

Scrollatravel

1. Member list

Student ID	Student Name	Responsibility(%)
63130500109	Siripong phiwkhaw	25
63130500129	Anuwat Pattanajan	25
63130500133	Aritach krudminburee	25
63130500148	Kittayod wongwuttinun	25

Table of Contents	Page
Business Requirements.....	3
Logical DB Design (Table Design).....	4
SQL statements (at least 5 statements).....	5
Data Dictionary.....	8
DDL script.....	14
Export Data.....	19

Business Requirements

User :

ผู้ใช้งานแบ่งออกเป็น 2 กลุ่ม ได้แก่

- ผู้เช่าที่พักอาศัย
- เจ้าของโรงแรม และ ผู้ปล่อยที่พักอาศัยให้เช่า

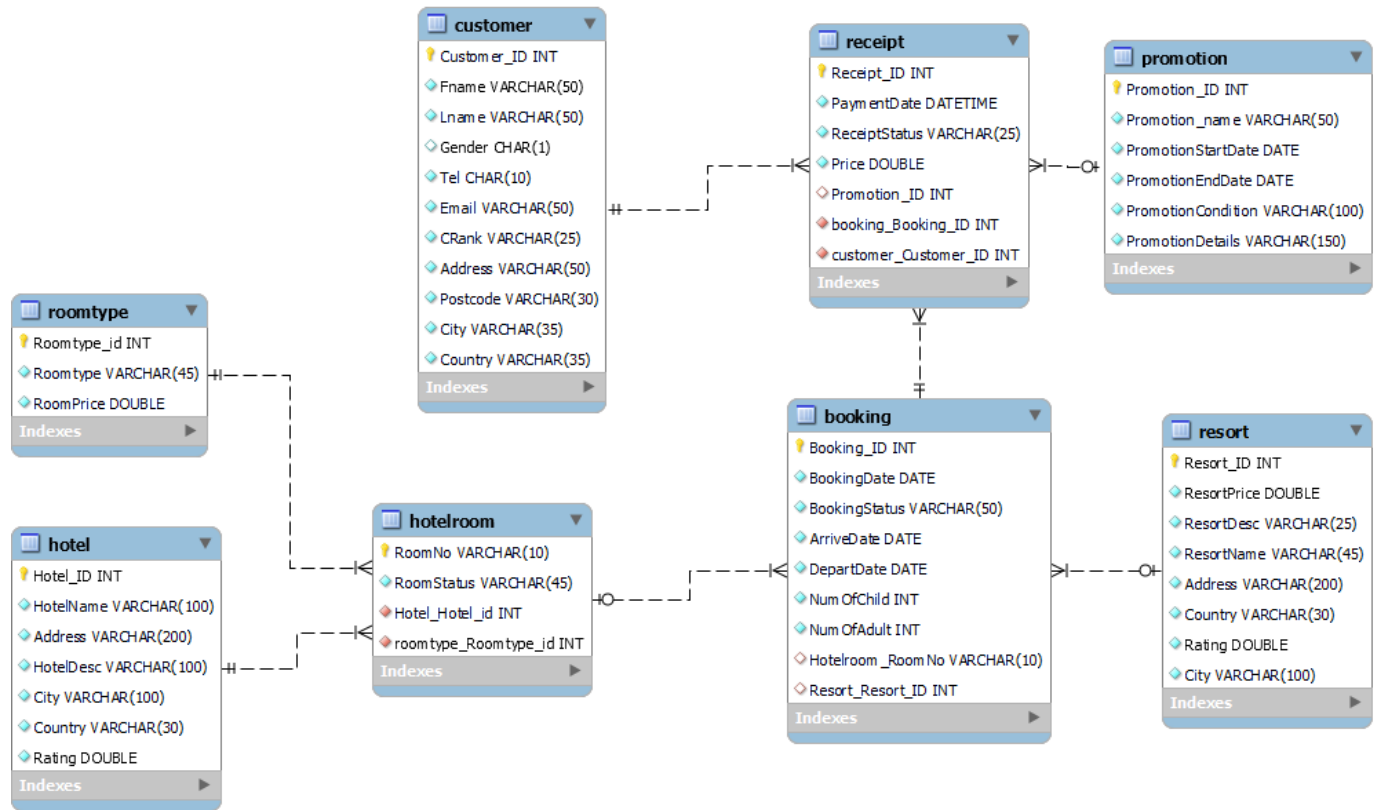
Features :

1. ผู้เช่าที่พักอาศัยสามารถเลือกได้ว่า จะเลือกพักโรงแรม หรือบ้านที่มีผู้ปล่อยที่พักอาศัยให้เช่า
2. ผู้ปล่อยที่พักอาศัยให้เช่าและผู้มาพักอาศัยสามารถระบุข้อมูลส่วนตัวของตัวเองได้
3. ผู้เช่าที่พักอาศัยสามารถจองที่พักอาศัยได้ล่วงหน้า และสามารถยกเลิกการจองได้ โดยมีกฎว่าหลังจากชำระเงินแล้วต้องการที่จะยกเลิกการจอง จะได้รับเงินคืนเพียง 80%
4. มีการให้คะแนนแก่ผู้ปล่อยที่พักอาศัยและผู้เช่าที่พักอาศัยเพื่อสร้างความน่าเชื่อถือและยังมีคะแนนมากจะมี Rating สูง
5. ผู้ที่ปล่อยให้เช่าหรือโรงแรมสามารถซื้อโฆษณาในแอปให้ขึ้นหน้าแรกๆได้แต่ต้องผ่านการตรวจสอบของแอปพลิเคชันของเราก่อนโดยการที่ต้องไปลงพื้นที่จริง
6. ฝ่ายผู้เช่าที่พักอาศัยยังมี rank สูงจะมีสิทธิพิเศษในการลดค่าใช้จ่ายของที่พักราคา

Benefits :

1. ความปลอดภัยในการชำระเงินและความสบายใจของทั้งผู้เช่ากับผู้อาศัย
2. ผู้เช่าที่พักอาศัยนั้นสามารถรอกข้อมูลแค่ครั้งเดียวพอ ทำให้ประหยัดเวลา
3. ความสะดวกสบายในการจองไม่มีความยุ่งยาก
4. ทางตัวของ ผู้ใช้งาน ทั้งฝั่งผู้มาพักอาศัยและผู้ปล่อยที่พักอาศัยให้เช่า มีโปรโมชั่นต่างๆ ตามข้อตกลงในแอปเรา
5. มีการให้บริการด้านความสะดวกสบายภายในแอป ทำให้ ผู้ใช้งาน ไม่ต้องใช้แอปอื่น
6. ผู้ปล่อยที่พักไม่ต้องเสียเวลาในการดำเนินเอกสารต่างๆ หรือหาข้อมูลเกี่ยวกับความน่าเชื่อถือของตัวผู้มาพักอาศัย
7. สามารถค้นหาโรงแรมและตรวจสอบห้องว่างได้แบบทันที

logical DB Design (Table Design)



SQL statements (at least 5 statements)

- Description
- SQL statement
- Query Result

1.

Description : แสดงชื่อผู้ใช้รวม ID และ ประวัติในการจอง ของ Booking ที่ ทำการจองที่ Hotelและบอกว่าห้องที่จองคือห้องอะไร

SQL statement : select c.Customer_ID,c.Fname,c.Lname,c.CRank,b.*, r.Price
from booking b join receipt r on b.Booking_ID = r.booking_Booking_ID join customer c on
c.Customer_ID = r.customer_Customer_ID
where b.Resort_Resort_ID is null;

Query Result :

```

214 • select c.Customer_ID,c.Fname,c.Lname,c.CRank,b.*, r.Price
215 from booking b join receipt r on b.Booking_ID = r.booking_Booking_ID join customer c on c.Customer_ID = r.customer_Customer_ID
216 where b.Resort_Resort_ID is null;
217
218
219

```

Customer_ID	Fname	Lname	CRank	Booking_ID	BookingDate	BookingStatus	ArriveDate	DepartDate	NumOfChild	NumOfAdult	Hotelroom_RoomNo	Resort_Resort_ID	Price
1111	Inwza	Aekwa	Gold	4001	2021-09-30	Book succeeded	2021-10-01	2021-10-03	1	2	2201	NULL	2985
1112	Siripong	phiwkhaw	silver	4002	2021-09-28	Book succeeded	2021-10-02	2021-10-06	1	0	2202	NULL	2700

2.

Description : ลูกค้าที่ลงไว้ว่ามีผู้ใหญ่มากกว่าเท่ากับ 2 คน ในการจอง

SQL statement : select c.Customer_ID, c.Fname, c.Lname, b.NumOfAdult
from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID
join booking b on r.booking_Booking_ID = b.Booking_ID
where b.NumOfAdult >= 2;

Query Result :

```

4 • select c.Customer_ID, c.Fname, c.Lname, b.NumOfAdult
5   from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID
6   join booking b on r.booking_Booking_ID = b.Booking_ID
7   where b.NumOfAdult >= 2;

```

Customer_ID	Fname	Lname	NumOfAdult
1111	Inwza	Aekwa	2
1113	Anuwat	Pattanjana	3

3.

Description : แสดง ID ชื่อต้นและชื่อท้ายของลูกค้าในรูปแบบ "ชื่อต้น ชื่อท้าย" เพศ เบอร์ติดต่อ ของลูกค้าที่เป็นผู้ชายและมีระดับ Rank เป็น Gold

SQL statement : SELECT Customer_ID , concat(Fname , ' ' , Lname) as 'Full Name',
Gender , Tel , CRank FROM customer where Gender = 'M' and CRank = 'Gold';

Query Result :

```

4 • SELECT Customer_ID , concat(Fname , ' ' , Lname) as 'Full Name', Gender , Tel , CRank
5   FROM customer where Gender = 'M' and CRank = 'Gold';

```

Customer_ID	Full Name	Gender	Tel	CRank
1111	Inwza Aekwa	M	0123456789	Gold
1113	Anuwat Pattanjana	M	54645464	Gold

4.

Description : แสดงชื่อ นามสกุล ของลูกค้าที่ทำการจองสำเร็จ

SQL statement : select c.Fname, c.Lname, b.BookingStatus

from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID

join booking b on r.booking_Booking_ID = b.Booking_ID

where b.BookingStatus = 'Book succeeded';

Query Result :

```

15 • select c.Fname, c.Lname, b.BookingStatus
16 from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID
17 join booking b on r.booking_Booking_ID = b.Booking_ID
18 where b.BookingStatus = 'Book succeeded';
19

```

Fname	Lname	BookingStatus
Inwza	Aekwa	Book succeeded
Siripong	phiwkhaw	Book succeeded
Aritach	krudminburee	Book succeeded
Kittayod	wongwuttinun	Book succeeded

5.

Description : แสดงเลข ID ที่ทำการจอง และ ชื่อและนามสกุลของลูกค้าที่ไม่ได้ใช้ promotion ใดๆเลยในการจอง

SQL statement : select b.Booking_ID ,c.Fname, c.Lname, r.Promotion_ID

from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID

join booking b on r.booking_Booking_ID = b.Booking_ID

where r.Promotion_ID is NULL;

Query Result :

```

22 • select b.Booking_ID ,c.Fname, c.Lname, r.Promotion_ID
23 from customer c join receipt r on c.Customer_ID = r.customer_Customer_ID
24 join booking b on r.booking_Booking_ID = b.Booking_ID
25 where r.Promotion_ID is NULL;
26

```

Booking_ID	Fname	Lname	Promotion_ID
4002	Siripong	phiwkhaw	NULL
4003	Anuwat	Pattanaajan	NULL
4004	Aritach	krudminburee	NULL
4005	Kittayod	wongwuttinun	NULL

Data Dictionary

Table : Customer

Description : เป็น table ที่จะเก็บข้อมูลต่างๆของผู้ใช้งาน เช่น ชื่อ ที่อยู่และการติดต่อของผู้ใช้งาน

Relation: One-to-Many with Receipt

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Customer_ID	INT	เลข ID ของ ผู้ใช้งาน	
NN	Fname	Varchar(50)	ชื่อต้นของผู้ใช้งาน	
NN	Lname	Varchar(50)	ชื่อท้ายของผู้ใช้งาน	
	Gender	Char(1)	เพศของผู้ใช้ โดยจะใส่หรือไม่ใส่ก็ได้ จะมีข้อมูลแค่ M(male) และ F(female)	
NN	Tel	Char(10)	เบอร์โทรศัพท์มือถือ 10 หลัก	
NN	Email	Varchar(50)	ที่อยู่ Email ไว้สำหรับแจ้งเตือนและติดต่อข่าวสาร	
NN	CRank	Varchar(25)	ระดับแรงค์ของผู้ใช้งาน เช่น Bronze, Silver , Gold , Platinum	
NN	Address	Varchar(50)	ที่อยู่ของผู้ใช้งาน บ้านเลขที่ เขต อื่นๆ	
NN	Postcode	Varchar(30)	เลขไปรษณีย์	
NN	City	Varchar(35)	ชื่อของเมืองที่ผู้ใช้งานอยู่อาศัย	
NN	Country	Varchar(35)	ชื่อของประเทศที่ผู้ใช้งานอยู่อาศัย	

Table : Booking

Description : เป็น table ที่จะเก็บข้อมูลประวัติในการจองของลูกค้าและเป็นตารางหลักในการใช้งาน จะเก็บข้อมูล เช่น วันที่ทำการจอง สถานะในการจอง เวลาที่ทำการจอง วันที่เริ่มพักจนถึงวันสุดท้ายที่พัก และยังมี FK จาก table ต่างๆเช่น เลขid ที่อยู่อาศัยหรือเลขid ของ ผู้ใช้งาน

Relation: Many-to-Zero or One with Hotelroom

Many-to-Zero or One with Resort

One-to-Many with Receipt

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Booking_ID	INT	เลข ID ในการจอง	
NN	BookingDate	DATE	วันและเวลาที่ทำการจอง	
NN	BookingStatus	Varchar(50)	สถานะของการจอง เช่น รอยืนยันตัวตน , ถูกยกเลิก , รอการชำระเงิน , ทริบถูกปฏิเสธ , จองแล้ว	
NN	ArriveDate	DATE	วันที่เริ่มพักอาศัย	
NN	DepartDate	DATE	วันสิ้นสุดการพักอาศัย	
NN	NumOfChild	INT	จำนวนเด็กที่พักอาศัย	
NN	NumOfAdult	INT	จำนวนผู้ใหญ่ที่พักอาศัย	
FK	Hotelroom_RoomNo	VARCHAR(10)	แสดงถึงประเภทห้องและจำนวนห้องที่จอง	Hotelroom
FK	Resort_ID	INT	เลข ID ของที่พัก	Resort

Table : Resort

Description : เป็น Table สำหรับที่พักถ้าหากเป็นประเภทรีสอร์ทหรือบ้านเช่า

Relation: Zero or One-to-Many with Booking

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
------------	----------------	------	-------------	------------------

PK	Resort_ID	INT	เลข ID ของที่พัก	
NN	ResortPrice	DOUBLE	ราคาที่พัก	
NN	ResortDesc	VARCHAR(25)	รายละเอียดที่พัก	
NN	ResortName	VARCHAR(45)	ชื่อที่พัก	
NN	Address	VARCHAR(200)	สถานที่ของที่พัก	
NN	Country	VARCHAR(30)	ประเทศที่ตั้งของที่พัก	
NN	City	VARCHAR(100)	เมืองที่ตั้งของที่พัก	
NN	Rating	DOUBLE	ระดับคะแนนการโหวต	

Table : Hotelroom

Description : เป็น Table สำหรับเก็บเลขห้องกับสถานะของห้อง

Relation: Many-to-One with RoomType

Many-to-One with Hotel

Zero or One-to-Many with Booking

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	RoomNo	VARCHAR(10)	เลข ID ห้อง	
NN	RoomStatus	VARCHAR(45)	สถานะของห้อง เช่น "ห้องว่าง,มีคนพักอยู่"	
FK	Hotel_Hotel_id	INT	Id ของโรงแรม	Hotel
FK	roomtype_ Roomtype_id	INT	Id ของชนิดของห้องพัก	Roomtype

Table : RoomType

Description : เป็น table สำหรับ เก็บประเภทของห้องพักที่โรงแรมจะใช้เก็บแครูปแบบห้อง ราคา รายละเอียดของห้อง

Relation : One-to-Many with Hotelroom

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Roomtype_id	INT	Id ของชนิดห้อง	
NN	RoomPrice	DOUBLE	ราคาของห้อง	
NN	RoomType	VARCHAR(45)	ประเภทของห้อง	

Table: Hotel

Description: เป็น table สำหรับเก็บชื่อที่อยู่และรายละเอียดของโรงแรม

Relation: One-to-Many with Hotelroom

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Hotel_ID	INT	เลข ID ของที่พัก	
NN	HotelDesc	VARCHAR(100)	รายละเอียดที่พัก	
NN	HotelName	VARCHAR(25)	ชื่อที่พัก	
NN	Address	VARCHAR(200)	สถานที่ของที่พัก	
NN	Country	VARCHAR(30)	ประเทศที่ตั้งของที่พัก	
NN	City	VARCHAR(100)	เมืองที่ตั้งของที่พัก	
NN	Rating	DOUBLE	ระดับคะแนนการโหวต	

Table : Receipt

Description : เป็น table ที่จะเก็บข้อมูลรายละเอียดของใบเสร็จ ทั้ง id ของใบเสร็จ,การจอง,โปรโมชั่น, วันที่จ่ายเงิน, สถานะ, ระยะเวลาที่พัก, จำนวนห้อง และ ราคา

Relation: Many-to-One with Customer

Many-to-Zero or One with Promotion

Many-to-One with Booking

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Receipt_ID	INT	เลข ID ของใบเสร็จ	
NN	PaymentDate	DATETIME	วันที่และเวลาจ่ายเงิน	
NN	ReceiptStatus	VARCHAR(25)	สถานะของใบเสร็จ "ชำระเงินแล้ว", "ยังไม่ได้ชำระเงิน"	
NN	Price	DOUBLE	ราคาที่ต้องจ่าย	
FK	Promotion_ID	INT	เลข ID โปรโมชั่น	Promotion
FK,NN	booking_Booking_ID	INT	เลข ID ในการจอง	Booking
FK,NN	customer_Customer_ID	INT	เลข ID ของ ผู้ใช้งาน	Customer

Table : Promotion

Description : เป็น table ที่จะเก็บข้อมูลรายละเอียดของโปรโมชั่น ทั้งชื่อ วัน และ เงื่อนไขของโปรโมชั่นต่างๆ

Relation: Zero or One-to-Many with Receipt

Constraint	ชื่อ Attribute	Type	Description	Referenced Table
PK	Promotion_ID	INT	เลขรหัสของโปรโมชั่น	
NN	Promotion_name	VARCHAR(50)	ชื่อของโปรโมชั่น	
NN	PromotionStartDate	DATE	วันที่โปรโมชั่นเริ่มใช้งาน	
NN	PromotionEndDate	DATE	วันที่โปรโมชั่นยุติการใช้งาน	
NN	PromotionCondition	VARCHAR(100)	เงื่อนไขของการใช้งานโปรโมชั่น	
NN	PromotionDetails	VARCHAR(150)	รายละเอียดของโปรโมชั่น โปรโมชั่นลดเป็นเปอร์เซ็นต์เท่านั้น เช่น 5%,10%,15% และบอกว่าเป็นโปร โมชั่นจาก rankอะไร	

DDL script for creating the database

```
CREATE DATABASE Scrollatravelnew;
```

```
USE Scrollatravelnew;
```

```
CREATE TABLE IF NOT EXISTS Hotel (  
    Hotel_ID INT NOT NULL,  
    HotelName VARCHAR(100) NOT NULL,  
    Address VARCHAR(200) NOT NULL,  
    HotelDesc VARCHAR(100) NOT NULL,  
    City VARCHAR(100) NOT NULL,  
    Country VARCHAR(30) NOT NULL,  
    Rating DOUBLE NOT NULL,  
    PRIMARY KEY (Hotel_ID),  
    CONSTRAINT Rating_ck check (Rating >= 0 or Rating <=5 )  
);
```

```
CREATE TABLE IF NOT EXISTS roomtype (  
    Roomtype_id INT NOT NULL,  
    Roomtype VARCHAR(45) NOT NULL,  
    RoomPrice DOUBLE NOT NULL,  
    PRIMARY KEY (Roomtype_id));
```

```
CREATE TABLE IF NOT EXISTS hotelroom (  
    RoomNo VARCHAR(10) NOT NULL,  
    RoomStatus VARCHAR(45) NOT NULL,
```

```

Hotel_Hotel_id INT NOT NULL,
roomtype_Roomtype_id INT NOT NULL,
PRIMARY KEY (RoomNo),
INDEX fk_Hotelroom_Hotel1_idx (Hotel_Hotel_id ASC) VISIBLE,
INDEX fk_hotelroom_roomtype1_idx (roomtype_Roomtype_id ASC) VISIBLE,
CONSTRAINT fk_Hotelroom_Hotel1
    FOREIGN KEY (Hotel_Hotel_id)
    REFERENCES hotel (Hotel_ID),
CONSTRAINT fk_hotelroom_roomtype1
    FOREIGN KEY (roomtype_Roomtype_id)
    REFERENCES roomtype (Roomtype_id),
CONSTRAINT RoomStatus_ck check (RoomStatus = 'Available' or RoomStatus =
'Unavailable'));

```

```

CREATE TABLE IF NOT EXISTS Resort (
    Resort_ID INT NOT NULL,
    ResortPrice DOUBLE NOT NULL,
    ResortDesc VARCHAR(25) NOT NULL,
    ResortName VARCHAR(45) NOT NULL,
    Address VARCHAR(200) NOT NULL,
    Country VARCHAR(30) NOT NULL,
    Rating DOUBLE NOT NULL,
    City VARCHAR(100) NOT NULL,
    PRIMARY KEY (Resort_ID),
    CONSTRAINT RatingResort_ck check (Rating >= 0 or Rating <=5 )
);

```

```
CREATE TABLE IF NOT EXISTS customer (  
    Customer_ID INT NOT NULL,  
    Fname VARCHAR(50) NOT NULL,  
    Lname VARCHAR(50) NOT NULL,  
    Gender CHAR(1) NULL DEFAULT NULL,  
    Tel CHAR(10) NOT NULL,  
    Email VARCHAR(50) NOT NULL,  
    CRank VARCHAR(25) NOT NULL,  
    Address VARCHAR(50) NOT NULL,  
    Postcode VARCHAR(30) NOT NULL,  
    City VARCHAR(35) NOT NULL,  
    Country VARCHAR(35) NOT NULL,  
    PRIMARY KEY (Customer_ID),  
    constraint CRank_ck check (CRank = 'Gold' or CRank = 'silver' or CRank = 'bronze' or  
CRank = 'platinum'),  
    constraint Gender_ck check (Gender = 'M' or Gender = 'F')  
);
```

```
CREATE TABLE IF NOT EXISTS promotion (  
    Promotion_ID INT NOT NULL,  
    Promotion_name VARCHAR(50) NOT NULL,  
    PromotionStartDate DATE NOT NULL,  
    PromotionEndDate DATE NOT NULL,  
    PromotionCondition VARCHAR(100) NOT NULL,  
    PromotionDetails VARCHAR(150) NOT NULL,
```


PRIMARY KEY (Promotion_ID)

);

CREATE TABLE IF NOT EXISTS booking (

Booking_ID INT NOT NULL,

BookingDate DATE NOT NULL,

BookingStatus VARCHAR(50) NOT NULL,

ArriveDate DATE NOT NULL,

DepartDate DATE NOT NULL,

NumOfChild INT NOT NULL,

NumOfAdult INT NOT NULL,

Hotelroom_RoomNo VARCHAR(10) NULL,

Resort_Resort_ID INT NULL,

PRIMARY KEY (Booking_ID),

INDEX fk_booking_Hotelroom1_idx (Hotelroom_RoomNo ASC) VISIBLE,

INDEX fk_booking_Resort1_idx (Resort_Resort_ID ASC) VISIBLE,

CONSTRAINT fk_booking_Hotelroom1

FOREIGN KEY (Hotelroom_RoomNo)

REFERENCES Hotelroom (RoomNo),

CONSTRAINT fk_booking_Resort1

FOREIGN KEY (Resort_Resort_ID)

REFERENCES Resort (Resort_ID),

constraint BookingStatus_ck check (BookingStatus = 'Book unsucceeded' or
BookingStatus = 'Book succeeded'));

CREATE TABLE IF NOT EXISTS receipt (

Receipt_ID INT NOT NULL,

PaymentDate DATETIME NOT NULL,

```

ReceiptStatus VARCHAR(25) NOT NULL,
Price DOUBLE NOT NULL,
Promotion_ID INT NULL DEFAULT NULL,
booking_Booking_ID INT NOT NULL,
customer_Customer_ID INT NOT NULL,
PRIMARY KEY (Receipt_ID),
INDEX FKPromotion_ID (Promotion_ID ASC) VISIBLE,
INDEX fk_receipt_booking1_idx (booking_Booking_ID ASC) VISIBLE,
INDEX fk_receipt_customer1_idx (customer_Customer_ID ASC) VISIBLE,
CONSTRAINT FKPromotion_ID
    FOREIGN KEY (Promotion_ID)
    REFERENCES promotion (Promotion_ID),
CONSTRAINT fk_receipt_booking1
    FOREIGN KEY (booking_Booking_ID)
    REFERENCES booking (Booking_ID),
CONSTRAINT fk_receipt_customer1
    FOREIGN KEY (customer_Customer_ID)
    REFERENCES customer (Customer_ID),
    constraint ReceiptStatus_ck check (ReceiptStatus = 'not paided' or
ReceiptStatus = 'paided' or ReceiptStatus = 'cancel'));

```

Export Data in the format of INSERT statements

-- customer --

INSERT INTO customer

(Customer_ID,Fname,Lname,Gender,Tel,Email,CRank,Address,Postcode,City,County)

VALUES (1111, 'Inwza', 'Aekwa',
'M','0123456789','inwza@gmail.com','Gold','123/157','10451','Bangkok','Thailand');

INSERT INTO customer

(Customer_ID,Fname,Lname,Gender,Tel,Email,CRank,Address,Postcode,City,County)

VALUES (1112, 'Siripong', 'phiwkhaw',
'M','45456464','inwza55@gmail.com','silver','44/157','44400','Bangkok','Thailand');

INSERT INTO customer

(Customer_ID,Fname,Lname,Gender,Tel,Email,CRank,Address,Postcode,City,County)

VALUES (1113, 'Anuwat', 'Pattanaajan',
'M','54645464','inwza44@gmail.com','Gold','113/157','54460','Bangkok','Thailand');

INSERT INTO customer

(Customer_ID,Fname,Lname,Gender,Tel,Email,CRank,Address,Postcode,City,County)

VALUES (1114, 'Aritach', 'krudminburee',
'F','04544644','inwza33@gmail.com','bronze','123/00','23232','Bangkok','Thailand');

INSERT INTO customer

(Customer_ID,Fname,Lname,Gender,Tel,Email,CRank,Address,Postcode,City,County)

VALUES (1115, 'Kittayod', 'wongwuttinun',
'F','1315416161','inwza12@gmail.com','silver','12/57','77777','Bangkok','Thailand');

-- hotel --

INSERT INTO Hotel (Hotel_ID,HotelName,Address,HotelDesc,City,Country, Rating)

VALUES (2000,'Diamond Hotel','952/142','7 floors','Bangkok','Thailand',3.6);

-- roomtype --

INSERT INTO roomtype (Roomtype_id,Roomtype,RoomPrice)

VALUES (2001, 'Duo Room', 3000);

INSERT INTO roomtype (Roomtype_id,Roomtype,RoomPrice)

VALUES (2002, 'Single Room', 3000);

-- hotelroom --

INSERT INTO hotelroom

(RoomNo,Roomstatus,Hotel_Hotel_id,roomtype_Roomtype_id)

VALUES ('2201', 'Available', 2000,2001);

INSERT INTO hotelroom

(RoomNo,Roomstatus,Hotel_Hotel_id,roomtype_Roomtype_id)

VALUES ('2202', 'Unavailable', 2000,2002);

-- resort --

INSERT INTO Resort

(Resort_ID,ResortPrice,ResortDesc,ResortName,Address,Country,Rating,City)

```
VALUES (3001,2700,'4 bedroom 3 bathroom','Ruby
Resort','875/5452','Thailand',3.6,'Bangkok');
```

```
INSERT INTO Resort
(Resort_ID,ResortPrice,ResortDesc,ResortName,Address,Country,Rating,City)
VALUES (3002,2800,'3 bedroom 2 bathroom','Sapphire
Resort','625/5415','Thailand',3.5,'Bangkok');
```

```
INSERT INTO Resort
(Resort_ID,ResortPrice,ResortDesc,ResortName,Address,Country,Rating,City)
VALUES (3003,2500,'2 bedroom 1 bathroom','Emerald
Resort','455/5461','Thailand',3.5,'Bangkok');
```

```
-- promotion --
```

```
INSERT INTO
promotion(Promotion_ID,Promotion_name,PromotionStartDate,PromotionEndD
ate,PromotionCondition,PromotionDetails)
VALUES (9000, '2.6%', '2021-09-15', '2021-12-27','Rank Silver or above','discount
2.6%');
```

```
-- booking --
```

```
INSERT INTO booking
(Booking_ID,BookingDate,BookingStatus,ArriveDate,DepartDate,NumOfChild,Nu
mOfAdult,Hotelroom_RoomNo,Resort_Resort_ID )
VALUES (4001,'2021-09-30','Book
succeeded','2021-10-01','2021-10-03',1,2,'2201',null);
```

INSERT INTO booking

(Booking_ID,BookingDate,BookingStatus,ArriveDate,DepartDate,NumOfChild,NumOfAdult,Hotelroom_RoomNo,Resort_Resort_ID)

VALUES (4002,'2021-09-28','Book
succeeded','2021-10-02','2021-10-06',1,0,'2202',null);

INSERT INTO booking

(Booking_ID,BookingDate,BookingStatus,ArriveDate,DepartDate,NumOfChild,NumOfAdult,Hotelroom_RoomNo,Resort_Resort_ID)

VALUES (4003,'2021-09-20','Book
unsucceeded','2021-10-01','2021-10-04',4,3,null,3001);

INSERT INTO booking

(Booking_ID,BookingDate,BookingStatus,ArriveDate,DepartDate,NumOfChild,NumOfAdult,Hotelroom_RoomNo,Resort_Resort_ID)

VALUES (4004,'2021-09-24','Book
unsucceeded','2021-10-01','2021-10-03',5,0,null,3002);

INSERT INTO booking

(Booking_ID,BookingDate,BookingStatus,ArriveDate,DepartDate,NumOfChild,NumOfAdult,Hotelroom_RoomNo,Resort_Resort_ID)

VALUES (4005,'2021-09-21','Book
succeeded','2021-09-29','2021-10-01',2,1,null,3003);

-- receipt --

INSERT INTO receipt

(Receipt_ID,PaymentDate,ReceiptStatus,Price,Promotion_ID,booking_Booking_ID
,customer_Customer_ID)

VALUES (7001, '2021-10-20', 'paided',2985,9000,4001,1111);

INSERT INTO receipt

(Receipt_ID,PaymentDate,ReceiptStatus,Price,Promotion_ID,booking_Booking_ID
,customer_Customer_ID)

VALUES (7002, '2021-10-21', 'paided',2700,null,4002,1112);

INSERT INTO receipt

(Receipt_ID,PaymentDate,ReceiptStatus,Price,Promotion_ID,booking_Booking_ID
,customer_Customer_ID)

VALUES (7003, '2021-10-22', 'not paided',2800,null,4003,1113);

INSERT INTO receipt

(Receipt_ID,PaymentDate,ReceiptStatus,Price,Promotion_ID,booking_Booking_ID
,customer_Customer_ID)

VALUES (7004, '2021-10-23', 'paided',2700,null,4004,1114);

INSERT INTO receipt

(Receipt_ID,PaymentDate,ReceiptStatus,Price,Promotion_ID,booking_Booking_ID
,customer_Customer_ID)

VALUES (7005, '2021-10-24', 'cancel',2500,null,4005,1115);