

Project Title:
Hotel Management

Members Name:
Jennie Yang,
Jooyul Yoon,
Tiffany Adrienne Phan

Team Name:
DBMSgame

3. Database Schema:

```
DROP TABLE IF EXISTS Guest;
CREATE TABLE Guest(
    guestID int,
    guestName text,
    Is_Active bool,
    hotelID int,
    roomID int,
    updatedAt TIMESTAMP NOT NULL DEFAULT NOW() ON UPDATE NOW(),
    PRIMARY KEY(guestID),
    FOREIGN KEY(hotelID) REFERENCES Hotel(hotelID)
);
```

```
DROP TABLE IF EXISTS Country;
CREATE TABLE Country(
    countryID varchar(50),
    countryName varchar(50),
    PRIMARY KEY (countryID)
);
```

```
DROP TABLE IF EXISTS City;
CREATE TABLE City(
    cityName varchar(50),
    countryID varchar(50),
    PRIMARY KEY(cityName),
    FOREIGN KEY(countryID) REFERENCES Country(countryID)
);
```

```
DROP TABLE IF EXISTS Hotel;
CREATE TABLE Hotel(
    hotelID int,
    hotelName varchar(50),
    cityName varchar(50),
    starsRating int,
    PRIMARY KEY(hotelID),
    FOREIGN KEY(cityName) REFERENCES City(cityName)
);
```

```
DROP TABLE IF EXISTS Room
CREATE TABLE Room(
```

```

    hotelID int,
    roomID int,
    roomType varchar(50),
    roomStatus varchar(50),
    price int not null,
    PRIMARY KEY(roomID),
    FOREIGN KEY(hotelID) REFERENCES Hotel (hotelID)
);

```

Archive:

```

DROP TABLE IF EXISTS Archive;
CREATE TABLE Archive(

```

```

    guestID int,
    guestName text,
    Is_Active bool,
    HotelID int,
    RoomID int,
    updatedAt date,
    PRIMARY KEY(guestID)

```

```
);

```

4. Population of Database with Real Data from a Public Dataset:

```

insert into Guest values(101, 'Ann', 0, 7, 101, NOW());
insert into Guest values(102, 'James', 1, 2, 105, NOW());
insert into Guest values(103, 'Betty', 1, 1, 105, NOW());
insert into Guest values(104, 'Inez', 1, 3, 105, NOW());
insert into Guest values(105, 'Taylor', 1, 6, 103, NOW());
insert into Guest values(106, 'Este', 0, 2, 107, NOW());
insert into Guest values(107, 'May', 1, 5, 102, NOW());
insert into Guest values(108, 'Dean', 0, 11, 110, NOW());

```

```

/*
used information from: https://datahub.io/core/country-list
*/

```

```

insert into Country values('AT', 'Austria');
insert into Country values('BR', 'Brazil');
insert into Country values('CA', 'Canada');
insert into Country values('CN', 'China');
insert into Country values('HK', 'HongKong');
insert into Country values('JP', 'Japan');

```

```
insert into Country values('US', 'United States');

/*
used information from: https://datahub.io/core/world-cities
*/
insert into City values('Vienna', 'AT');
insert into City values('Salzburg', 'AT');
insert into City values('Perth', 'BR');
insert into City values('Patos', 'BR');
insert into City values('Toronto', 'CA');
insert into City values('Montreal', 'CA');
insert into City values('Beijing', 'CN');
insert into City values('Shanghai', 'CN');
insert into City values('Victoria', 'HK');
insert into City values('Tokyo', 'JP');
insert into City values('Osaka', 'JP');
insert into City values('California', 'US');
insert into City values('NewYork', 'US');

insert into Hotel values(1, 'Marriott', 'Vienna', 5);
insert into Hotel values(2, 'Hyatt', 'Salzburg', 4);
insert into Hotel values(3, 'Four Seasons', 'Perth', 2);
insert into Hotel values(4, 'Ritz-Carlton Hotel', 'Patos', 2);
insert into Hotel values(5, 'Marriott', 'Toronto', 5);
insert into Hotel values(6, 'Hyatt', 'Montreal', 3);
insert into Hotel values(7, 'Four Seasons', 'Beijing', 2);
insert into Hotel values(8, 'Four Seasons', 'Shanghai', 1);
insert into Hotel values(9, 'Ritz-Carlton Hotel', 'Victoria', 3);
insert into Hotel values(10, 'Ritz-Carlton Hotel', 'Tokyo', 3);
insert into Hotel values(11, 'Marriott', 'Osaka', 5);
insert into Hotel values(12, 'Hyatt', 'NewYork', 4);
insert into Hotel values(13, 'Four Seasons', 'California', 2);
insert into Hotel values(14, 'Ritz-Carlton Hotel', 'Salzburg', 2);

insert into Room values(1, 101, 'suite', 'booked', 300);
insert into Room values(2, 102, 'single', 'empty', 100);
insert into Room values(3, 103, 'suite', 'booked', 50);
insert into Room values(4, 104, 'single', 'empty', 80);
insert into Room values(5, 105, 'suite', 'booked', 350);
insert into Room values(6, 106, 'single', 'empty', 70);
```

```

insert into Room values(7, 107, 'suite', 'booked', 120);
insert into Room values(7, 108, 'single', 'empty', 100);
insert into Room values(8, 109, 'suite', 'booked', 250);
insert into Room values(8, 110, 'single', 'empty', 160);
insert into Room values(9, 111, 'suite', 'booked', 300);
insert into Room values(9, 112, 'single', 'empty', 120);
insert into Room values(10, 113, 'suite', 'booked', 180);
insert into Room values(12, 114, 'single', 'empty', 80);
insert into Room values(13, 115, 'suite', 'booked', 110);
insert into Room values(14, 116, 'single', 'empty', 90);
insert into Room values(11, 117, 'suite', 'booked', 140);
insert into Room values(12, 118, 'single', 'empty', 150);
insert into Room values(13, 119, 'suite', 'booked', 370);
insert into Room values(14, 120, 'single', 'empty', 95);

```

5. Description to show relation schemas are in BCNF or in 3NF:art

Guest {guestID}+ -> {guestID, guestName, Is_Active, HotelID}

Country {countryID }+ ->{countryID, countryName}

City {cityName}+ ->{cityName, countryID}

Hotel {hotelID}+ -> {hotelID, hotelName, cityName, starsRating}

Room {HotelID}+ -> {HotelID, roomID - KEY, roomtype, roomStatus, Price)

There are no BCNF violations in schemas.

6. Screenshots of Relations after initial population of data:

Guest Relation:

```

mysql> select * from Guest;
+-----+-----+-----+-----+-----+-----+
| guestID | guestName | Is_Active | HotelID | RoomID | updatedAt |
+-----+-----+-----+-----+-----+-----+
|    101 | Ann      |      0 |     0 |    101 | 2021-11-11 |
|    102 | James    |      1 |     2 |    105 | 2021-11-11 |
|    103 | Betty    |      1 |     1 |    105 | 2021-11-11 |
|    104 | Inez    |      1 |     3 |    105 | 2021-11-11 |
|    105 | Taylor   |      1 |     6 |    103 | 2021-11-11 |
|    106 | Este     |      0 |     0 |    107 | 2021-11-11 |
|    107 | May      |      1 |     5 |    102 | 2021-11-11 |
|    108 | Dean     |      0 |     0 |    110 | 2021-11-11 |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)

```

Country relation:

```
[mysql]> select * from Country;
+-----+-----+
| countryID | countryName |
+-----+-----+
| AT        | Austria      |
| BR        | Brazil       |
| CA        | Canada       |
| CN        | China        |
| HK        | HongKong    |
| JP        | Japan        |
| US        | United States|
+-----+-----+
7 rows in set (0.00 sec)
```

City Relation:

```
[mysql]> select * from City;
+-----+-----+
| cityName   | countryID |
+-----+-----+
| Salzburg   | AT        |
| Vienna     | AT        |
| Patos      | BR        |
| Perth      | BR        |
| Montreal   | CA        |
| Toronto    | CA        |
| Beijing    | CN        |
| Shanghai   | CN        |
| Victoria   | HK        |
| Osaka      | JP        |
| Tokyo      | JP        |
| California | US        |
| NewYork   | US        |
+-----+-----+
13 rows in set (0.00 sec)
```

Hotel Relation:

```
[mysql] > select * from Hotel;
+-----+-----+-----+-----+
| hotelID | hotelName          | cityName | starsRating |
+-----+-----+-----+-----+
|      1 | Marriott           | Vienna   |      5 |
|      2 | Hyatt               | Salzburg |      4 |
|      3 | Four Seasons        | Perth    |      2 |
|      4 | Ritz-Carlton Hotel | Patos    |      2 |
|      5 | Marriott           | Toronto  |      5 |
|      6 | Hyatt               | Montreal |      3 |
|      7 | Four Seasons        | Beijing  |      2 |
|      8 | Four Seasons        | Shanghai |      1 |
|      9 | Ritz-Carlton Hotel | Victoria |      3 |
|     10 | Ritz-Carlton Hotel | Tokyo    |      3 |
|     11 | Marriott           | Osaka   |      5 |
|     12 | Hyatt               | NewYork |      4 |
|     13 | Four Seasons        | California |      2 |
|     14 | Ritz-Carlton Hotel | Salzburg |      2 |
+-----+-----+-----+-----+
14 rows in set (0.00 sec)
```

Room Relation:

```
[mysql] > select * from Room;
+-----+-----+-----+-----+-----+
| hotelID | roomID | roomType | roomStatus | price |
+-----+-----+-----+-----+-----+
|      1 |    101 | suite    | booked    | 300  |
|      2 |    102 | single   | empty    | 100  |
|      3 |    103 | suite    | booked    | 50   |
|      4 |    104 | single   | empty    | 80   |
|      5 |    105 | suite    | booked    | 350  |
|      6 |    106 | single   | empty    | 70   |
|      7 |    107 | suite    | booked    | 120  |
|      7 |    108 | single   | empty    | 100  |
|      8 |    109 | suite    | booked    | 250  |
|      8 |    110 | single   | empty    | 160  |
|      9 |    111 | suite    | booked    | 300  |
|      9 |    112 | single   | empty    | 120  |
|     10 |    113 | suite    | booked    | 180  |
|     12 |    114 | single   | empty    | 80   |
|     13 |    115 | suite    | booked    | 110  |
|     14 |    116 | single   | empty    | 90   |
|     11 |    117 | suite    | booked    | 140  |
|     12 |    118 | single   | empty    | 150  |
|     13 |    119 | suite    | booked    | 370  |
|     14 |    120 | single   | empty    | 95   |
+-----+-----+-----+-----+
20 rows in set (0.00 sec)
```

Archive Relation:

```
[mysql> select * from Archive;
Empty set (0.00 sec)
```

7/9. Supportive Triggers, Procedures, Fifteen Functions (use cases), Archiving, Key Constraint and Foreign Key Constraint Violations, and Screenshots:

Triggers:

```
/*
```

Trigger 1:

Automatically deletes guests when their status becomes inactive / when they check out.

```
*/
```

```
CREATE TRIGGER deleteGuests
```

```
AFTER UPDATE ON Guest
```

```
FOR EACH ROW
```

```
    DELETE FROM Guest WHERE Is_Active = false;
```

```
/*
```

Trigger 2:

When a guest moves to another room, everyone in their room moves as well

```
*/
```

```
CREATE TRIGGER moveGuests
```

```
AFTER UPDATE ON Guest
```

```
FOR EACH ROW
```

```
    UPDATE Guest
```

```
        SET roomID = new.roomID
```

```
        WHERE guestID IN (SELECT guestID2 FROM Guest g2 WHERE g2.roomID =
old.roomID);
```

```
/*
```

Trigger 3:

Automatically updates guest's room attribute to null when the guest's is_active status changes to 0

```
*/
```

```
CREATE TRIGGER updateGuestRoom
```

```
AFTER UPDATE ON Guest
FOR EACH ROW
    UPDATE Guest
        SET roomID = NULL WHERE is_Active = false;
```

Procedures:

```
/* Obtain guest list */
DROP PROCEDURE IF EXISTS getGuestList;
DELIMITER //
CREATE PROCEDURE getGuestList()
BEGIN
    SELECT guestID
        FROM Guest
        WHERE Is_Active = True;
END //
DELIMITER ;
```

Fifteen Functions:

1. Show the available Hotel with the cityName and CountryName

```
1 • select hotelName, Hotel.cityName, countryName
2   from Hotel, Country, City
3   where Hotel.hotelID in
4     (select hotelID
5      from Room
6      where roomStatus = 'empty') and
7        Hotel.cityName = City.cityName and
8        City.countryID = Country.countryID
9    ;
```

00% 2:9 |

Result Grid Filter Rows: Search Export:

hotelName	cityName	countryName
Hyatt	Salzburg	Austria
Ritz-Carlton Hotel	Patos	Brazil
Hyatt	Montreal	Canada
Four Seasons	Beijing	China
Four Seasons	Shanghai	China
Ritz-Carlton Hotel	Victoria	HongKong
Hyatt	NewYork	United States
Ritz-Carlton Hotel	Salzburg	Austria

1

Available Hotel

Hotel	City	Country
Hyatt	Salzburg	Austria
Ritz-Carlton Hotel	Patos	Brazil
Hyatt	Montreal	Canada
Four Seasons	Beijing	China
Four Seasons	Shanghai	China
Ritz-Carlton Hotel	Victoria	HongKong
Hyatt	NewYork	United States
Ritz-Carlton Hotel	Salzburg	Austria

2. Show the hotel name, cityName booked by the user

```
1 • select hotelName, guestName, City.cityName  
2   from Guest, Hotel, City  
3   where Guest.hotelID = Hotel.HotelID and  
4     City.cityName = Hotel.cityName;  
5  
6
```

6
100% ◇ 1:7

Result Grid Filter Rows: Search Export:

	hotelName	guestName	cityName	
▶	Four Seasons	Ann	Beijing	
◀	Hyatt	James	Salzburg	
◀	Marriott	Betty	Vienna	
▶	Four Seasons	Inez	Perth	
◀	Hyatt	Taylor	Montreal	
◀	Hyatt	Este	Salzburg	
◀	Marriott	May	Toronto	
◀	Marriott	Dean	Osaka	

2

Guest	Hotel	City
Ann	Four Seasons	Beijing
James	Hyatt	Salzburg
Betty	Marriott	Vienna
Inez	Four Seasons	Perth
Taylor	Hyatt	Montreal
Este	Hyatt	Salzburg
May	Marriott	Toronto
Dean	Marriott	Osaka

3. Show the available Hotel with rating over 3

```
1 •  select distinct hotelName, price, City.cityName
2      from Hotel inner join Room on Hotel.hotelID = Room.hotelID,
3          City
4      where starsRating >= 3 and
5          roomStatus = 'empty' and
6          Hotel.cityName = City.cityName
7      ;
```

100% 31:6

Result Grid Filter Rows: Search Export:

hotelName	price	cityName
Hyatt	100	Salzburg
Hyatt	70	Montreal
Ritz-Carlton Hotel	120	Victoria
Hyatt	80	NewYork
Hyatt	150	NewYork

3

Available Hotel Over 3 stars

Hotel	Price	City
Hyatt	100	Salzburg
Hyatt	70	Montreal
Ritz-Carlton Hotel	120	Victoria
Hyatt	80	NewYork
Hyatt	150	NewYork

4. Show the best rating Hotel and cityName, countryName

```
1 •  select distinct hotelName, Hotel.cityName, Country.countryName
2      from Hotel inner join City on Hotel.cityName = City.cityName
3      inner join Country on Country.countryID = City.countryID
4      where starsRating >= all (select starsRating from Hotel)
5      ;
```

100% ▼ 63:1

Result Grid Filter Rows: Search Export: grid

hotelName	cityName	countryName
Marriott	Vienna	Austria
Marriott	Toronto	Canada
Marriott	Osaka	Japan

Best Rated Hotel

Hotel	City	Country
Marriott	Vienna	Austria
Marriott	Toronto	Canada
Marriott	Osaka	Japan

5. Show the most expensive hotel and cityName

```
1 •  select hotelName, Hotel.cityName
2   from Hotel inner join Room on Hotel.hotelID = Room.hotelID
3   inner join City on City.cityName = Hotel.cityName
4   inner join Country on City.countryID = Country.countryID
5   where price >= all
6   (select price from Hotel inner join Room on Hotel.hotelID = Room.hotelID
7   inner join City on City.cityName = Hotel.cityName
8   inner join Country on City.countryID = Country.countryID);
9 ;
```

100% 2:9

Result Grid Filter Rows: Search Export:

hotelName	cityName
Four Seasons	California

5

Most expensive hotel

Hotel	City	Country
Four Seasons	California	United States

6. Show the cheapest hotel and cityName

```
1 •  select hotelName, Hotel.cityName
2   from Hotel inner join Room on Hotel.hotelID = Room.hotelID
3   inner join City on City.cityName = Hotel.cityName
4   inner join Country on City.countryID = Country.countryID
5   where price <= all
6   (select price from Hotel inner join Room on Hotel.hotelID = Room.hotelID
7   inner join City on City.cityName = Hotel.cityName
8   inner join Country on City.countryID = Country.countryID);
9 ;
```

100% 2:9

Result Grid Filter Rows: Search Export:

hotelName	cityName
Four Seasons	Perth

6

Cheapest hotel

Hotel	City	Country
Four Seasons	Perth	Brazil

7. Show the worst rating Hotel and cityName, price

```
1 •   select hotelName, price, Hotel.cityName
2     from Hotel inner join Room on Hotel.hotelID = Room.hotelID
3       inner join City on City.cityName = Hotel.cityName
4       inner join Country on City.countryID = Country.countryID
5     where starsRating <= all (select starsRating from Hotel)
6   ;
```

100% 32:1

Result Grid Filter Rows: Search Export:

hotelName	price	cityName
Four Seasons	250	Shanghai
Four Seasons	160	Shanghai

Worst Rating

```
+-----+
|      Hotel |    Price |      City |    Country |
+-----+
| Four Seasons |    250 | Shanghai |    China |
| Four Seasons |    160 | Shanghai |    China |
+-----+
```

8. Sort hotel by the highest price

```
1 •  select hotelName, price, Hotel.cityName, countryName
2   from Hotel inner join City on Hotel.cityName = City.cityName
3   inner join Country on City.countryID = Country.countryID
4   inner join Room on Room.hotelID = Hotel.hotelID
5   order by price desc;
```

100% 20:5

Result Grid Filter Rows: Search Export:

hotelName	price	cityName	countryName
Four Seasons	370	California	United States
Marriott	350	Toronto	Canada
Marriott	300	Vienna	Austria
Ritz-Carlton Hotel	300	Victoria	HongKong
Four Seasons	250	Shanghai	China
Ritz-Carlton Hotel	180	Tokyo	Japan
Four Seasons	160	Shanghai	China
Hyatt	150	NewYork	United States
Marriott	140	Osaka	Japan
Four Seasons	120	Beijing	China
Ritz-Carlton Hotel	120	Victoria	HongKong
Four Seasons	110	California	United States
Hyatt	100	Salzburg	Austria
Four Seasons	100	Beijing	China
Ritz-Carlton Hotel	95	Salzburg	Austria
Ritz-Carlton Hotel	90	Salzburg	Austria
Ritz-Carlton Hotel	80	Patos	Brazil
Hyatt	80	NewYork	United States
Hyatt	70	Montreal	Canada
Four Seasons	50	Perth	Brazil

8

Sorted by Highest price

Hotel	Price	City	Country
Four Seasons	370	California	United States
Marriott	350	Toronto	Canada
Marriott	300	Vienna	Austria
Ritz-Carlton Hotel	300	Victoria	HongKong
Four Seasons	250	Shanghai	China
Ritz-Carlton Hotel	180	Tokyo	Japan
Four Seasons	160	Shanghai	China
Hyatt	150	NewYork	United States
Marriott	140	Osaka	Japan
Four Seasons	120	Beijing	China
Ritz-Carlton Hotel	120	Victoria	HongKong
Four Seasons	110	California	United States
Hyatt	100	Salzburg	Austria
Four Seasons	100	Beijing	China
Ritz-Carlton Hotel	95	Salzburg	Austria
Ritz-Carlton Hotel	90	Salzburg	Austria
Ritz-Carlton Hotel	80	Patos	Brazil
Hyatt	80	NewYork	United States
Hyatt	70	Montreal	Canada
Four Seasons	50	Perth	Brazil

9. Sort hotel by the highest price

```
1 •  select hotelName, price, Hotel.cityName, countryName
2      from Hotel inner join City on Hotel.cityName = City.cityName
3      inner join Country on City.countryID = Country.countryID
4      inner join Room on Room.hotelID = Hotel.hotelID
5      order by price asc;
```

100% 20:5

Result Grid Filter Rows: Search Export:

hotelName	price	cityName	countryName
Four Seasons	50	Perth	Brazil
Hyatt	70	Montreal	Canada
Ritz-Carlton Hotel	80	Patos	Brazil
Hyatt	80	NewYork	United States
Ritz-Carlton Hotel	90	Salzburg	Austria
Ritz-Carlton Hotel	95	Salzburg	Austria
Hyatt	100	Salzburg	Austria
Four Seasons	100	Beijing	China
Four Seasons	110	California	United States
Four Seasons	120	Beijing	China
Ritz-Carlton Hotel	120	Victoria	HongKong
Marriott	140	Osaka	Japan
Hyatt	150	NewYork	United States
Four Seasons	160	Shanghai	China
Ritz-Carlton Hotel	180	Tokyo	Japan
Four Seasons	250	Shanghai	China
Marriott	300	Vienna	Austria
Ritz-Carlton Hotel	300	Victoria	HongKong
Marriott	350	Toronto	Canada
Four Seasons	370	California	United States

9

Sorted by Lowest price

	Hotel	Price	City	Country
	Four Seasons	50	Perth	Brazil
	Hyatt	70	Montreal	Canada
	Ritz-Carlton Hotel	80	Patos	Brazil
	Hyatt	80	NewYork	United States
	Ritz-Carlton Hotel	90	Salzburg	Austria
	Ritz-Carlton Hotel	95	Salzburg	Austria
	Hyatt	100	Salzburg	Austria
	Four Seasons	100	Beijing	China
	Four Seasons	110	California	United States
	Four Seasons	120	Beijing	China
	Ritz-Carlton Hotel	120	Victoria	HongKong
	Marriott	140	Osaka	Japan
	Hyatt	150	NewYork	United States
	Four Seasons	160	Shanghai	China
	Ritz-Carlton Hotel	180	Tokyo	Japan
	Four Seasons	250	Shanghai	China
	Marriott	300	Vienna	Austria
	Ritz-Carlton Hotel	300	Victoria	HongKong
	Marriott	350	Toronto	Canada
	Four Seasons	370	California	United States

10. Sort hotel by the highest Rating

```
1 •   select hotelName, starsRating, Hotel.cityName, countryName
2     from Hotel inner join City on Hotel.cityName = City.cityName
3     inner join Country on City.countryID = Country.countryID
4     inner join Room on Room.hotelID = Hotel.hotelID
5     order by starsRating desc;
```

100% 26:5

Result Grid Filter Rows: Search Export:

hotelName	starsRating	cityName	countryName
Marriott	5	Vienna	Austria
Marriott	5	Toronto	Canada
Marriott	5	Osaka	Japan
Hyatt	4	Salzburg	Austria
Hyatt	4	NewYork	United States
Hyatt	4	NewYork	United States
Hyatt	3	Montreal	Canada
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Tokyo	Japan
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Patos	Brazil
Four Seasons	2	Perth	Brazil
Four Seasons	2	Beijing	China
Four Seasons	2	Beijing	China
Four Seasons	2	California	United States
Four Seasons	2	California	United States
Four Seasons	1	Shanghai	China
Four Seasons	1	Shanghai	China

10

Sorted by Highest Rating

```
+-----+
|       Hotel | Stars |      City |      Country |
+-----+
| Marriott | 5 | Vienna | Austria |
| Marriott | 5 | Toronto | Canada |
| Marriott | 5 | Osaka | Japan |
| Hyatt | 4 | Salzburg | Austria |
| Hyatt | 4 | NewYork | United States |
| Hyatt | 4 | NewYork | United States |
| Hyatt | 3 | Montreal | Canada |
| Ritz-Carlton Hotel | 3 | Victoria | HongKong |
| Ritz-Carlton Hotel | 3 | Victoria | HongKong |
| Ritz-Carlton Hotel | 3 | Tokyo | Japan |
| Ritz-Carlton Hotel | 2 | Salzburg | Austria |
| Ritz-Carlton Hotel | 2 | Salzburg | Austria |
| Ritz-Carlton Hotel | 2 | Patos | Brazil |
| Four Seasons | 2 | Perth | Brazil |
| Four Seasons | 2 | Beijing | China |
| Four Seasons | 2 | Beijing | China |
| Four Seasons | 2 | California | United States |
| Four Seasons | 2 | California | United States |
| Four Seasons | 1 | Shanghai | China |
| Four Seasons | 1 | Shanghai | China |
+-----+
```

11. Sort hotel by the lowest Rating

```
1 •  select hotelName, starsRating, Hotel.cityName, countryName
2    from Hotel inner join City on Hotel.cityName = City.cityName
3    inner join Country on City.countryID = Country.countryID
4    inner join Room on Room.hotelID = Hotel.hotelID
5    order by starsRating asc;
```

100% 25:5

Result Grid Filter Rows: Search Export:

hotelName	starsRating	cityName	countryName
Four Seasons	1	Shanghai	China
Four Seasons	1	Shanghai	China
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Patos	Brazil
Four Seasons	2	Perth	Brazil
Four Seasons	2	Beijing	China
Four Seasons	2	Beijing	China
Four Seasons	2	California	United States
Four Seasons	2	California	United States
Hyatt	3	Montreal	Canada
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Tokyo	Japan
Hyatt	4	Salzburg	Austria
Hyatt	4	NewYork	United States
Hyatt	4	NewYork	United States
Marriott	5	Vienna	Austria
Marriott	5	Toronto	Canada
Marriott	5	Osaka	Japan

11

Sorted by Lowest Rating

Hotel	Stars	City	Country
Four Seasons	1	Shanghai	China
Four Seasons	1	Shanghai	China
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Salzburg	Austria
Ritz-Carlton Hotel	2	Patos	Brazil
Four Seasons	2	Perth	Brazil
Four Seasons	2	Beijing	China
Four Seasons	2	Beijing	China
Four Seasons	2	California	United States
Four Seasons	2	California	United States
Hyatt	3	Montreal	Canada
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Victoria	HongKong
Ritz-Carlton Hotel	3	Tokyo	Japan
Hyatt	4	Salzburg	Austria
Hyatt	4	NewYork	United States
Hyatt	4	NewYork	United States
Marriott	5	Vienna	Austria
Marriott	5	Toronto	Canada
Marriott	5	Osaka	Japan

12. Show the hotels whose Rating is over 3 and price is under 200.

```
1 •  select distinct hotelName, Hotel.cityName, starsRating, price
2   from Hotel inner join City on Hotel.cityName = City.cityName
3   inner join Country on City.countryID = Country.countryID
4   inner join Room on Room.hotelID = Hotel.hotelID
5   where starsRating >= 3 and
6   price <= 200;
```

100% 14:6

Result Grid Filter Rows: Search Export:

hotelName	cityName	starsRating	price
Hyatt	Salzburg	4	100
Hyatt	Montreal	3	70
Ritz-Carlton Hotel	Victoria	3	120
Ritz-Carlton Hotel	Tokyo	3	180
Marriott	Osaka	5	140
Hyatt	NewYork	4	80
Hyatt	NewYork	4	150

12

Hotels < \$200 and > 3 stars

Hotel	City	Stars	Price
Hyatt	Salzburg	4	100
Hyatt	Montreal	3	70
Ritz-Carlton Hotel	Victoria	3	120
Ritz-Carlton Hotel	Tokyo	3	180
Marriott	Osaka	5	140
Hyatt	NewYork	4	80
Hyatt	NewYork	4	150

13. Show the hotels with starsRating is under 2 or the price is over 200

```
1 • (select distinct hotelName, Hotel.cityName
2   from Hotel inner join City on Hotel.cityName = City.cityName
3   inner join Country on City.countryID = Country.countryID
4   inner join Room on Room.hotelID = Hotel.hotelID
5   where price >= 200) union all
6 (select distinct hotelName, Hotel.cityName
7   from Hotel inner join City on Hotel.cityName = City.cityName
8   inner join Country on City.countryID = Country.countryID
9   inner join Room on Room.hotelID = Hotel.hotelID
10  where starsRating <= 2)
11 ;
```

100% 18:6

Result Grid Filter Rows: Search Export:

hotelName	cityName
Marriott	Vienna
Marriott	Toronto
Four Seasons	Shanghai
Ritz-Carlton Hotel	Victoria
Four Seasons	California
Four Seasons	Perth
Ritz-Carlton Hotel	Patos
Four Seasons	Beijing
Four Seasons	Shanghai
Four Seasons	California
Ritz-Carlton Hotel	Salzburg

13

Hotels > \$200 or < 2 stars

Hotel	City	Stars	Price
Marriott	Vienna	5	300
Marriott	Toronto	5	350
Four Seasons	Shanghai	1	250
Ritz-Carlton Hotel	Victoria	3	300
Four Seasons	California	2	370
Four Seasons	Perth	2	50
Ritz-Carlton Hotel	Patos	2	80
Four Seasons	Beijing	2	120
Four Seasons	Beijing	2	100
Four Seasons	Shanghai	1	250
Four Seasons	Shanghai	1	160
Four Seasons	California	2	110
Four Seasons	California	2	370
Ritz-Carlton Hotel	Salzburg	2	90
Ritz-Carlton Hotel	Salzburg	2	95

14. Show the hotelName, total price, countryName if stays 3 days

```
1 •  select distinct hotelName, Hotel.cityName, countryname, price * 3
2   from Hotel inner join City on Hotel.cityName = City.cityName
3   inner join Country on City.countryID = Country.countryID
4   inner join Room on Room.hotelID = Hotel.hotelID
5   ;
```

100% 2:5

Result Grid Filter Rows: Search Export:

hotelName	cityName	countryname	price * 3
Hyatt	Salzburg	Austria	300
Ritz-Carlton Hotel	Salzburg	Austria	270
Ritz-Carlton Hotel	Salzburg	Austria	285
Marriott	Vienna	Austria	900
Ritz-Carlton Hotel	Patos	Brazil	240
Four Seasons	Perth	Brazil	150
Hyatt	Montreal	Canada	210
Marriott	Toronto	Canada	1050
Four Seasons	Beijing	China	360
Four Seasons	Beijing	China	300
Four Seasons	Shanghai	China	750
Four Seasons	Shanghai	China	480
Ritz-Carlton Hotel	Victoria	HongKong	900
Ritz-Carlton Hotel	Victoria	HongKong	360
Marriott	Osaka	Japan	420
Ritz-Carlton Hotel	Tokyo	Japan	540
Four Seasons	California	United States	330
Four Seasons	California	United States	1110
Hyatt	NewYork	United States	240
Hyatt	NewYork	United States	450

14

Price if stays 3 days

Hotel	City	Country	Type	Price
Hyatt	Salzburg	Austria	single	300
Ritz-Carlton Hotel	Salzburg	Austria	single	270
Ritz-Carlton Hotel	Salzburg	Austria	single	285
Marriott	Vienna	Austria	suite	900
Ritz-Carlton Hotel	Patos	Brazil	single	240
Four Seasons	Perth	Brazil	suite	150
Hyatt	Montreal	Canada	single	210
Marriott	Toronto	Canada	suite	1050
Four Seasons	Beijing	China	suite	360
Four Seasons	Beijing	China	single	300
Four Seasons	Shanghai	China	suite	750
Four Seasons	Shanghai	China	single	480
Ritz-Carlton Hotel	Victoria	HongKong	suite	900
Ritz-Carlton Hotel	Victoria	HongKong	single	360
Marriott	Osaka	Japan	suite	420
Ritz-Carlton Hotel	Tokyo	Japan	suite	540
Four Seasons	California	United States	suite	330
Four Seasons	California	United States	suite	1110
Hyatt	NewYork	United States	single	240
Hyatt	NewYork	United States	single	450

15. Show the hotel that the difference between the highest price and the lowest price is the biggest.

```
1 •   select HID, difference from
2   (select hotelName as HID, max(price) - min(price) as difference
3    from Hotel, Room
4    where Hotel.hotelID = Room.hotelID
5    group by Hotel.hotelID) DHP
6   Where difference >= all (select max(price) - min(price) as difference
7    from Hotel, Room
8    where Hotel.hotelID = Room.hotelID
9    group by Hotel.hotelID)
10   ;
```

100% 23:1

Result Grid Filter Rows: Search Export:

HID	difference
Four Seasons	260

```
15
Hotel with biggest Difference between highest and lowest price
+-----+
|       Hotel | Difference |
+-----+
| Four Seasons |      260 |
+-----+
```

Archiving:

```
mysql> SELECT * FROM Guest;
[+-----+-----+-----+-----+-----+-----+-----+]
| guestID | guestName | Is_Active | HotelID | RoomID | updatedAt |
[+-----+-----+-----+-----+-----+-----+-----+]
|    101 | Ann       |      0   |     7   |    101 | 2021-11-29 23:45:56 |
|    102 | James     |      1   |     2   |    105 | 2021-11-29 23:46:05 |
|    103 | Betty     |      1   |     1   |    105 | 2021-11-29 23:46:07 |
|    104 | Inez      |      1   |     3   |    105 | 2021-11-29 23:46:08 |
|    105 | Taylor    |      1   |     6   |    103 | 2021-11-29 22:53:41 |
|    106 | Este      |      0   |     2   |    107 | 2021-11-29 22:53:42 |
|    107 | May       |      1   |     5   |    102 | 2021-11-29 22:53:44 |
|    108 | Dean      |      0   |    11   |    110 | 2021-11-29 22:53:45 |
[+-----+-----+-----+-----+-----+-----+-----+]
8 rows in set (0.00 sec)

mysql> SELECT * FROM Archive;
[Empty set (0.00 sec)

mysql> CALL getGuestList('2021-11-29 23:00:00');
Query OK, 4 rows affected (0.01 sec)

mysql> SELECT * FROM Guest;
[+-----+-----+-----+-----+-----+-----+-----+]
| guestID | guestName | Is_Active | HotelID | RoomID | updatedAt |
[+-----+-----+-----+-----+-----+-----+-----+]
|    101 | Ann       |      0   |     7   |    101 | 2021-11-29 23:45:56 |
|    102 | James     |      1   |     2   |    105 | 2021-11-29 23:46:05 |
|    103 | Betty     |      1   |     1   |    105 | 2021-11-29 23:46:07 |
|    104 | Inez      |      1   |     3   |    105 | 2021-11-29 23:46:08 |
[+-----+-----+-----+-----+-----+-----+-----+]
4 rows in set (0.00 sec)

mysql> SELECT * FROM Archive;
[+-----+-----+-----+-----+-----+-----+-----+]
| guestID | guestName | Is_Active | HotelID | RoomID | updatedAt |
[+-----+-----+-----+-----+-----+-----+-----+]
|    105 | Taylor    |      1   |     6   |    103 | 2021-11-29 |
|    106 | Este      |      0   |     2   |    107 | 2021-11-29 |
|    107 | May       |      1   |     5   |    102 | 2021-11-29 |
|    108 | Dean      |      0   |    11   |    110 | 2021-11-29 |
[+-----+-----+-----+-----+-----+-----+-----+]
4 rows in set (0.00 sec)
```

```
Archive - please enter timestamp (YYYY-MM-DD HH:MM:SS)
2021-12-02 12:05:50
```

```
Table Guest (OLD)
```

ID	Name	Updated At
101	Ann	2021-12-02 12:05:27
102	James	2021-12-02 12:05:34
103	Betty	2021-12-02 12:05:39
104	Inez	2021-12-02 12:05:40
105	Taylor	2021-12-02 12:05:55
106	Este	2021-12-02 12:05:56
107	May	2021-12-02 12:05:57
108	Dean	2021-12-02 12:05:58

```
Table Archive (OLD)
```

ID	Name	Updated At

```
The procedure getGuestList has been called (2021-12-02 12:05:50)
```

```
Table Guest (NEW)
```

ID	Name	Updated At
105	Taylor	2021-12-02 12:05:55
106	Este	2021-12-02 12:05:56
107	May	2021-12-02 12:05:57
108	Dean	2021-12-02 12:05:58

```
Table Archive (NEW)
```

ID	Name	Updated At
101	Ann	2021-12-02
102	James	2021-12-02
103	Betty	2021-12-02
104	Inez	2021-12-02

Key Constraint and Foreign Key Constraint:

Hotel key Constraint:

```
[mysql> INSERT INTO Hotel Values(1, 'Hyatt', 'Santa Clara', 5);
ERROR 1062 (23000): Duplicate entry '1' for key 'hotel.PRIMARY'
```

Key Constraints

```
Executed Query: INSERT INTO Hotel Values(1, 'Hyatt', 'Shanghai', 5)
Error msg from mysql: ERROR 1062 (23000): Duplicate entry '1' for key 'hotel.PRIMARY'
java.sql.SQLException: Can not issue data manipulation statements with executeQuery().
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:129)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:97)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:89)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:63)
    at com.mysql.cj.jdbc.StatementImpl.checkForDml(StatementImpl.java:367)
    at com.mysql.cj.jdbc.StatementImpl.executeQuery(StatementImpl.java:1137)
    at utils.jdbcUtils.getResultSet(jdbcUtils.java:15)
    at Runner.kConstraints(Runner.java:630)
    at Runner.showOption(Runner.java:68)
    at Runner.main(Runner.java:16)
```

Foreign Key Constraint:

```
[mysql> INSERT INTO Hotel Values(15, 'Hyatt', 'Santa Clara', 5);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('hotel_reservation` .
`hotel`, CONSTRAINT `hotel_ibfk_1` FOREIGN KEY (`cityName`) REFERENCES `city` (`cityName`))
```

Foreign Key Constraints

```
Executed Query: INSERT INTO Hotel Values(15, 'Hyatt', 'Santa Clara', 5)
Error msg from mysql: ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('hotel_reservation` .`hotel`, CONSTRAINT `hotel_ibfk_1` FOREIGN KEY (`cityName`)
) REFERENCES `city` (`cityName`)
java.sql.SQLException: Can not issue data manipulation statements with executeQuery().
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:129)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:97)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:89)
    at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:63)
    at com.mysql.cj.jdbc.StatementImpl.checkForDml(StatementImpl.java:367)
    at com.mysql.cj.jdbc.StatementImpl.executeQuery(StatementImpl.java:1137)
    at utils.jdbcUtils.getResultSet(jdbcUtils.java:15)
    at Runner.fkConstants(Runner.java:649)
    at Runner.showOption(Runner.java:69)
    at Runner.main(Runner.java:16)
```

8. Requirements to check 5 significantly different queries involving different relations and attributes

SQL Select Statements:

- SELECT * FROM Room
- SELECT * FROM Hotel

- `SELECT * FROM Guest`
- `SELECT * FROM City`
- `SELECT * FROM Country`

co-related query:

- `SELECT City.countryID, City.cityName, guestHotel.guestName
FROM
(SELECT guestName, cityName FROM Guest, Hotel WHERE Guest.hotelID =
Hotel.hotelID) guestHotel,
City
WHERE guestHotel.cityName = City.cityName;`

outer join:

- `SELECT Guest.HotelID
FROM Guest
LEFT OUTER JOIN Hotel
ON Guest.HotelID = Hotel.HotelID AND Guest.Is_Active = 0;`

group by:

- `SELECT HID, difference FROM
(SELECT hotelName AS HID, max(price) - min(price) AS difference
FROM Hotel, Room
WHERE Hotel.hotelID = Room.hotelID
GROUP BY Hotel.hotelID) DHP
WHERE difference >= ALL (SELECT max(price) - min(price) AS difference
FROM Hotel, Room
WHERE Hotel.hotelID = Room.hotelID
GROUP BY Hotel.hotelID);`

aggregation:

- `SELECT COUNT(*)
FROM Guest
WHERE Is_Active = 1;`

mathematical set operation:

- `SELECT DISTINCT hotelName, Hotel.cityName, countryName, price * 3
FROM Hotel INNER JOIN City ON Hotel.cityName = City.cityName
INNER JOIN Country ON City.countryID = Country.countryID
INNER JOIN Room ON Room.hotelID = hotelID;`

SQL Update Statements:

- `UPDATE Room SET roomStatus = 'Booked' WHERE roomID = 120`
- `UPDATE Hotel SET starsRating=3 WHERE hotelName='Four Seasons'`
- `UPDATE Guest SET is_Active=0 WHERE guestName = 'Inez'`
- `UPDATE City SET cityName='Seoul' WHERE countryID='HK'`

- UPDATE Country SET countryName='Australia' WHERE countryID='AT'

SQL Delete Statements:

- DELETE FROM Room WHERE
- DELETE FROM HOTEL WHERE cityName = 'Toronto'
- DELETE FROM Guest WHERE HotelID = 6
- DELETE FROM City WHERE cityName='Seoul'
- DELETE FROM Country WHERE countryName='Canada';

All SQL Insert Statements:

- INSERT INTO Room VALUES(1, 121, 'suite', 'empty', 55);
- INSERT INTO Hotel VALUES(5, 'Friendship Inn', 'Toronto', 4)
- INSERT INTO Guest VALUES (105, 'Jay', 0, 8);
- INSERT INTO City VALUES ('Victoria', 'HK');
- INSERT INTO Country VALUES ('KR', 'Korea');

All SQL Stored Triggers:

/*

Trigger 1:

Automatically deletes guests when their status becomes inactive / when they check out.

*/

CREATE TRIGGER deleteGuests

AFTER UPDATE ON Guest

FOR EACH ROW

 DELETE FROM Guest WHERE Is_Active = false;

/*

Trigger 2:

When a guest moves to another room, everyone in their room moves as well

*/

CREATE TRIGGER moveGuests

AFTER UPDATE ON Guest

FOR EACH ROW

 UPDATE Guest

 SET roomID = new.roomID

 WHERE guestID IN (SELECT guestID2 FROM Guest g2 WHERE g2.roomID = old.roomID);

/*

Trigger 3:

```
Automatically updates guest's room attribute to null when the guest's is_active status changes to
0
*/
CREATE TRIGGER updateGuestRoom
AFTER UPDATE ON Guest
FOR EACH ROW
    UPDATE Guest
    SET roomID = NULL WHERE is_Active = false;
```

All SQL Stored Procedures:

```
DROP PROCEDURE IF EXISTS getGuestList;
DELIMITER //
CREATE PROCEDURE getGuestList(cutoffDate timestamp)
BEGIN
    INSERT INTO Archive
        (SELECT *
         FROM Guest
         WHERE Guest.updatedAt < cutoffDate);
    DELETE FROM Guest WHERE Guest.updatedAt < cutoffDate;
END //
DELIMITER ;
```