the price of all shows.

2. Select the sum of the price of all shows whose prices is less than £20.

3. Select the name and price of the most expensive show.

4. Select the name and price of the second from cheapest show.

5. Select the names of all users whose names start with the letter "A".

6. Select the names of users whose names contain "el".

2.3   Section 3

The following questions can be answered by using nested SQL statements but ideally you should learn about JOIN clauses to answer them.

1. Select the time for the Edinburgh Royal Tattoo.

2. Select the number of users who want to see "Le Haggis".

3. Select all of the user names and the count of shows they're going to see.

4. SELECT all users who are going to a show at 13:30.

2.4   Hints

- As with anything, if you get stuck, move on, then go back if you have time.

- Don't spend all night on it!

- Use resources online to solve harder ones - there are solutions to these questions that work with what we've learnt today, but other tools exist in SQL that could make the queries 'better' or 'easier'.