**Database Systems**

**Coursework**

Contents

[1. Introduction 3](#_Toc57657356)

[1.1 Amendment made to Original Schema: 4](#_Toc57657366)

[1.2 Amendment made Coursework Stage1: 5](#_Toc57657382)

[2. Relational Schema and Justification of Normal Forms 6](#_Toc57657394)

[2.1 Relational Schema 6](#_Toc57657395)

[2.2 Justify Schema till 3rd Normal Form 6](#_Toc57657397)

[3. Schema Creation and Data Insertion 7](#_Toc57657421)

[3.1 DDL Statement/Table, Constrain, Index Creation 7](#_Toc57657422)

[3.2 DML Statement to insert test data in all the tables 8](#_Toc57657432)

[4. Report Query and Testing 8](#_Toc57657435)

[5. Appendix 10](#_Toc57657446)

1. Introduction

In coursework stage 2, I am going to present the final ER diagram. And I am deriving the relational schema as per my ER diagram, DDL statements to create database objects, inserting data into database. Finally present query to generate a report and test the report as per the coursework stage 1 test plan.

**Reference of the original model:**

Reference: <http://www.databaseanswers.org/data_models/hotels/index.htm>

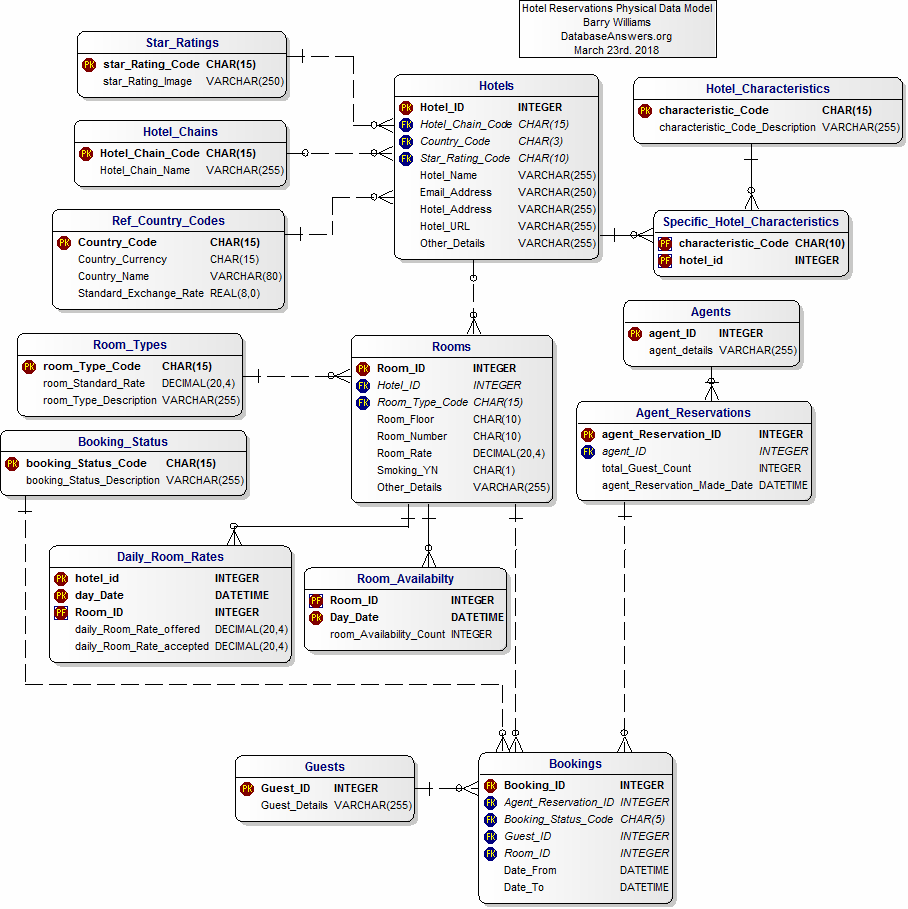
Barry Williams

Principal Consultant

Database Answers

London, England

**Original Schema Diagram:**



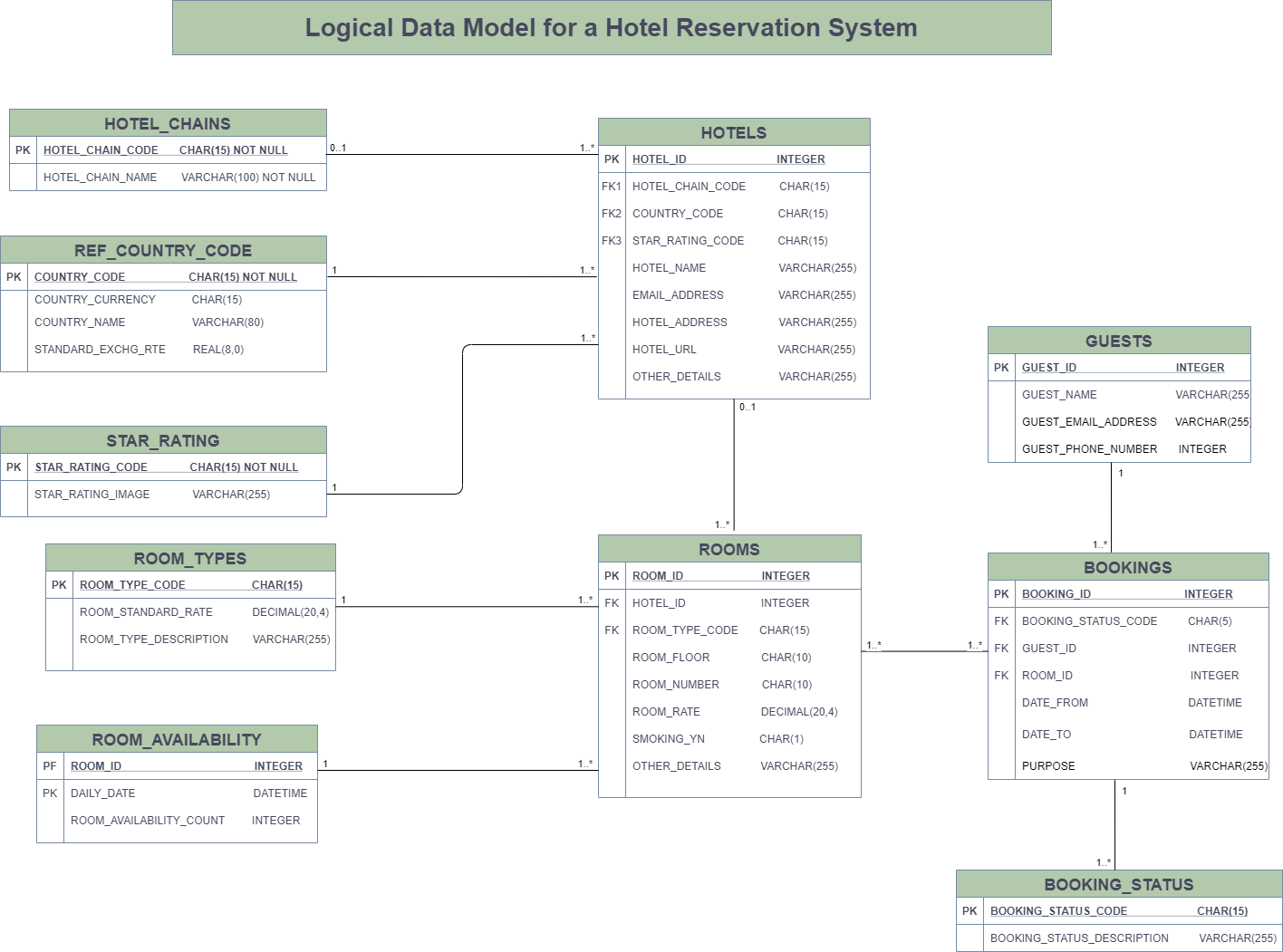
* 1. **Amendment made to Original Schema:**

Below amendments are made to the original schema:

|  |  |
| --- | --- |
| Amendment Made | Reason |
| HOTEL\_CHARACTERISTIC was removed | As I will not keep the provision of hotel characteristics. |
| SPECIFIC\_HOTEL\_CHARACTERISTIC was removed | As I will not keep the provision of hotel characteristics. |
| AGENT was removed | As I will not keep the provision of any agent or agent booking |
| AGENT\_RESERVATIONS | As I will not keep the provision of any agent or agent booking |
| DAILY\_ROOM\_RATES was removed | As I will not provision the estimated, actual room rate |
| GUESTS few columns were added | To provide detailed guest information |

* 1. **Amendment made Coursework Stage1:**

Below is the ER diagram of coursework stage1:

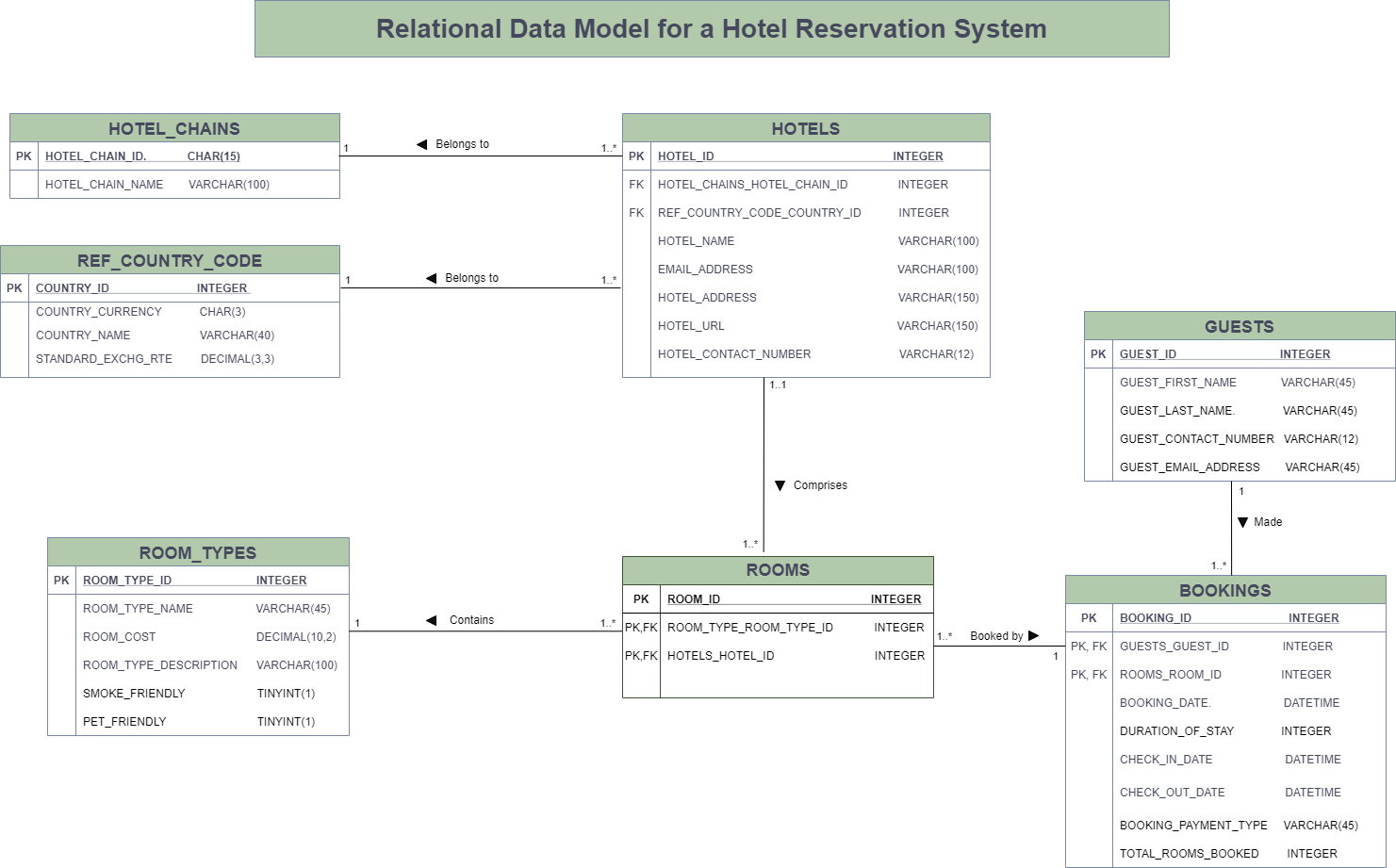


Below amendments are made to the coursework stage 1:

|  |  |
| --- | --- |
| Amendment Made | Reason |
| STAR\_RATING was removed | As this is not required for our database designing. |
| ROOM\_AVAILABILITY was removed | As this is not required for our database designing. |
| BOOKING\_STATUS was removed | As this is not required for our database designing. |

2. Relational Schema and Justification of Normal Forms

* 1. **Relational Schema**

****

* 1. **Justify Schema till 3rd Normal Form**

**First Normal Form:**

In first normal form the table should design in such a way so that the data can be easily retrieved and the table can be extendable.

I achieved the first normal form by:

* Keeping the data scalar in each field in the table. However in the ‘HOTEL’ table Hotel\_address column I could have separated the address by street name, city name and country name but I did not separate them because I don’t need them for any further reference.
* All the data items in a column has of same type. For an example: In the ‘HOTEL’ table Hotel\_id column has integer type data which stores the id of the hotel and Hotel\_name column has varchar type data which will only store the name of the hotel.
* Each column has unique name in all the tables. No column name has repeated.

**Second Normal Form:**

Coming to the second normal form, I have achieved this by:

* Keeping the tables in first normal form.
* I tried to make the relationships depend on the primary key.
* Each table has contain data about only one type of things.

**Third Normal form:**

For the tables to be in third normal form

* I kept the tables in second normal form.
* It doesn’t have transitive dependencies which means there is no other non-key attribute would need to change in a table if I have to change a non-key attribute.

HOTEL\_CHAINS (**HOTEL\_CHAIN\_ID**, HOTEL\_CHAIN\_NAME)

REF\_COUNTRY\_CODE (**COUNTRY\_ID**, COUNTRY\_CURRENCY, COUNTRY\_NAME, STANDARD\_EXCHG\_RATE)

HOTELS (**HOTEL\_ID**, *HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID, REF\_COUNTRY\_CODE\_COUNTRY\_ID*, HOTEL\_NAME, EMAIL\_ADDRESS, HOTEL\_ADDRESS, HOTEL\_URL, HOTEL\_CONTACT\_NUMBER)

ROOM\_TYPE (**ROOM\_TYPE\_ID**, ROOM\_TYPE\_NAME, ROOM\_COST, ROOM\_TYPE\_DESCRIPTION, SMOKE\_FRIENDLY, PET\_FRIENDLY)

ROOMS (**ROOM\_ID*,*** ***ROOM\_TYPE\_ROOM\_TYPE\_ID,*** ***HOTELS\_HOTEL\_ID***)

GUESTS (**GUEST\_ID**, GUEST\_FIRST\_NAME, GUEST\_LAST\_NAME, GUEST\_CONTACT\_NUMBER, GUEST\_EMAIL\_ADDRESS)

BOOKINGS (**BOOKING\_ID**, ***ROOMS\_ROOM\_ID***, ***GUESTS\_GUEST\_ID***, BOOKING\_DATE, DURATION\_OF\_STAY, CHECK\_IN\_DATE, CHECK\_OUT\_DATE, BOOKING\_PAYMENT\_TYPE, TOTAL\_ROOMS\_BOOKED)

Above represented schema is in third normal form. As per the third normal form rule we should also remove the derived data, however in the ‘BOOKINGS’ table I have not removed the column ‘DURATION\_OF\_STAY’ for the purpose of my manual testing of report query.

3. Schema Creation and Data Insertion

Below we have mentioned the details of schema creation.

**3.1 DDL Statement/Table, Constrain, Index Creation**

Please refer to Appendix-A for table structure for the 7 tables mentioned below:

* HOTEL\_CHAINS: Table consists of information related to hotel chain. It has One-to-Many relationship with HOTELS table.
* REF\_COUNTRY\_CODE: Table consists of information related to country of reference, currency and exchange rate for each individual hotel. It has One-to-Many relationship with HOTELS table.
* HOTELS: Table consists of information related to hotels. It has Many-to-One relationship with hotel chains, One-to-Many relationship with rooms, and Many-to-One relationship with ref country code tables.
* ROOM\_TYPE: Table consists of information related to room type in each hotels. It has One-to-Many relationship with rooms.
* ROOMS: Table is a relational entity between hotels and room type. It has Many-to-One relationship with hotels as well as room type.
* GUESTS: Table consists of information related to guests that made a reservation in the hotel. It has One-to-Many relationship with bookings.
* BOOKINGS: Table consists of information related to all the bookings made by each guests in each of the hotels. It has Many-to-One relationship with guests and Many-to-One relationship with rooms.

Attachment is given below for the DDL Statements:

(For text format of the same please refer Appendix B)



**3.2 DML Statement to insert test data in all the tables**

Attachment is given below for all the DML Statements:

(For text format of the same please refer Appendix B)



4. Report Query and Testing

**Test Plan Outline:**

As per our report from stage 1, we are going to generate revenue for each hotel chain by aggregating the booking amount for each of the booking per hotel chains. So in this case we will be showing the hotel chain id, hotel chain name, then sum of each room cost multiplied by duration of stay multiplied by number of rooms booked based on the group hotel chain id and hotel chain name.

So in our case an inner join between the table hotel chains, hotels, room types and rooms and bookings is required to get the relevant information.

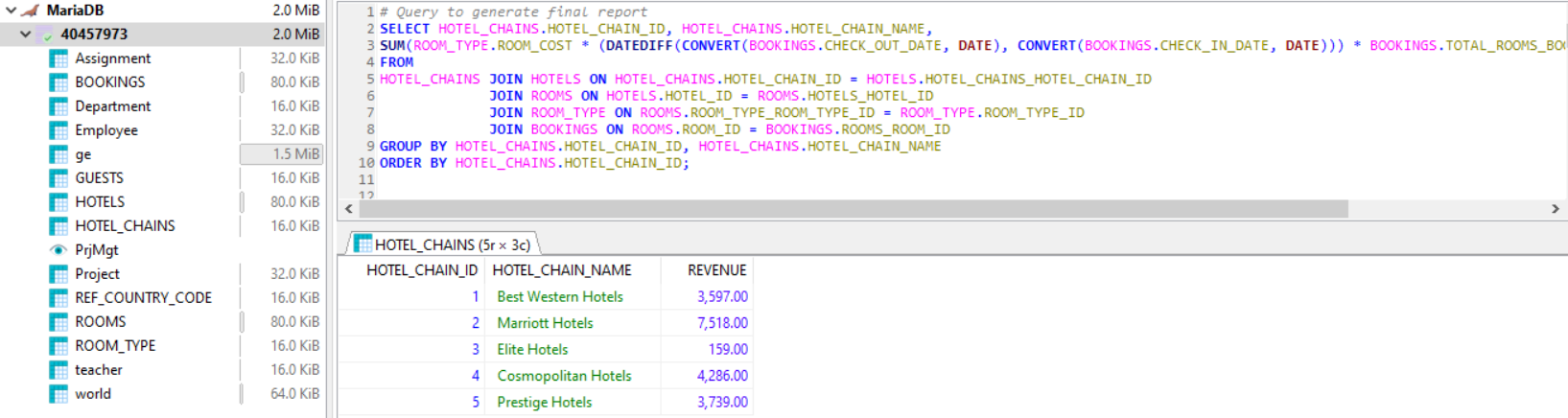
To test the above report we can take revenue for each individual hotel chain and then combine them using UNION operation to match with the above report. If both the results match, then we will ensure our report query is correct.

Below attached is our query to generate the report and UNION query to test our report:

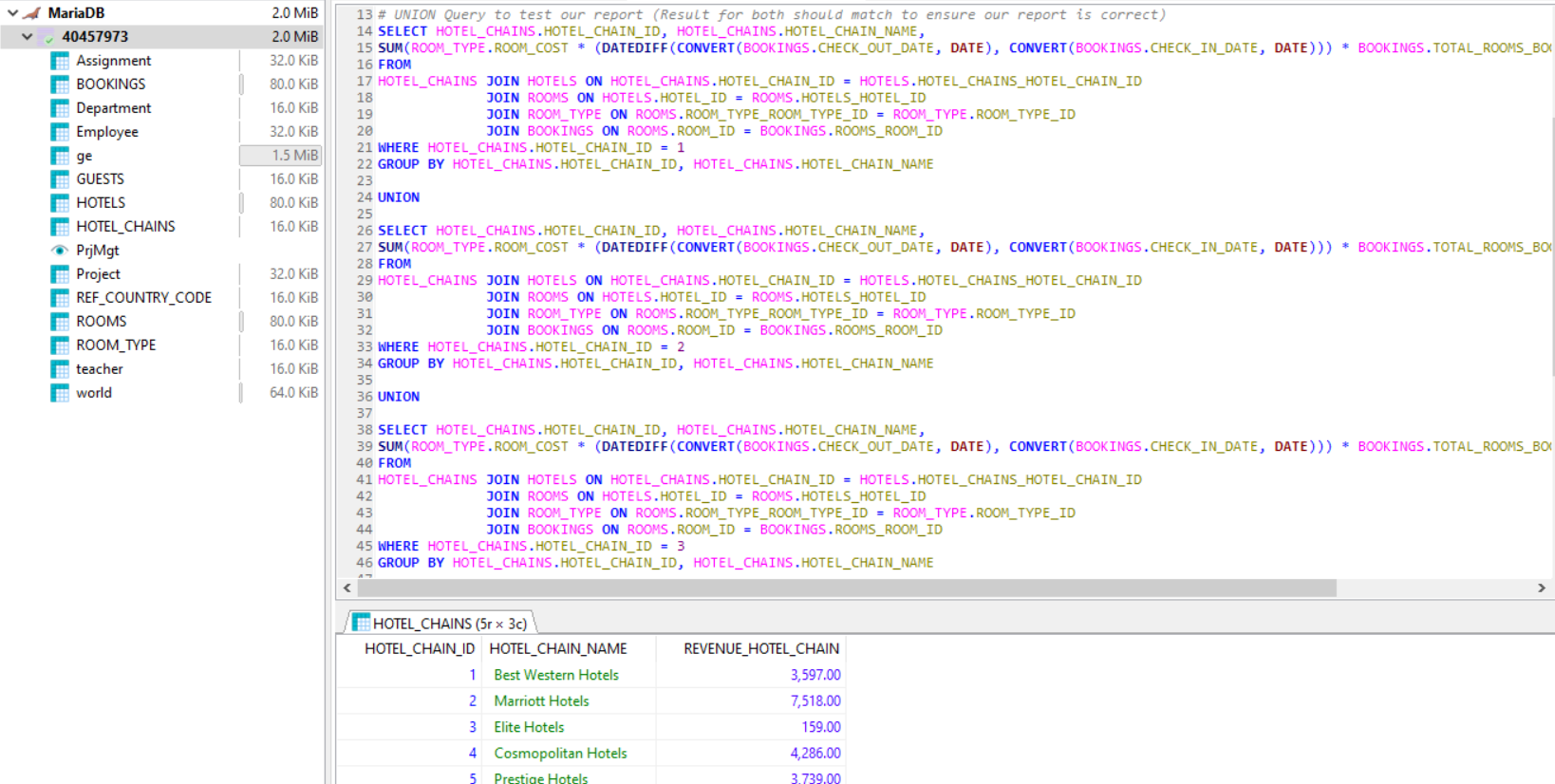
(For text format of the same please refer Appendix B)

****

**Screen Shot for Report:**

****

**Screen Shot for UNION Query to Test Report:**

****

5. Appendix

1. **Table Structure and Column Description:**

* **HOTEL\_CHAIN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **HOTEL\_CHAINS** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| HOTEL\_CHAIN\_ID | INT | PK | No | Uniquely Identify Hotel Chains |
| HOTEL\_CHAIN\_NAME | VARCHAR(100) |  | No | Name of Hotel Chain |

* **REF\_COUNTRY\_CODE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **REF\_COUNTRY\_CODE** | | |  |  |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| COUNTRY\_ID | INT | PK | No | Uniquely Identify Country in which hotel is present |
| COUNTRY\_CURRENCY | CHAR(3) |  | No | Currency of the Country |
| COUNTRY\_NAME | VARCHAR(40) |  | No | Country Name |
| STANDARD\_EXCHG\_RATE | DECIMAL(3,3) |  | No | Exchange rate wrt US Dollar |

* **HOTELS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **HOTELS** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| HOTEL\_ID | INT | PK | No | Uniquely Identify the hotel |
| HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID | INT | FK | Yes | Foreign Key from HOTEL\_CHAINS table |
| REF\_COUNTRY\_CODE\_COUNTRY\_ID | INT | FK | Yes | Foreign Key from REF\_COUNTRY\_CODE table |
| HOTEL\_NAME | VARCHAR(100) |  | No | Name of Hotel |
| EMAIL\_ADDRESS | VARCHAR(100) |  | No | Email Address Of Hotel |
| HOTEL\_ADDRESS | VARCHAR(150) |  | No | Address of Hotel |
| HOTEL\_URL | VARCHAR(150) |  | No | URL for the Hotel |
| HOTEL\_CONTACT\_NUMBER | VARCHAR(12) |  | No | Contact Number of the Hotel |

* **ROOM\_TYPE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROOM\_TYPE** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| ROOM\_TYPE\_ID | INT | PK | No | Uniquely Identifies each room type |
| ROOM\_TYPE\_NAME | VARCHAR(45) |  | No | Name For Room Type |
| ROOM\_COST | DECIMAL(10,2) |  | No | Cost Of Each Room Type |
| ROOM\_TYPE\_DESCRIPTION | VARCHAR(100) |  | No | Description For Room Type |
| SMOKE\_FRIENDLY | TINYINT(1) |  | No | 1 For Some Friendly 0 For Not Some Friendly |
| PET\_FRIENDLY | TINYINT(1) |  | No | 1 For Pet Friendly 0 For Not Pet Friendly |

* **ROOMS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ROOMS** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| ROOM\_ID | INT | PK | No | Uniquely Identifies each room |
| ROOM\_TYPE\_ROOM\_TYPE\_ID | INT | PK,FK | Yes | Foreign Key for ROOM\_TYPE |
| HOTELS\_HOTEL\_ID | INT | PK,FK | Yes | Foreign Key for HOTELS |

* **GUESTS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GUESTS** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| GUEST\_ID | INT | PK | No | Uniquely Identifies each Guest |
| GUEST\_FIRST\_NAME | VARCHAR(45) |  | No | First Name of Guest |
| GUEST\_LAST\_NAME | VARCHAR(45) |  | No | Last Name of Guest |
| GUEST\_CONTACT\_NUMBER | VARCHAR(12) |  | No | Phone Number of Guest |
| GUEST\_EMAIL\_ADDRESS | VARCHAR(45) |  | No | Email Address of Guest |

* **BOOKINGS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BOOKINGS** | | | | |
| ***Column Name*** | ***Data Type*** | ***Key*** | ***Index*** | ***Purpose*** |
| BOOKING\_ID | INT | PK | No | Uniquely Identifies Each Booking |
| BOOKING\_DATE | DATETIME |  | No | Date Of Booking |
| DURATION\_OF\_STAY | VARCHAR(10) |  | No | Duration for which booking is made |
| CHECK\_IN\_DATE | DATETIME |  | No | Check-In Date For The Booking |
| CHECK\_OUT\_DATE | DATETIME |  | No | Check-Out Date for The Booking |
| BOOKING\_PAYMENT\_TYPE | VARCHAR(45) |  | No | Payment Type For the Booking |
| TOTAL\_ROOMS\_BOOKED | INT |  | No | Number Of Rooms Booked per Booking |
| ROOMS\_ROOM\_ID | INT | PK, FK | Yes | Foreign Key From ROOMS Table |
| GUESTS\_GUEST\_ID | INT | PK,FK | Yes | Foreign Key From GUESTS Table |

1. **DDL, DML & Report Query:**

**DDL:**

*#------------------------------------------------------------------------------------------------*

*#-- Table HOTEL\_CHAINS*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `HOTEL\_CHAINS`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `HOTEL\_CHAINS`(

HOTEL\_CHAIN\_ID **INT** **NOT** **NULL**,

HOTEL\_CHAIN\_NAME **VARCHAR**(100) **NOT** **NULL**,

**PRIMARY** **KEY** (HOTEL\_CHAIN\_ID)

)**ENGINE** = **InnoDB**;

*#------------------------------------------------------------------------------------------------*

*#-- Table REF\_COUNTRY\_CODE*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `REF\_COUNTRY\_CODE`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `REF\_COUNTRY\_CODE`(

COUNTRY\_ID **INT** **NOT** **NULL**,

COUNTRY\_CURRENCY **CHAR**(3) **NOT** **NULL**,

COUNTRY\_NAME **VARCHAR**(40) **NOT** **NULL**,

STANDARD\_EXCHG\_RATE **DECIMAL**(6,3) **NOT** **NULL**,

**PRIMARY** **KEY** (COUNTRY\_ID)

)**ENGINE** = **InnoDB**;

*#------------------------------------------------------------------------------------------------*

*#-- Table HOTELS*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `HOTELS`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `HOTELS`(

HOTEL\_ID **INT** **NOT** **NULL**,

HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID **INT** **NOT** **NULL**,

REF\_COUNTRY\_CODE\_COUNTRY\_ID **INT** **NOT** **NULL**,

HOTEL\_NAME **VARCHAR**(100) **NULL**,

EMAIL\_ADDRESS **VARCHAR**(100) **NULL**,

HOTEL\_ADDRESS **VARCHAR**(150) **NULL**,

HOTEL\_URL **VARCHAR**(150) **NULL**,

HOTEL\_CONTACT\_NUMBER **VARCHAR**(12) **NULL**,

**PRIMARY** **KEY** (HOTEL\_ID),

**CONSTRAINT** HOTELS\_HOTEL\_CHAINS

**FOREIGN** **KEY** (HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID)

**REFERENCES** `HOTEL\_CHAINS`(HOTEL\_CHAIN\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**,

**CONSTRAINT** HOTELS\_REF\_COUNTRY\_CODE

**FOREIGN** **KEY** (REF\_COUNTRY\_CODE\_COUNTRY\_ID)

**REFERENCES** `REF\_COUNTRY\_CODE`(COUNTRY\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**

) **ENGINE** = **InnoDB**;

**CREATE** **INDEX** `FK\_HOTELS\_HOTEL\_CHAINS\_IDX` **ON** `HOTELS` (HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID **ASC**);

**CREATE** **INDEX** `FK\_HOTELS\_REF\_COUNTRY\_CODE\_IDX` **ON** `HOTELS` (REF\_COUNTRY\_CODE\_COUNTRY\_ID **ASC**);

*#------------------------------------------------------------------------------------------------*

*#-- Table ROOM\_TYPE*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `ROOM\_TYPE`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `ROOM\_TYPE`(

ROOM\_TYPE\_ID **INT** **NOT** **NULL**,

ROOM\_TYPE\_NAME **VARCHAR**(45) **NULL**,

ROOM\_COST **DECIMAL**(10,2) **NULL**,

ROOM\_TYPE\_DESCRIPTION **VARCHAR**(100),

SMOKE\_FRIENDLY **TINYINT**(1) **NULL**,

PET\_FRIENDLY **TINYINT**(1) **NULL**,

**PRIMARY** **KEY**(ROOM\_TYPE\_ID)

) **ENGINE** = **InnoDB**;

*#------------------------------------------------------------------------------------------------*

*#-- Table ROOMS*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `ROOMS`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `ROOMS`(

ROOM\_ID **INT** **NOT** **NULL**,

ROOM\_TYPE\_ROOM\_TYPE\_ID **INT** **NOT** **NULL**,

HOTELS\_HOTEL\_ID **INT** **NOT** **NULL**,

**PRIMARY** **KEY** (ROOM\_ID, ROOM\_TYPE\_ROOM\_TYPE\_ID, HOTELS\_HOTEL\_ID),

**CONSTRAINT** `FK\_ROOMS\_ROOM\_TYPE`

**FOREIGN** **KEY** (ROOM\_TYPE\_ROOM\_TYPE\_ID)

**REFERENCES** `ROOM\_TYPE`(ROOM\_TYPE\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**,

**CONSTRAINT** `FK\_ROOMS\_HOTELS`

**FOREIGN** **KEY** (HOTELS\_HOTEL\_ID)

**REFERENCES** `HOTELS`(HOTEL\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**

) **ENGINE** = **InnoDB**;

**CREATE** **INDEX** `FK\_ROOMS\_ROOM\_TYPE\_IDX` **ON** `ROOMS` (ROOM\_TYPE\_ROOM\_TYPE\_ID **ASC**);

**CREATE** **INDEX** `FK\_ROOMS\_HOTELS\_IDX` **ON** `ROOMS` (HOTELS\_HOTEL\_ID **ASC**);

*#------------------------------------------------------------------------------------------------*

*#-- Table GUESTS*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `GUESTS`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `GUESTS`(

GUEST\_ID **INT** **NOT** **NULL**,

GUEST\_FIRST\_NAME **VARCHAR**(45) **NULL**,

GUEST\_LAST\_NAME **VARCHAR**(45) **NULL**,

GUEST\_CONTACT\_NUMBER **VARCHAR**(12) **NULL**,

GUEST\_EMAIL\_ADDRESS **VARCHAR**(45) **NULL**,

**PRIMARY** **KEY**(GUEST\_ID)

) **ENGINE** = **InnoDB**;

*#------------------------------------------------------------------------------------------------*

*#-- Table BOOKINGS*

*#------------------------------------------------------------------------------------------------*

**DROP** **TABLE** **IF** **EXISTS** `BOOKINGS`;

**CREATE** **TABLE** **IF** **NOT** **EXISTS** `BOOKINGS`(

BOOKING\_ID **INT** **NOT** **NULL**,

ROOMS\_ROOM\_ID **INT** **NOT** **NULL**,

BOOKING\_DATE **DATETIME** **NULL**,

DURATION\_OF\_STAY **INT** **NULL**,

CHECK\_IN\_DATE **DATETIME** **NULL**,

CHECK\_OUT\_DATE **DATETIME** **NULL**,

BOOKING\_PAYMENT\_TYPE **VARCHAR**(45) **NULL**,

TOTAL\_ROOMS\_BOOKED **INT** **NULL**,

GUESTS\_GUEST\_ID **INT** **NOT** **NULL**,

**PRIMARY** **KEY** (BOOKING\_ID, GUESTS\_GUEST\_ID, ROOMS\_ROOM\_ID),

**CONSTRAINT** `FK\_BOOKINGS\_ROOMS`

**FOREIGN** **KEY**(ROOMS\_ROOM\_ID)

**REFERENCES** `ROOMS`(ROOM\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**,

**CONSTRAINT** `FK\_BOOKINGS\_GUESTS`

**FOREIGN** **KEY**(GUESTS\_GUEST\_ID)

**REFERENCES** `GUESTS`(GUEST\_ID)

**ON** **DELETE** **CASCADE**

**ON** **UPDATE** **RESTRICT**

) **ENGINE** = **InnoDB**;

**CREATE** **INDEX** `FK\_BOOKINGS\_ROOMS\_IDX` **ON** `BOOKINGS` (ROOMS\_ROOM\_ID **ASC**);

**CREATE** **INDEX** `FK\_BOOKINGS\_GUESTS\_IDX` **ON** `BOOKINGS` (GUESTS\_GUEST\_ID **ASC**);

**DML:**

**INSERT** **INTO** ROOM\_TYPE(`room\_type\_id`, `room\_type\_name`, `room\_cost`, `room\_type\_description`, `smoke\_friendly`, `pet\_friendly`)

**VALUES**

(1, 'Standard Room','103',"1 King Bed 323-sq-foot (30-sq-meter) room with city views",0,1),

(2, 'Standard Twin Room','123',"Two Twin Bed 323-sq-foot (30-sq-meter) room with city views",1,1),

(3, 'Executive Room','130',"1 King Bed 323-sq-foot (30-sq-meter) room with city views",0,0),

(4, 'Club Room','159',"2 King Bed 323-sq-foot (30-sq-meter) room with city views",1,1);

**INSERT** **INTO** GUESTS(`guest\_id`, `guest\_first\_name`, `guest\_last\_name`, `guest\_contact\_number`, `guest\_email\_address`)

**VALUES**

(1,'Jane','Doe','132-456-8564','jane.doe@gmail.com'),

(2,'Jerry','Thachter','564-896-4752','jerry.ytsvg@gmail.com'),

(3,'Rihanna','Perry','745-986-7451','rih.vfdj89@gmail.com'),

(4,'Mathew','Jose','489-624-8633','mathew.jose@gmail.com'),

(5,'Jessica','Smith','487-956-8963','jess.smith@gmail.com');

**INSERT** **INTO** HOTEL\_CHAINS(`hotel\_chain\_id`, `hotel\_chain\_name`)

**VALUES**

(1,'Best Western Hotels'),

(2,'Marriott Hotels'),

(3,'Elite Hotels'),

(4,'Cosmopolitan Hotels'),

(5,'Prestige Hotels');

**INSERT** **INTO** REF\_COUNTRY\_CODE (`country\_id`,`country\_currency`,`country\_name`,`standard\_exchg\_rate`)

**VALUES**

(1, 'GBP', 'United Kingdom',1.33),

(2, 'USD','United States Of America',1),

(3, 'INR','India',0.013),

(4, 'AUD','Australia',0.73),

(5,'CAD','Canada',0.76),

(6, 'EUR','Europe',1.18);

**INSERT** **INTO** HOTELS (`hotel\_id`, `hotel\_chains\_hotel\_chain\_id`,`ref\_country\_code\_country\_id`,`hotel\_name`, `email\_address`,`hotel\_address`,`hotel\_url`,`hotel\_contact\_number`)

**VALUES**

(1, 1, 2, 'Best Western Cabrillo Garden Inn', "groups@bestwestern.co.uk", "840 A Street San Diego, California 92101 United States", 'https://www.bestwesternCA.co.uk/', '619-234-8477'),

(2, 1, 6, "Yooma Urban Lodge, BW Premier Collection", "groupsbrussel@bestwestern.co.uk", "Square de L'Aviation 23-27 Brussels, 1070 Belgium", 'https://www.bestwesternbrussels.co.uk/', '32-2-5206565'),

(3, 2, 3, 'Bengaluru Marriott Hotel Whitefield', "mhrs.blrwf.reservations@marriott.com", "8th Road, Plot No 75, EPIP Area, Whitefield, Bengaluru 560066 India", 'https://www.marriott.co.uk/hotels/travel/blrwf-bengaluru-marriott-hotel-whitefield/', '80-4943-5000'),

(4, 2, 4, 'Melbourne Marriott Hotel', "mhrs.melmc.reservations@marriott.com", "Corner Exhibition & Lonsdale Streets, Melbourne 3000 Australia", 'https://www.marriott.com/hotels/travel/melmc-melbourne-marriott-hotel/', '+61396623900'),

(5, 2, 5, 'Toronto Marriott City Centre Hotel', "mhrs.yyzcc.reservations@marriott.com", "One Blue Jays Way, Toronto, Ontario M5V 1J4 Canada", 'https://www.marriott.com/hotels/travel/yyzcc-toronto-marriott-city-centre-hotel/', '416-341-7100'),

(6, 3, 1, "Ashdown Park Hotel & Country Club", "enquiries@ashdownpark.co.uk", "Wych Cross, Forest Row, East Grinstead RH18 5JR", 'https://www.ashdownpark.com/', '01342-824988'),

(7, 4, 6, 'Cosmopolitan Hotels', "info@chgroup.eu", "Largo Belvedere, 26 - 56128 - TIRRENIA - PISA - IT", 'https://www.chgroup.eu/', '39-050-37031'),

(8, 5, 5, 'Prestige Beach House', "Kelowna@PrestigeHotels.ca", "1675 Abbott Street Kelowna, BC V1Y 8S3", 'https://www.prestigehotelsandresorts.com/locations/kelowna/overview/', '250-860-7900'),

(9, 5, 5, 'Prestige Oceanfront Resort', "Sooke@PrestigeHotels.ca", "6929 West Coast Road Sooke, BC V9Z 0V1", 'https://www.prestigehotelsandresorts.com/locations/sooke/overview/', '250-642-0805'),

(10, 5, 5, 'Prestige Hotel Vernon', "Vernon@PrestigeHotels.ca", "Vernon, BC V1T 9G8", 'https://www.prestigehotelsandresorts.com/locations/vernon/overview/', '250-558-5991');

**INSERT** **INTO** ROOMS (`room\_id`, `room\_type\_room\_type\_id`, `hotels\_hotel\_id`)

**VALUES**

(1, 1, 1),

(2, 2, 1),

(3, 3, 1),

(4, 4, 1),

(5, 1, 2),

(6, 2, 2),

(7, 3, 2),

(8, 4, 2),

(9, 1, 3),

(10, 2, 3),

(11, 3, 3),

(12, 4, 3),

(13, 1, 4),

(14, 2, 4),

(15, 3, 4),

(16, 4, 4),

(17, 1, 5),

(18, 2, 5),

(19, 3, 5),

(20, 4, 5),

(21, 1, 6),

(22, 2, 6),

(23, 3, 6),

(24, 4, 6),

(25, 1, 7),

(26, 2, 7),

(27, 3, 7),

(28, 4, 7),

(29, 1, 8),

(30, 2, 8),

(31, 3, 8),

(32, 4, 8),

(33, 1, 9),

(34, 2, 9),

(35, 3, 9),

(36, 4, 9),

(37, 1, 10),

(38, 2, 10),

(39, 3, 10),

(40, 4, 10);

**INSERT** **INTO** BOOKINGS (`booking\_id`, `booking\_date`, `duration\_of\_stay`, `check\_in\_date`, `check\_out\_date`, `booking\_payment\_type`, `total\_rooms\_booked`, `rooms\_room\_id`, `guests\_guest\_id`)

**VALUES**

(101, '2020-01-08 00:00:00', 5, '2020-01-10 12:00:00', '2020-01-15 23:00:00', 'card', 1, 3, 4 ),

(102, '2020-01-01 00:00:00', 2, '2020-01-01 12:00:00', '2020-01-03 23:00:00', 'card', 1, 20, 1 ),

(103, '2020-01-15 00:00:00', 7, '2020-01-20 12:00:00', '2020-01-27 23:00:00', 'card', 1, 5, 2 ),

(104, '2020-02-02 00:00:00', 4, '2020-02-05 12:00:00', '2020-02-09 23:00:00', 'card', 2, 1, 3 ),

(105, '2020-02-05 00:00:00', 10, '2020-02-08 12:00:00', '2020-02-18 23:00:00', 'card', 1, 14, 5 ),

(106, '2020-02-10 00:00:00', 2, '2020-02-12 12:00:00', '2020-02-14 23:00:00', 'card', 3, 10, 4 ),

(107, '2020-02-18 00:00:00', 5, '2020-02-20 12:00:00', '2020-02-25 23:00:00', 'card', 1, 40, 2 ),

(108, '2020-02-25 00:00:00', 14, '2020-03-01 12:00:00', '2020-03-15 23:00:00', 'card', 1, 15, 1 ),

(109, '2020-03-02 00:00:00', 2, '2020-03-10 12:00:00', '2020-03-12 23:00:00', 'card', 1, 28, 3 ),

(110, '2020-03-15 00:00:00', 4, '2020-03-20 12:00:00', '2020-03-24 23:00:00', 'card', 1, 37, 2 ),

(111, '2020-03-18 00:00:00', 1, '2020-03-18 12:00:00', '2020-03-19 23:00:00', 'cash', 1, 24, 5 ),

(112, '2020-03-20 00:00:00', 12, '2020-03-25 12:00:00', '2020-04-06 23:00:00', 'card', 1, 28, 3 ),

(113, '2020-04-05 00:00:00', 6, '2020-04-07 12:00:00', '2020-04-13 23:00:00', 'cash', 1, 9, 4 ),

(114, '2020-04-15 00:00:00', 4, '2020-04-15 12:00:00', '2020-04-19 23:00:00', 'cash', 1, 10, 2 ),

(115, '2020-04-15 00:00:00', 5, '2020-04-20 12:00:00', '2020-04-25 23:00:00', 'card', 3, 30, 1 ),

(116, '2020-04-20 00:00:00', 2, '2020-04-25 12:00:00', '2020-04-27 23:00:00', 'card', 1, 36, 5 ),

(117, '2020-05-01 00:00:00', 7, '2020-05-07 12:00:00', '2020-05-14 23:00:00', 'card', 1, 7, 3 ),

(118, '2020-05-08 00:00:00', 3, '2020-05-15 12:00:00', '2020-05-18 23:00:00', 'card', 1, 38, 2 ),

(119, '2020-06-01 00:00:00', 5, '2020-06-05 12:00:00', '2020-06-10 23:00:00', 'card', 4, 25, 5 ),

(120, '2020-06-15 00:00:00', 2, '2020-06-17 12:00:00', '2020-06-19 23:00:00', 'card', 2, 6, 1 ),

(121, '2020-07-02 00:00:00', 8, '2020-07-04 12:00:00', '2020-07-12 23:00:00', 'card', 1, 16, 3 ),

(122, '2020-08-01 00:00:00', 10, '2020-08-08 12:00:00', '2020-08-18 23:00:00', 'card', 1, 9, 4 );

**Report Query:**

*# Query to generate final report*

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME

**ORDER** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID;

**Test Plan Validation Query:**

*# UNION Query to test our report (Result for both should match to ensure our report is correct)*

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE\_HOTEL\_CHAIN

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**WHERE** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = 1

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME

**UNION**

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE\_HOTEL\_CHAIN

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**WHERE** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = 2

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME

**UNION**

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE\_HOTEL\_CHAIN

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**WHERE** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = 3

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME

**UNION**

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE\_HOTEL\_CHAIN

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**WHERE** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = 4

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME

**UNION**

**SELECT** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME,

**SUM**(ROOM\_TYPE.ROOM\_COST \* (**DATEDIFF**(**CONVERT**(BOOKINGS.CHECK\_OUT\_DATE, **DATE**), **CONVERT**(BOOKINGS.CHECK\_IN\_DATE, **DATE**))) \* BOOKINGS.TOTAL\_ROOMS\_BOOKED) **AS** REVENUE\_HOTEL\_CHAIN

**FROM**

HOTEL\_CHAINS **JOIN** HOTELS **ON** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = HOTELS.HOTEL\_CHAINS\_HOTEL\_CHAIN\_ID

**JOIN** ROOMS **ON** HOTELS.HOTEL\_ID = ROOMS.HOTELS\_HOTEL\_ID

**JOIN** ROOM\_TYPE **ON** ROOMS.ROOM\_TYPE\_ROOM\_TYPE\_ID = ROOM\_TYPE.ROOM\_TYPE\_ID

**JOIN** BOOKINGS **ON** ROOMS.ROOM\_ID = BOOKINGS.ROOMS\_ROOM\_ID

**WHERE** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID = 5

**GROUP** **BY** HOTEL\_CHAINS.HOTEL\_CHAIN\_ID, HOTEL\_CHAINS.HOTEL\_CHAIN\_NAME;