SQL – Relational Assignment

Before you begin please do the following:

1. Watch the video lecture posted. Many of the commands I will show you on the video.
2. You can also look at the SQLCheatsheet under "Resources" for many of the common SQL commands.
3. The below exercises corresponds to the lecture titled Relational Model.

Submission:

You must create a SQL file with all your answers: The SQL file should have your name in it as a comment (using #) on the first line. The following line should have your "use" command preceded by a "#" comment. The next line will contain the date and number that results from: **`SELECT @d:=NOW(),SHA(@d);`** separated by a space or TAB. This line should also be preceded by a "#" comment, followed by a blank line.

Then, for every answer, first write a "#" followed by the question number and in the following lines write your answer. Then, skip a blank line and repeat the process for the next question. For example, a file from John Doe with 2 questions, where the first question has 2 items, may look like this:

```
# John Doe
#use jdoe_db;
#2017-01-05 17:38:27  41860934238f8b3a1fdd51dee11e2d4219c92068

#1.1
Some SQL code goes
Here as the answer to
The question. SELECTs, CREATEs, whatever.

#1.2
More SQL code answering
Question number 1.2 (or I.2)
The answer can take up as many lines as needed in SQL
-- non sql line to explain a result (if asked)
-- should be commented with '--'

#2.1
Yet more SQL commands
That answer
The first part of the second question
```

If you want to add additional comments, use the -- notation. You should be able to load your entire SQL file into your IDE, uncomment the second line and run your code without **syntax** errors, at least, before submitting.
You will need to submit the SQL file. **Other formats will not be graded**. The SQL file ends in **.sql** you can choose whatever name you like WITHOUT spaces. A good idea is your last and first name and homework number. For example: DoeJohnHW2.sql

Exercise:

I Primary Keys

1. Write the statements to **drop** the 2 tables: Booking and Guest.
2. Modify your Create Table statements (from the SQL Basics assignment) to use a **primary key** for each table. In addition the **default** price in Table Booking should be set to 500. Paste the 2 new create statements below.
   Hint: One of them is a composite key!
3. Rerun your exact same insert statements (from the first homework on SQL) to insert the rows in the 2 tables[1]. Did they work (yes or no)? If not, explain what happened and fix your errors (paste only the updated inserts below).
4. Write the SQL statement to insert a row into the Guest table where the guestNo is NULL. *Note: Yes, specify the word NULL instead of a number.*
5. Describe what happened. Was there an error? If so, paste it.[2] If there was an error, your description cannot contain any of the words in the error (this will help you articulate your ideas)
6. Write the SQL to insert a row into Guest where the guestno is 101.
7. Describe what happened. Was there an error ? If so, paste it.[2] If there was an error, your description cannot contain any of the words in the error (this will help you articulate your ideas)
8. Write a statement to insert into Booking, guest number (105) and **do not** specify the price. ***Note****: In the insert specify the columns that you do want to add.*
9. Check out what value got entered into the price. Paste the command used. What happened? Describe it in one line (as a comment).

II Create another Table

[1] If your insert statements were incorrect on the first homework, please correct them before answering.

[2] You must paste an error using -- , that is, paste it as a comment.

Since subletting your basement was so successful you decide to purchase additional properties to sublet.

1. Please write the correct statement to create the following table, choose a **primary key**:

    a. Table **Room** with the following fields:
        i. *roomNo (*a unique integer)
        ii. *address*
        iii. *city*
        iv. *state*
        v. *zipcode*
        vi. *aptNo (can have characters in it)*

2. Insert 3 rows into the above table. The first is roomNo 1 which is the address info for your current place that you have been subletting in the previous assignments (you can use your address or any of your choosing). Insert 2 more addresses with room numbers 2 and 3, for the two new. Paste the SQL commands to do this.

III Alter

1. Write a statement to insert into Room a new entry with roomNo 4, without specifying the apt number. You will need to specify the columns you are inserting values into.
2. Write the statement to alter the table Room by specifying a default aptNo. Remember you will need single quotes around the aptNo.
3. Write a statement to insert another row into Room with roomNo 5, without specifying an aptNo.
4. What happened? What is now in the aptNo column? Write your answer as a comment
5. Write the statement to drop the column aptNo from the Room table.
6. Write **one** statement to alter the table Booking to add a column, roomNo, which is an integer representing the roomNo for each booking. The default value should be set to 1 (i.e., your basement that all guests had previously been subletting).
7. Check out the roomNo attribute (paste your command). What happened (write in a comment)?
8. Guest 104 would like to sublet one of your new properties, room number 2 from June 1, 2014 until June 30, 2014 for a price of 650. Write an insert statement.