Database Design, CSCI 340, Spring 2016
SQL - Basic, Chapter 6, pages 133-151, Jan. 15

Open putty and login to your katie account.

From within your katie account, we'll all access the same MySQL account.
  username: csci340User
  password: csci340Pass

Login to MySQL via the command:
  mysql -u csci340User -p
  Enter password: csci340Pass

Useful SQL commands:
  SHOW DATABASES;   // View a list of available databases
  USE HotelReservations; // Move into the HotelReservations database
  SHOW TABLES;   // View list of tables in current database
  DESCRIBE tableName;   // View "intension" of a specified table

  SELECT *
  FROM tableName;   // View "extension" of a table

Write the SQL commands to do the following:
1. Give the name of all databases to which csci340User has access.
     SHOW DATABASES;

2. Move into, use, the HotelReservations database.
     USE HotelReservations;

3. Give the name of all tables in the HotelReservations database.
     SHOW TABLES;

4. List the column names and type of all columns in the Hotel table.
     DECRIBE Hotel;

5. List the full details of all hotels. (The result will contain 8 records.)
     SELECT *
     FROM Hotel;

6. List the full details of all hotels in Butte. (The result will contain 4 records.)
> SELECT *
> FROM Hotel
> WHERE city='Butte';

7. List the name and addresses of all guests. (The result will contain 5 records.)
> SELECT guestName, guestAddress
> FROM Guest;

8. List the name and addresses of all guests whose first name starts with M. (The result will contain 2 records.)
> SELECT guestName, guestAddress
> FROM Guest
> WHERE guestName LIKE 'M%';

9. List the name and addresses of all guests whose address contains the word Columbia. (The result will contain 1 record.)
> SELECT guestName, guestAddress
> FROM Guest
> WHERE guestAddress LIKE '%Columbia%';

10. List all hotel names. (The result will contain 8 records.)
> SELECT hotelName
> FROM Hotel;

11. List all hotel names but remove duplicates. (The result will contain 6 records.)
> SELECT DISTINCT hotelName
> FROM Hotel;

12. List hotel names alphabetically but remove duplicates.

```
SELECT DISTINCT hotelName
FROM Hotel
ORDER BY hotelName;
```

13. List hotel names and the city in which they are located, alphabetically by city and within a city, alphabetically by hotel name and without duplicates. (The result will contain 7 records.)

```
SELECT DISTINCT hotelName, city
FROM Hotel
ORDER BY city, hotelName;
```

14. List the distinct prices and types of all hotel rooms in the database. (The result will contain 12 records.)

```
SELECT DISTINCT price, type
FROM Room
ORDER BY price;
```

15. List the average price of all rooms in the database. (The average price is 122.291667)

```
SELECT AVG(price)
FROM Room;
```

16. List the average price of all rooms in the database but make the column name read "Average Room Price". The result should look like the following:

```
+-------------------+
| Average Room Price |
+-------------------+
|         122.291667 |
+-------------------+
1 row in set (0.00 sec)
```

       SELECT AVG(price) AS "Average Room Price"
       FROM Room;

17. Write a query that tells the number of distinct hotels in the database. (The three hotels in the database called Holiday Inn should be counted as different.) Make the column heading be: # of hotels.

       There are 8 such hotels.

       SELECT COUNT(hotelName) AS '# of hotels'
       FROM Hotel;

18. Write a query that tells the number of distinct hotel names in the database. (This time the three hotels called Holiday Inn should not be counted as different.) Make the column heading be: # of hotels.

       There are 6 such hotels

       SELECT COUNT(DISTINCT hotelName) AS '# of hotels'
       FROM Hotel;

19. Write two queries which list the name, address and city of all hotels which are in Butte or Missoula. Order these by the hotel name. Note that your queries must work for any extension of the database, not just the current extension. (Both results will contain 7 records.)

> SELECT hotelName, street, city
> FROM Hotel
> WHERE city='Butte' OR city='Missoula'
> ORDER BY hotelName;
>
> and
>
> SELECT hotelName, street, city
> FROM Hotel
> WHERE city IN ('Butte','Missoula')
> ORDER BY hotelName;

Challenge:
Do question 16 again but format the result so that it looks like a dollar amount.

> SELECT CONCAT('$', ROUND(AVG(price),2))
>   AS "Average Room Price"
> FROM Room;