DATABASE SYSTEM

Aster Hotel

On Friday, 1:00 - 5:00 PM.
Section 5
2564/CSB

1. 6004062660023   Jirarat Apikireewong
2. 6204062660021   Jetsada Puthong
3. 6204062663021   Chissanupong Bunnag

Table of contents

1. ER definition

Aster hotel are the hotel that have many branches in Thailand for such as Chiangmai Bangkok and etc. For each of the branches must collected the data of hotels that have Branch_ID, Address and B_Phone_number, In the part of each the employee the branches must collect the data information E_Name , E_Surname , position , salary , Employee_ID , date of birth (DOB) and E_Phone_number. For each employee must enroll by 1 position only and in 1 position can have multiple employees. Every employee are controlled by manager and manager are the part of employee. 1 employee can be service many customer and 1 customer can take service by many employee.The Customer must add there information to our branch for collected data when they using our service such as Customer_ID,Check_in , Check_out , C_Name , C_Surname, and C_Phone_number. 1 Branch consist of many room and each of 1 room consist of by 1 branches.The Branches of hotel have many types of room to offer the customer such as single room , duo room , deluxe room and pool side view. For each room must have Type_room, Room_number and Price. 1 Customer can booking many room and in each room can be booking by many customer.In the booking part it will show the date check in and out and booking id to make customer know when they need to come stay. When customer payment booking succeed the bill and key room will be add to customer inventory, in the bill it have Bill_number ,Total_Price and Date_time. each customer can have only payment in the single time and 1 bills belong to 1 customer only.

2. Functional & Non Functional Requirement

Functional Requirement	Non Functional Requirement
Customer can select room type	Record the data of customer and employee when registration
Customer and Employee can register to the system	Update rooms of service that it empty or in queue
Show rooms of service that it empty or in queue	Collect the total money of each day for each branch
Show the information of each room type	The system must not be crashing
Show the information of customer service gifts	The system must available of all time

3. System boundary diagram
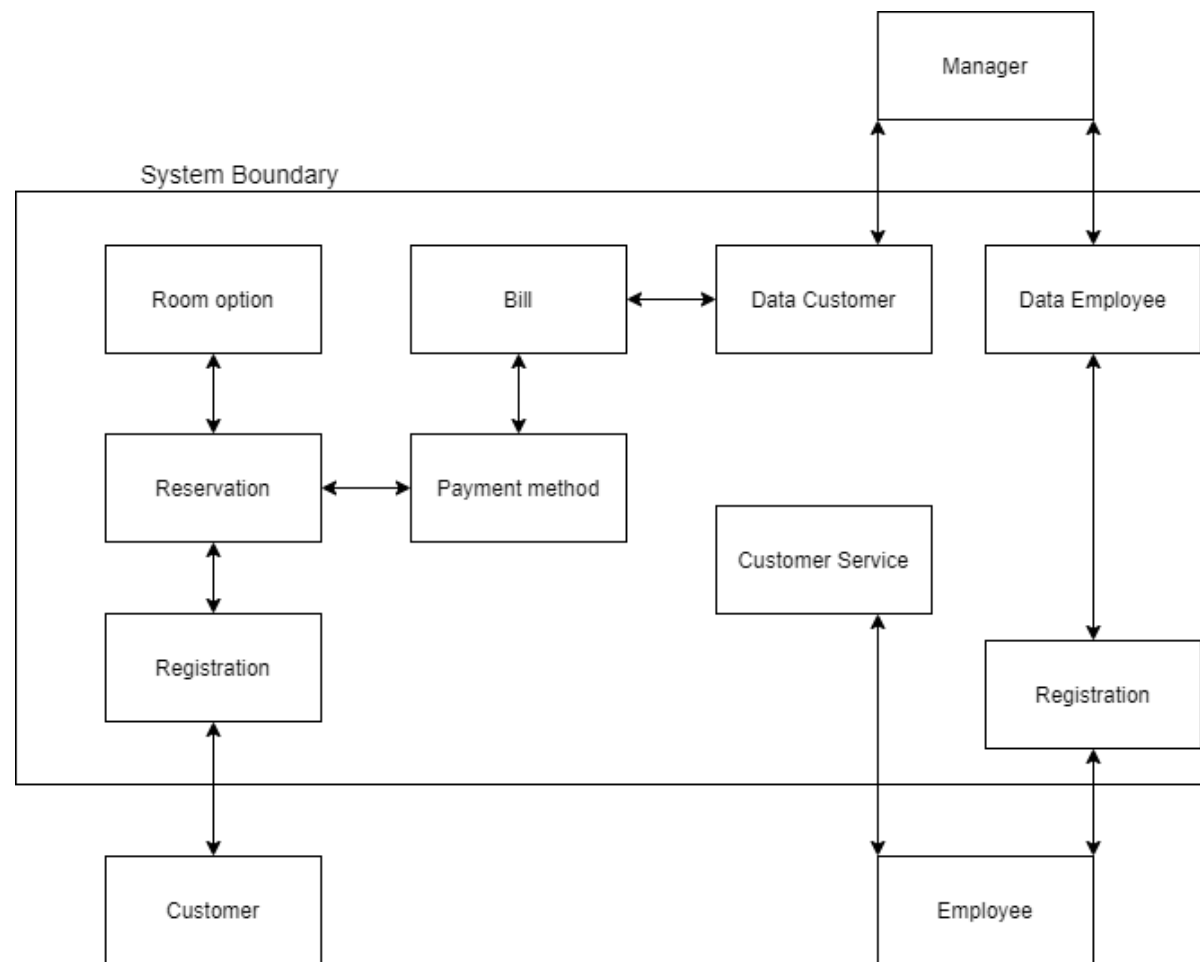
4. ER diagram
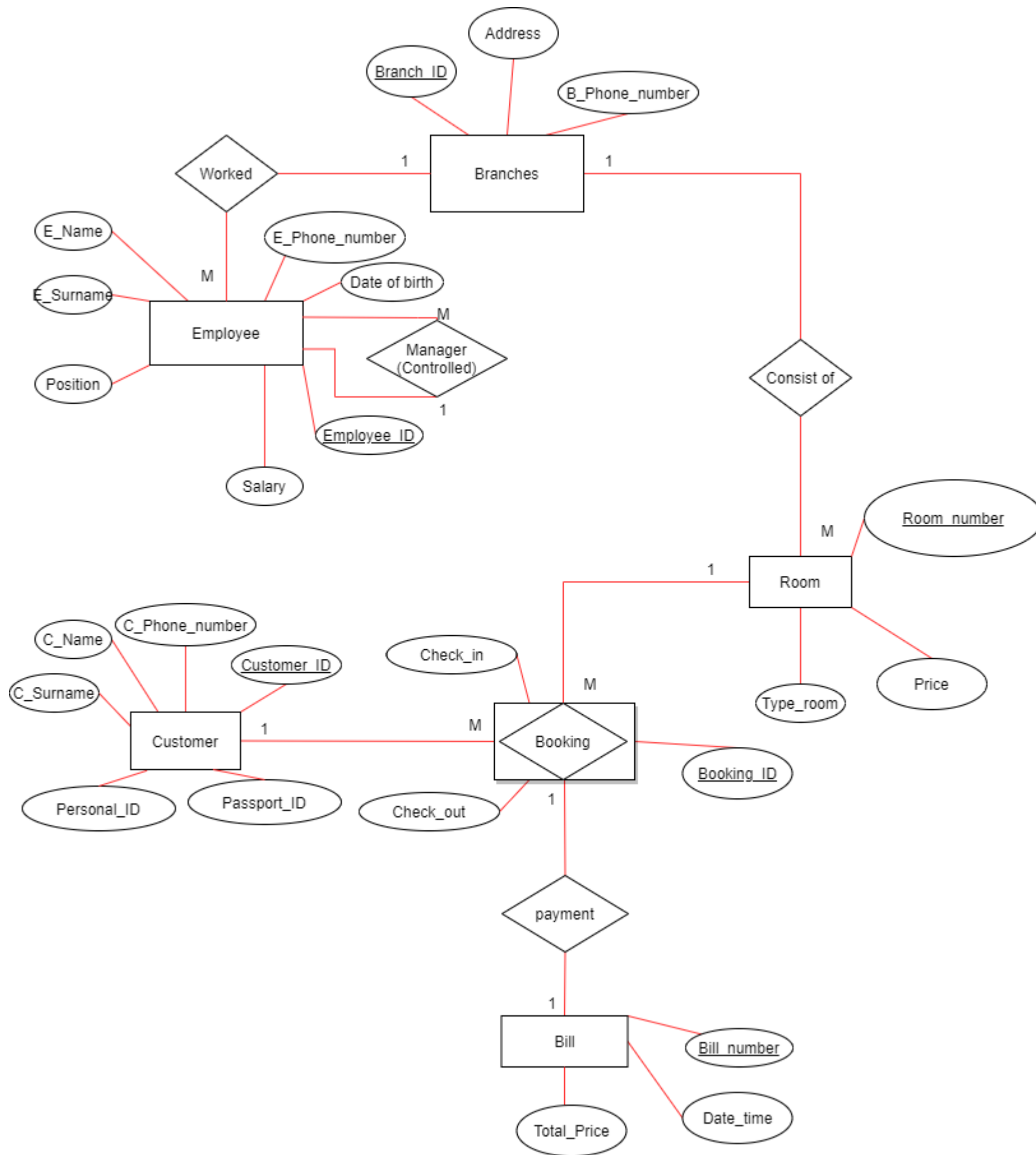
5. Relational schema

Branches {Branch_ID, Address, B_Phone_number}
PK : {Branch_ID}
FK : No

Room {Room _number, Branch_ID ,Type_room, Price}
PK : {Room _number}
FK : {Branch_ID} link to Branches Table.

Employee {Employee_ID, {E_Name,E_Surname}, Dateofbirth, E_Phone_number, Salary, Position, Branch_ID, Manager_ID}
Primary key : {Employee_ID}
Foreign key : {Branch_ID} link to Branch table.
             {Manager_ID} link to Employee table.

Customer {Customer_ID, Personal_ID, Passport_ID {C_Name,C_Surname}, C_Phone_number}
PK : {Customer_ID}
FK : No

Bill {Bill_number, Date_time , Total_Price}
PK : Bill_number
FK : No

Booking {Booking_ID, Customer_ID, Room _number, Check_in, Check_out}
Primary key : {Booking_ID}
Foreign key : {Customer_ID} link to Customer table.
             {Room_number} link to Room table.

6. Database normalization

6.1 Branch Table

Branch_ID	Branch_Address	B_Phone_number
C001	164 Khaosan Road Bangkok	02-215-4333
C002	1542 Ramkhamhaeng Road Bangkok	02-793-5123
C003	55/4 Rattanathibet Road Nonthaburi	02-213-4321
N004	17/9 Tha Phae Road Chiang Mai	02-852-6548
N005	725/3 Rat Burana Road Chiang Rai	02-459-9214
N006	442 Sumon Thewarat Road Nan	02-945-1258
S007	2/60 Permphol Road Ranong	02-147-1478
S008	42/17 Charoen Sub Road Krabi	02-258-3214
S009	475 Saiburi Road Songkhla	02-021-0214
NE010	182 Srichan Road Khon Kaen	02-213-0014
NE011	408/9 Wareeratchadet Road Yasothon	02-541-1004
NE012	43 Pho Si Road Udon Thani	02-050-5040

1NF:

Branch_ID	Branch_Address	B_Phone_number

Primary key : {Branch_ID}

Foreign key : No

2NF: No Partial dependence

Branch(Branch_ID, Branch_Address, B_Phone_number)

Primary key : {Branch_ID}

Foreign key : No

3NF: No Transitive dependence

Branch(Branch_ID, Branch_Address, B_Phone_number)

Primary key : {Branch_ID}

Foreign key : No

6.2 Employee Table

Employee_ID	E_Name	E_Surname	E_Phone_number	Date of birth	Position	Salary	Branch_ID	Manager_ID
E10001	Tanthai	Phupanya	073-554-0011	01/09/1949	Room Service	18,000.00	C002	E10003
E10002	Suyada	Jaidee	081-789-0458	21/05/1990	Front Desk	20,000.00	C002	E10003
E10003	Preetrib	Fongdou	088-888-9524	31/12/1980	Hotel General	35,000.00	C003	null
E20001	Rungroat	Markmee	089-459-5201	10/01/1989	Housekeeping	17,000.00	N004	E10003
E20002	Suchadee	jutarat	075-213-5110	23/05/2000	Waiting Staff	20,000.00	N006	E10003
E20003	Longdee	Meejai	061-161-6370	18/06/2000	Hotel Porter	15,000.00	N006	E10003
E30001	Phisong	Arwon	071-782-9404	23/10/1974	Front Desk	20,000.00	S007	E10003
E30002	Somsee	Meechoke	075-257-8099	7/02/1960	Housekeeping	17,000.00	S008	E10003
E30003	Patakorn	Pongpiteg	075-249-4110	8/01/1980	Maintenance	25,000.00	S009	E10003
E40001	Chulakarn	Jaijitmon	079-940-0497	30/09/1981	Room Service	18,000.00	NE010	E10003
E40002	Chulakorn	Jaijetmon	077-048-2322	01/09/1973	Front Desk	20,000.00	C002	E10003
E40003	Hasiwat	Kaewdee	081-808-2197	23/05/2000	Accounting	30,000.00	C002	E10003

1NF:

Employee_ID	Name	Surname	Phone_number	Date of birth	Position	Salary	Branch_ID	Manager_ID

Primary key : {Employee_ID}

Foreign key : {Branch_ID} link to Branch table.

{Manager_ID} link to Employee table.

2NF: No Partial dependence

Employee(Employee_ID, E_Name, E_Surname, E_Phone_number, Date of birth, Position, Salary, BranchID , Manager_ID)

Primary key : {Employee_ID}

Foreign key : {Branch_ID} link to Branch table.

{Manager_ID} link to Employee table.

3NF: No Transitive dependence

Employee(Employee_ID, E_Name, E_Surname, E_Phone_number, Date of birth, Position, Salary, BranchID , Manager_ID)

Primary key : {Employee_ID}

Foreign key : {Branch_ID} link to Branch table.

{Manager_ID} link to Employee table.

6.3 Customer Table

Customer_ID	Personal_ID	Passport_ID	C_Name	C_Surname	C_Phone_number
C0000001	1825553634408	null	Suda	Aromdee	081-428-0909
C0000002	5363578858551	null	Nin	Kasamsun	081-526-8900
C0000003	7495450214522	null	Sawat	Jurangkool	075-321-4150
C0000004	9180993611772	null	Jettrin	Sunthorn	083-258-0015
C0000005	7463239716443	null	Kulpat	Benjawan	079-985-3015
C0000006	4321653876527	null	Somwang	Klinpraneet	061-577-8788
C0000007	null	A123461	Kasidid	Yodsuwan	063-144-2242
C0000008	null	A473629	Chalerm	Panyarachun	063-616-6466
C0000009	null	B273492	Bumpen	Ratanarak	089-999-9968
C0000010	null	B583821	Choochai	Kongpaisarn	087-889-2589
C0000011	null	A846200	Koradol	Rattanakosin	087-478-2147
C0000012	null	A473294	Pawin	Wattana	080-133-9919

1NF:

Customer_ID	Personal_ID	Passport_ID	C_Name	C_Surname	Phone_number

Primary key : {Customer_ID}

Foreign key : No

2NF: No Partial dependence

Customer(Customer_ID, Personal_ID, Passport_ID, C_Name, C_Surname, C_Phone_number)

Primary key : {Customer_ID}

Foreign key : No

3NF: No Transitive dependence

Customer(Customer_ID, Personal_ID, Passport_ID, C_Name, C_Surname, C_Phone_number)

Primary key : {Customer_ID}

Foreign key : No

6.4 Bill Table

Bill_number	Date_time	Total_Price
B300001	18/02/2021	18,000.00
B300002	20/02/2021	4,250.00
B300003	15/05/2021	4,500.00
B300004	15/05/2021	36,000.00
B300005	30/05/2021	234,000.00
B300006	01/08/2021	52,000.00
B300007	18/08/2021	22,500.00
B300008	21/10/2021	8,250.00
B300009	18/10/2021	49,500.00
B300010	30/11/2021	25,500.00
B300011	21/12/2021	182,000.00
B300012	31/12/2021	27,000.00

1NF :

Bill_number	Date_time	Total_Price

Primary key : {Bill_number}
Foreign key : No

2NF : No Partial Dependence.

Bill(Bill_number, Date_time, Total_Price)

Primary key : {Bill_number}
Foreign key : No

3NF : No Transitive dependence

Bill(Bill_number, Date_time, Total_Price)

Primary key : {Bill_number}
Foreign key : No

6.5 Room Table

Room_number	Branch_ID	Type_room	Price
A001	C001	Single Bed Room	750.00
A002	C001	Single Bed Room	750.00
A003	C001	Double Bed Room	1500.00
A004	C001	Double Bed Room	1500.00
A005	C001	Quad Bed Room	4000.00
A006	C001	Quad Bed Room	4000.00
A007	C001	Queen Bed Room	7500.00
A008	C001	Queen Bed Room	7500.00
A009	C001	King Bed Room	8500.00
A010	C001	King Bed Room	8500.00
A011	C001	Executive Suite Bedroom	12,500.00
A012	C001	Executive Suite Bedroom	12,500.00
A013	C001	President Suit Bedroom	13,000.00
A014	C001	President Suit Bedroom	13,000.00
A015	C001	Cabana Bed Room	4000.00
A016	C001	Cabana Bed Room	4000.00
A017	C001	Poolside view Bed Room	6500.00
A018	C001	Poolside view Bed Room	6500.00
A019	C001	Accessible Bed Room	4500.00
A020	C001	Accessible Bed Room	4500.00
A021	C001	Murphy Bed Room	1500.00
A022	C001	Murphy Bed Room	1500.00
A023	C001	Smoking Bed Room	850.00
A024	C001	Smoking Bed Room	850.00
B001	N004	Single Bed Room	750.00
B002	N004	Single Bed Room	750.00
B003	N004	Double Bed Room	1500.00
B004	N004	Double Bed Room	1500.00
B005	N004	Quad Bed Room	4000.00
B006	N004	Quad Bed Room	4000.00
B007	N004	Queen Bed Room	7500.00
B008	N004	Queen Bed Room	7500.00
B009	N004	King Bed Room	8500.00
B010	N004	King Bed Room	8500.00
B011	N004	Executive Suite Bed Room	12,500.00

B012	N004	Executive Suite Bed Room	12,500.00
B013	N004	President Suit Bed Room	13,000.00
B014	N004	President Suit Bed Room	13,000.00
B015	N004	Cabana Bed Room	4000.00
B016	N004	Cabana Bed Room	4000.00
B017	N004	Poolside view Bed Room	6500.00
B018	N004	Poolside view Bed Room	6500.00
B019	N004	Accessible Bed Room	4500.00
B020	N004	Accessible Bed Room	4500.00
B021	N004	Murphy Bed Room	1500.00
B022	N004	Murphy Bed Room	1500.00
B023	N004	Smoking Bed Room	850.00
B024	N004	Smoking Bed Room	850.00

1NF :

Room_number	Branch_ID	Type_room	Price

Primary key : {Room_number}

Foreign Key : {Branch_ID} link to Branch table

2NF : No Partial dependence

Room(Room_number, Branch_ID, Type_room, Price)

Primary key : {Room_number}

Foreign Key : {Branch_ID} link to Branch table

3NF : No Transitive dependence

Room(Room_number, Branch_ID, Type_room, Price)

Primary key : {Room_number}

Foreign Key : {Branch_ID} link to Branch table

6.6 Booking Table

Booking_ID	Customer_ID	Room_number	Check_in	Check_out
BK0001	C0000001	A003	01/01/2022	07/01/2022
BK0002	C0000001	A004	01/01/2022	07/01/2022
BK0003	C0000002	B023	15/02/2022	20/02/2022
BK0004	C0000003	A004	13/04/2022	16/04/2022
BK0005	C0000004	B019	13/04/2022	20/04/2022
BK0006	C0000005	A014	13/04/2022	30/04/2022
BK0007	C0000006	A013	03/06/2022	05/06/2022
BK0008	C0000007	A020	10/06/2022	15/06/2022
BK0009	C0000008	B002	17/08/2022	28/08/2022
BK0010	C0000009	A019	20/09/2022	01/10/2022
BK0011	C0000010	B004	11/11/2022	28/11/2022
BK0012	C0000011	B013	01/12/2022	15/12/2022
BK0013	C0000001	A003	23/12/2022	01/01/2023
BK0014	C0000001	A004	23/12/2022	01/01/2023

1NF:

Booking_ID	Customer_ID	Room_number	Check_in	Check_out

Primary key : {Booking_ID}

Foreign key : {Customer_ID} link to Customer table.

{Room_number} link to Room table.

2NF: No Partial dependence

Booking(Booking_ID, Customer_ID, Room_number,Check_in, Check_out)

Primary key : {Booking_ID}

Foreign key : {Customer_ID} link to Customer table.

{Room_number} link to Room table.

3NF: No Transitive dependence

Booking(Booking_ID, Customer_ID, Room_number,Check_in, Check_out)

Primary key : {Booking_ID}

Foreign key : {Customer_ID} link to Customer table.

{Room_number} link to Room table.

7. Data directory

7.1 Branch table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Branch_ID	Branch's id example "C001"	VARCHAR(5)	xxxxx	Y	PK	
Address	Branch's address	VARCHAR (70)	xxxxxxxxxx	Y		
B_Phone_number	Branch's phone number	VARCHAR(12)	xx-xxx-xxxx	Y		

7.2 Room table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Room_number	Room number example "A001"	CHAR(4)	xxxx	Y	PK	
Branch_ID	Branch's id example "C001"	VARCHAR(5)	xxxxx	Y	FK	Branch table
Type_room	Room type	VARCHAR(50)	xxxxxxxxxx	Y		
Price	Room's price	DECIMAL(10, 2)	xxxxxxxxx.xx	Y		

7.3 Employee table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Employee_ID	Employee's ID	CHAR(6)	xxxxxx	Y	PK	
E_Name	Employee's name	VARCHAR(50)	xxxxxxxxxx	Y		
E_Surname	Employee's surname	VARCHAR(50)	xxxxxxxxxx	Y		
Dateofbirth	Date of birth	DATE	yyyy-mm-dd	Y		
E_Phone_number	Employee's phone number	VARCHAR(13)	xxx-xxx-xxxx	Y		
Salary	Employee's salary	DECIMAL(7,2)	xxxxxxx.xx			
Position	Employee's position	VARCHAR(50)	xxxxxxxxxx			
Branch_ID	Branch's id example "C001"	VARCHAR(5)	xxxxx		FK	Branch table
Manager_ID	Manager's id	CHAR(6)	xxxxxx		FK	Employee table

7.4 Customer table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Customer_ID	Customer's id	CHAR(8)	xxxxxxxx	Y	PK	
Personal_ID	Customer's personal id	CHAR(13)	xxxxxxxxxxxxx			
Passport_ID	Customer's passport id	CHAR(7)	xxxxxxx			
C_Name	Customer's name	VARCHAR(50)	xxxxxxxxxx	Y		
C_Surname	Customer's surname	VARCHAR(50)	xxxxxxxxxx	Y		
C_Phone_number	Customer's phone number	VARCHAR(13)	xxx-xxx-xxxx	Y		

7.5 Bill Table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Bill_number	Bill's number	CHAR(7)	xxxxxxx	Y	PK	
Date_time	Bill's date	DATE	yyyy-mm-dd			
Total_Price	Total price of service	DECIMAL(10, 2)	xxxxxxxxxx.xx			

7.6 Booking table

ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	REQUIRED	PK OR FK	FK REFERENCED TABLE
Booking_ID	Booking's id	CHAR(6)	xxxxxx	Y	PK	
Customer_ID	Customer's id	CHAR(8)	xxxxxxxx		FK	Customer table
Room_number	Room number	CHAR(4)	xxxx		FK	Room table
Check_in	Date of check in	DATE	yyyy-mm-dd			
Check_out	Date of check out	DATE	yyyy-mm-dd			

8. Relational Algebra

8.1 Find the Branch address where the elder employee who works in the Aster hotel which required more that 60 years old.

1.1) $A = \sigma_{\text{Date of birth} < \text{'01/01/1961'}}$ (Employee $\bowtie_{\text{Employee.Branch\_ID = Branch.Branch\_ID}}$ Branch)

Employ ee_ID	E_Nam e	E_Surna me	E_Phone _number	Date of birth	Positio n	Salary	Branc h_ID	Manage r_ID	Branc h_ID	Branch_Address	B_Phone _number
E1000 1	Tanthai	Phupany a	073-554-0011	01/09/194 9	Room Servic e	18,000. 00	C002	E10003	C002	1542 Ramkhamhaeng Road Bangkok	02-793-51 23
E3000 2	Somsee	Meechok e	075-257-8099	7/02/1960	House keepin g	17,000. 00	S008	E10003	S008	42/17 Charoen Sub Road Krabi	02-258-32 14

1.2) $\pi_{\text{Branch\_Address}}$ (A)

Branch_Address
1542 Ramkhamhaeng Road Bangkok
42/17 Charoen Sub Road Krabi

8.2 Find the customer name that reserve the room that have price less than 2,000 at branch number is N004

2.1) $\sigma_{\text{Price} < \text{'2,000'}}$ (Room)

Room_number	Branch_ID	Type_room	Price
A001	C001	Single Bed Room	750.00
A002	C001	Single Bed Room	750.00
A003	C001	Double Bed Room	1500.00
A004	C001	Double Bed Room	1500.00
A021	C001	Murphy Bed Room	1500.00
A022	C001	Murphy Bed Room	1500.00
A023	C001	Smoking Bed Room	850.00
A024	C001	Smoking Bed Room	850.00
B001	N004	Single Bed Room	750.00
B002	N004	Single Bed Room	750.00
B003	N004	Double Bed Room	1500.00
B004	N004	Double Bed Room	1500.00
B021	N004	Murphy Bed Room	1500.00
B022	N004	Murphy Bed Room	1500.00
B023	N004	Smoking Bed Room	850.00
B024	N004	Smoking Bed Room	850.00

2.2) $\sigma_{Branch\_ID = 'N004'}$ ($\sigma_{Price < '2,000'}$ (Room))

Room_number	Branch_ID	Type_room	Price
B001	N004	Single Bed Room	750.00
B002	N004	Single Bed Room	750.00
B003	N004	Double Bed Room	1500.00
B004	N004	Double Bed Room	1500.00
B021	N004	Murphy Bed Room	1500.00
B022	N004	Murphy Bed Room	1500.00
B023	N004	Smoking Bed Room	850.00
B024	N004	Smoking Bed Room	850.00

2.3) A = π_{Room_number}($\sigma_{Branch\_ID = 'N004'}$($\sigma_{Price < '2,000'}$ (Room $\bowtie_{Room.Room\_number = Booking.Room\_number}$ Booking)))

Room_number
B001
B002
B003
B004
B021
B022
B023
B024

2.4) B = $\pi_{Customer\_ID}$ (A)

Customer_ID
C0000002
C0000008
C0000010

2.5) π_{C_Name} ((B)$\bowtie$Booking.Customer_ID = Customer.Customer_ID Customer)

C_Name
Nin
Chalerm
Choochai

8.3 Find the type of room that has been reserved by Customer ID = C0000005

3.1) A = $\sigma_{\text{Customer\_ID = C0000005}}$ (Booking $\bowtie_{\text{Booking.Room\_number = Room.Room\_number}}$ Room)

Booking_ID	Customer_ID	Room_number	Check_in	Check_out	Room_number	Branch_ID	Type_room	Price
BK0006	C0000005	A014	13/04/2022	30/04/2022	A014	C001	President Suit Bedroom	13,000.00

3.2) $\pi_{\text{Type\_room}}$ (A)

Type_room
President Suit Bedroom

8.4 Find the customer name and surname that reserve the President Suit Bedroom at branch number is C001

4.1) $\sigma_{\text{Type\_room = 'President Suit Bedroom'}}$ (Room)

Room_number	Branch_ID	Type_room	Price
A013	C001	President Suit Bedroom	13,000.00
A014	C001	President Suit Bedroom	13,000.00
B013	N004	President Suit Bedroom	13,000.00
B014	N004	President Suit Bedroom	13,000.00

4.2) $\sigma_{\text{Branch\_ID = 'C001'}}$ ($\sigma_{\text{Type\_room = 'President Suit Bedroom'}}$ (Room))

Room_number	Branch_ID	Type_room	Price
A013	C001	President Suit Bedroom	13,000.00
A014	C001	President Suit Bedroom	13,000.00

4.3) A = $\pi_{\text{Room\_number}}$ ($\sigma_{\text{Branch\_ID = 'C001'}}$($\sigma_{\text{Type\_room = "President Suit Bedroom'}}$ (Room $\bowtie_{\text{Room.Room\_number = Booking.Room\_number}}$ Booking)))

Room_number
A013
A014

4.4) B = $\pi_{\text{Customer\_ID}}$ ((A) $\bowtie_{\text{Booking.Customer\_ID = Customer.Customer\_ID}}$ Customer)

Customer_ID
C0000005
C0000006

4.5) $\pi_{C\_Name,\ C\_Surname}$ (B)

C_Name	C_Surname
Kulpat	**Benjawan**
Somwang	**Klinpraneet**

9. SQL command

9.1 CREATE TABLE

9.1.1 Branch Table

CREATE TABLE Branch (
Branch_ID VARCHAR(5) NOT NULL,
Branch_Address VARCHAR(70),
B_Phone_number VARCHAR(12),
PRIMARY KEY (Branch_ID)
);

```
MariaDB [hotel]> CREATE TABLE Branch (
    -> Branch_ID VARCHAR(5) NOT NULL,
    -> Branch_Address VARCHAR(70),
    -> B_Phone_number VARCHAR(12),
    -> PRIMARY KEY (Branch_ID)
    -> );
Query OK, 0 rows affected (0.019 sec)
```

9.1.2 Employee Table

CREATE TABLE Employee (
Employee_ID CHAR(6) NOT NULL,
E_Name VARCHAR(50) NOT NULL,
E_Surname VARCHAR(50) NOT NULL,
E_Phone_number VARCHAR(13) NOT NULL,
Date_of_birth DATE NOT NULL,
Position VARCHAR(50),
Salary DECIMAL(7, 2),
Branch_ID VARCHAR(5),
Manager_ID CHAR(6),
PRIMARY KEY (Employee_ID),
FOREIGN KEY (Branch_ID) REFERENCES Branch (Branch_ID),
FOREIGN KEY (Manager_ID) REFERENCES Employee (Employee_ID)
);

```
MariaDB [hotel]> CREATE TABLE Employee (
    -> Employee_ID CHAR(6) NOT NULL,
    -> E_Name VARCHAR(50) NOT NULL,
    -> E_Surname VARCHAR(50) NOT NULL,
    -> E_Phone_number VARCHAR(13) NOT NULL,
    -> Date_of_birth DATE NOT NULL,
    -> Position VARCHAR(50),
    -> Salary DECIMAL(7, 2),
    -> Branch_ID VARCHAR(5),
    -> Manager_ID CHAR(6),
    -> PRIMARY KEY (Employee_ID),
    -> FOREIGN KEY (Branch_ID) REFERENCES Branch (Branch_ID),
    -> FOREIGN KEY (Manager_ID) REFERENCES Employee (Employee_ID)
    -> );
Query OK, 0 rows affected (0.015 sec)
```
TABLE Customer

9.1.3 Customer Table

```
CREATE TABLE Customer (
Customer_ID CHAR(8) NOT NULL,
Personal_ID CHAR(13),
Passport_ID CHAR(7),
C_Name VARCHAR(50) NOT NULL,
C_Surname VARCHAR(50) NOT NULL,
C_Phone_number VARCHAR(13) NOT NULL,
PRIMARY KEY (Customer_ID)
);
```

```
MariaDB [hotel]> CREATE TABLE Customer (
    -> Customer_ID CHAR(8) NOT NULL,
    -> Personal_ID CHAR(13),
    -> Passport_ID CHAR(7),
    -> C_Name VARCHAR(50) NOT NULL,
    -> C_Surname VARCHAR(50) NOT NULL,
    -> C_Phone_number VARCHAR(13) NOT NULL,
    -> PRIMARY KEY (Customer_ID)
    -> );
Query OK, 0 rows affected (0.025 sec)
```

9.1.4 Bill table

```
CREATE TABLE Bill (
Bill_number CHAR(7) NOT NULL,
Date_time DATE,
Total_Price DECIMAL(10, 2),
PRIMARY KEY (Bill_number)
);
```

```
MariaDB [hotel]> CREATE TABLE Bill (
    -> Bill_number CHAR(7) NOT NULL,
    -> Date_time DATE,
    -> Total_Price DECIMAL(10, 2),
    -> PRIMARY KEY (Bill_number)
    -> );
Query OK, 0 rows affected (0.026 sec)
```

9.1.5 Room table

```
CREATE TABLE Room (
Room_number CHAR(4) NOT NULL,
Branch_ID VARCHAR(5),
Type_room VARCHAR(50) NOT NULL,
Price DECIMAL(10, 2),
PRIMARY KEY (Room_number),
FOREIGN KEY (Branch_ID) REFERENCES Branch (Branch_ID)
);
```

```
MariaDB [hotel]> CREATE TABLE Room (
    -> Room_number CHAR(4) NOT NULL,
    -> Branch_ID VARCHAR(5),
    -> Type_room VARCHAR(50) NOT NULL,
    -> Price DECIMAL(10, 2),
    -> PRIMARY KEY (Room_number),
    -> FOREIGN KEY (Branch_ID) REFERENCES Branch (Branch_ID)
    -> );
Query OK, 0 rows affected (0.022 sec)
```

9.1.6 Booking table

```
CREATE TABLE Booking (
Booking_ID CHAR(6) NOT NULL,
Customer_ID CHAR(8),
Room_number CHAR(4),
Check_in DATE,
Check_out DATE,
PRIMARY KEY (Booking_ID),
FOREIGN KEY (Customer_ID) REFERENCES Customer (Customer_ID),
FOREIGN KEY (Room_number) REFERENCES Room (Room_number)
);
```

```
MariaDB [hotel]> CREATE TABLE Booking (
    -> Booking_ID CHAR(6) NOT NULL,
    -> Customer_ID CHAR(8),
    -> Room_number CHAR(4),
    -> Check_in DATE,
    -> Check_out DATE,
    -> PRIMARY KEY (Booking_ID),
    -> FOREIGN KEY (Customer_ID) REFERENCES Customer (Customer_ID),
    -> FOREIGN KEY (Room_number) REFERENCES Room (Room_number)
    -> );
Query OK, 0 rows affected (0.028 sec)
```

9.2 INSERT DATA TO TABLE

9.2.1 Branch table

```
INSERT INTO Branch (Branch_ID, Branch_Address, B_Phone_number) VALUES
('C001','164 Khaosan Road Bangkok','02-215-4333'),
('C002','1542 Ramkhamhaeng Road Bangkok','02-793-5123'),
('C003','55/4 Rattanathibet Road Nonthaburi','02-213-4321'),
('N004','17/9 Tha Phae Road Chiang Mai','02-852-6548'),
('N005','725/3 Rat Burana Road Chiang Rai','02-459-9214'),
('N006','442 Sumon Thewarat Road Nan','02-945-1258'),
('S007','2/60 Permphol Road Ranong','02-147-1478'),
('S008','42/17 Charoen Sub Road Krabi','02-258-3214'),
('S009','475 Saiburi Road Songkhla','02-021-0214'),
('NE010','182 Srichan Road Khon Kaen','02-213-0014'),
('NE011','408/9 Wareeratchadet Road Yasothon','02-541-1004'),
('NE012','43 Pho Si Road Udon Thani','02-050-5040');
```

```
MariaDB [hotel]> INSERT INTO Branch (Branch_ID, Branch_Address, B_Phone_number) VALUES
    -> ('C001','164 Khaosan Road Bangkok','02-215-4333'),
    -> ('C002','1542 Ramkhamhaeng Road Bangkok','02-793-5123'),
    -> ('C003','55/4 Rattanathibet Road Nonthaburi','02-213-4321'),
    -> ('N004','17/9 Tha Phae Road Chiang Mai','02-852-6548'),
    -> ('N005','725/3 Rat Burana Road Chiang Rai','02-459-9214'),
    -> ('N006','442 Sumon Thewarat Road Nan','02-945-1258'),
    -> ('S007','2/60 Permphol Road Ranong','02-147-1478'),
    -> ('S008','42/17 Charoen Sub Road Krabi','02-258-3214'),
    -> ('S009','475 Saiburi Road Songkhla','02-021-0214'),
    -> ('NE010','182 Srichan Road Khon Kaen','02-213-0014'),
    -> ('NE011','408/9 Wareeratchadet Road Yasothon','02-541-1004'),
    -> ('NE012','43 Pho Si Road Udon Thani','02-050-5040');
Query OK, 12 rows affected (0.060 sec)
Records: 12  Duplicates: 0  Warnings: 0
```

9.2.2 Employee table

INSERT INTO Employee (Employee_ID, E_Name, E_Surname, E_Phone_number, Date_of_birth, Position, Salary, Branch_ID) VALUES
('E10001','Tanthai','Phupanya','073-554-0011','1949-09-01','Room Service','18000.00','C002'),
('E10002','Suyada','Jaidee','081-789-0458','1990-05-21','Front Desk ','20000.00','C002'),
('E10003','Preetrib','Fongdou','088-888-9524','1980-12-31','Hotel General','35000.00','C003'),
('E20001','Rungroat','Markmee','089-459-5201','1989-01-10','Housekeeping','17000.00','N004'),
('E20002','Suchadee','jutarat','075-213-5110','2000-05-23','Waiting Staff','20000.00','N006'),
('E20003','Longdee','Meejai','061-161-6370','2000-06-18','Hotel Porter','15000.00','N006'),
('E30001','Phisong','Arwon','071-782-9404','1974-10-23','Front Desk','20000.00','S007'),
('E30002','Somsee','Meechoke','075-257-8099','1960-02-07','Housekeeping','17000.00','S008'),
('E30003','Patakorn','Pongpiteg','075-249-4110','1980-01-08','Maintenance','25000.00','S009'),
('E40001','Chulakarn','Jaijitmon','079-940-0497','1981-09-30','Room Service','18000.00','NE010'),
('E40002','Chulakarn','Jaijetmon','077-048-2322','1973-09-01','Front Desk ','20000.00','C002'),
('E40003','Hasiwat','Kaewdee','081-808-2197','2000-05-23','Accounting','30000.00','C002');

UPDATE Employee SET Manager_ID = 'E10003' WHERE NOT Employee_ID = 'E10003';

```
MariaDB [hotel]> INSERT INTO Employee (Employee_ID, E_Name, E_Surname, E_Phone_number, Date_of_birth, Position, Salary,
Branch_ID) VALUES
    -> ('E10001','Tanthai','Phupanya','073-554-0011','1949-09-01','Room Service','18000.00','C002'),
    -> ('E10002','Suyada','Jaidee','081-789-0458','1990-05-21','Front Desk ','20000.00','C002'),
    -> ('E10003','Preetrib','Fongdou','088-888-9524','1980-12-31','Hotel General','35000.00','C003'),
    -> ('E20001','Rungroat','Markmee','089-459-5201','1989-01-10','Housekeeping','17000.00','N004'),
    -> ('E20002','Suchadee','jutarat','075-213-5110','2000-05-23','Waiting Staff','20000.00','N006'),
    -> ('E20003','Longdee','Meejai','061-161-6370','2000-06-18','Hotel Porter','15000.00','N006'),
    -> ('E30001','Phisong','Arwon','071-782-9404','1974-10-23','Front Desk','20000.00','S007'),
    -> ('E30002','Somsee','Meechoke','075-257-8099','1960-02-07','Housekeeping','17000.00','S008'),
    -> ('E30003','Patakorn','Pongpiteg','075-249-4110','1980-01-08','Maintenance','25000.00','S009'),
    -> ('E40001','Chulakarn','Jaijitmon','079-940-0497','1981-09-30','Room Service','18000.00','NE010'),
    -> ('E40002','Chulakarn','Jaijetmon','077-048-2322','1973-09-01','Front Desk ','20000.00','C002'),
    -> ('E40003','Hasiwat','Kaewdee','081-808-2197','2000-05-23','Accounting','30000.00','C002');
Query OK, 12 rows affected (0.003 sec)
Records: 12  Duplicates: 0  Warnings: 0

MariaDB [hotel]> UPDATE Employee SET Manager_ID = 'E10003' WHERE NOT Employee_ID = 'E10003';
Query OK, 11 rows affected (0.003 sec)
Rows matched: 11  Changed: 11  Warnings: 0
```

9.2.3 Customer table

INSERT INTO Customer (Customer_ID, Personal_ID, Passport_ID, C_Name, C_Surname, C_Phone_number) VALUES
('C0000001','1825553634408',NULL,'Suda','Aromdee','081-428-0909'),
('C0000002','5363578858551',NULL,'Nin','Kasamsun','081-526-8900'),
('C0000003','7495450214522',NULL,'Sawat','Jurangkool','075-321-4150'),
('C0000004','9180993611772',NULL,'Jettrin','Sunthorn','083-258-0015'),
('C0000005','7463239716443',NULL,'Kulpat','Benjawan','079-985-3015'),
('C0000006','4321653876527',NULL,'Somwang','Klinpraneet','061-577-8788'),
('C0000007',NULL,'A123461','Kasidid','Yodsuwan','063-144-2242'),
('C0000008',NULL,'A473629','Chalerm','Panyarachun','063-616-6466'),
('C0000009',NULL,'B273492','Bumpen','Ratanarak','089-999-9968'),
('C0000010',NULL,'B583821','Choochai ','Kongpaisarn','087-889-2589'),
('C0000011',NULL,'A846200','Koradol','Rattanakosin','087-478-2147'),
('C0000012',NULL,'A473294','Pawin','Wattana','080-133-9919');

```
MariaDB [hotel]> INSERT INTO Customer (Customer_ID, Personal_ID, Passport_ID, C_Name, C_Surname, C_Phone_number) VALUES
    -> ('C0000001','1825553634408',NULL,'Suda','Aromdee','081-428-0909'),
    -> ('C0000002','5363578858551',NULL,'Nin','Kasamsun','081-526-8900'),
    -> ('C0000003','7495450214522',NULL,'Sawat','Jurangkool','075-321-4150'),
    -> ('C0000004','9180993611772',NULL,'Jettrin','Sunthorn','083-258-0015'),
    -> ('C0000005','7463239716443',NULL,'Kulpat','Benjawan','079-985-3015'),
    -> ('C0000006','4321653876527',NULL,'Somwang','Klinpraneet','061-577-8788'),
    -> ('C0000007',NULL,'A123461','Kasidid','Yodsuwan','063-144-2242'),
    -> ('C0000008',NULL,'A473629','Chalerm','Panyarachun','063-616-6466'),
    -> ('C0000009',NULL,'B273492','Bumpen','Ratanarak','089-999-9968'),
    -> ('C0000010',NULL,'B583821','Choochai ','Kongpaisarn','087-889-2589'),
    -> ('C0000011',NULL,'A846200','Koradol','Rattanakosin','087-478-2147'),
    -> ('C0000012',NULL,'A473294','Pawin','Wattana','080-133-9919');
Query OK, 12 rows affected (0.003 sec)
Records: 12  Duplicates: 0  Warnings: 0
```

9.2.4 Bill table

INSERT INTO Bill (Bill_number, Date_time, Total_Price) VALUES
('B300001','2021-02-18','18000.00'),
('B300002','2021-02-20','4250.00'),
('B300003','2021-05-15','4500.00'),
('B300004','2021-05-15','36000.00'),
('B300005','2021-05-30','234000.00'),
('B300006','2021-08-01','52000.00'),
('B300007','2021-08-18','22500.00'),
('B300008','2021-10-21','8250.00'),
('B300009','2021-10-18','49500.00'),
('B300010','2021-11-30','25500.00'),
('B300011','2021-12-21','182000.00'),
('B300012','2021-12-31','27000.00');

```
MariaDB [hotel]> INSERT INTO Bill (Bill_number, Date_time, Total_Price) VALUES
    -> ('B300001','2021-02-18','18000.00'),
    -> ('B300002','2021-02-20','4250.00'),
    -> ('B300003','2021-05-15','4500.00'),
    -> ('B300004','2021-05-15','36000.00'),
    -> ('B300005','2021-05-30','234000.00'),
    -> ('B300006','2021-08-01','52000.00'),
    -> ('B300007','2021-08-18','22500.00'),
    -> ('B300008','2021-10-21','8250.00'),
    -> ('B300009','2021-10-18','49500.00'),
    -> ('B300010','2021-11-30','25500.00'),
    -> ('B300011','2021-12-21','182000.00'),
    -> ('B300012','2021-12-31','27000.00');
Query OK, 12 rows affected (0.002 sec)
Records: 12  Duplicates: 0  Warnings: 0
```

9.2.5 Room table

INSERT INTO Room (Room_number, Branch_ID, Type_room, Price) VALUES
('A001','C001','Single Bed Room','750.00'),
('A002','C001','Single Bed Room','750.00'),
('A003','C001','Double Bed Room ','1500.00'),
('A004','C001','Double Bed Room ','1500.00'),
('A005','C001','Quad Bed Room','4000.00'),
('A006','C001','Quad Bed Room','4000.00'),
('A007','C001','Queen Bed Room','7500.00'),
('A008','C001','Queen Bed Room','7500.00'),
('A009','C001','King Bed Room','8500.00'),
('A010','C001','King Bed Room','8500.00'),
('A011','C001','Executive Suite Bedroom','12500.00'),
('A012','C001','Executive Suite Bedroom','12500.00'),
('A013','C001','President Suit Bedroom','13000.00'),
('A014','C001','President Suit Bedroom','13000.00'),
('A015','C001','Cabana Bed Room','4000.00'),
('A016','C001','Cabana Bed Room','4000.00'),
('A017','C001','Poolside view Bed Room','6500.00'),
('A018','C001','Poolside view Bed Room','6500.00'),
('A019','C001','Accessible Bed Room','4500.00'),
('A020','C001','Accessible Bed Room','4500.00'),
('A021','C001','Murphy Bed Room','1500.00'),
('A022','C001','Murphy Bed Room','1500.00'),
('A023','C001','Smoking Bed Room','850.00'),
('A024','C001','Smoking Bed Room','850.00'),
('B001','N004','Single Bed Room','750.00'),

('B002','N004','Single Bed Room','750.00'),
('B003','N004','Double Bed Room ','1500.00'),
('B004','N004','Double Bed Room ','1500.00'),
('B005','N004','Quad Bed Room','4000.00'),
('B006','N004','Quad Bed Room','4000.00'),
('B007','N004','Queen Bed Room','7500.00'),
('B008','N004','Queen Bed Room','7500.00'),
('B009','N004','King Bed Room','8500.00'),
('B010','N004','King Bed Room','8500.00'),
('B011','N004','Executive Suite Bed Room','12500.00'),
('B012','N004','Executive Suite Bed Room','12500.00'),
('B013','N004','President Suit Bed Room','13000.00'),
('B014','N004','President Suit Bed Room','13000.00'),
('B015','N004','Cabana Bed Room','4000.00'),
('B016','N004','Cabana Bed Room','4000.00'),
('B017','N004','Poolside view Bed Room','6500.00'),
('B018','N004','Poolside view Bed Room','6500.00'),
('B019','N004','Accessible Bed Room','4500.00'),
('B020','N004','Accessible Bed Room','4500.00'),
('B021','N004','Murphy Bed Room','1500.00'),
('B022','N004','Murphy Bed Room','1500.00'),
('B023','N004','Smoking Bed Room','850.00'),
('B024','N004','Smoking Bed Room','850.00');

```
MariaDB [hotel]> INSERT INTO Room (Room_number, Branch_ID, Type_room, Price) VALUES
    -> ('A001','C001','Single Bed Room','750.00'),
    -> ('A002','C001','Single Bed Room','750.00'),
    -> ('A003','C001','Double Bed Room ','1500.00'),
    -> ('A004','C001','Double Bed Room ','1500.00'),
    -> ('A005','C001','Quad Bed Room','4000.00'),
    -> ('A006','C001','Quad Bed Room','4000.00'),
    -> ('A007','C001','Queen Bed Room','7500.00'),
    -> ('A008','C001','Queen Bed Room','7500.00'),
    -> ('A009','C001','King Bed Room','8500.00'),
    -> ('A010','C001','King Bed Room','8500.00'),
    -> ('A011','C001','Executive Suite Bedroom','12500.00'),
    -> ('A012','C001','Executive Suite Bedroom','12500.00'),
    -> ('A013','C001','President Suit Bedroom','13000.00'),
    -> ('A014','C001','President Suit Bedroom','13000.00'),
    -> ('A015','C001','Cabana Bed Room','4000.00'),
    -> ('A016','C001','Cabana Bed Room','4000.00'),
    -> ('A017','C001','Poolside view Bed Room','6500.00'),
    -> ('A018','C001','Poolside view Bed Room','6500.00'),
    -> ('A019','C001','Accessible Bed Room','4500.00'),
    -> ('A020','C001','Accessible Bed Room','4500.00'),
    -> ('A021','C001','Murphy Bed Room','1500.00'),
    -> ('A022','C001','Murphy Bed Room','1500.00'),
    -> ('A023','C001','Smoking Bed Room','850.00'),
    -> ('A024','C001','Smoking Bed Room','850.00'),
    -> ('B001','N004','Single Bed Room','750.00'),
    -> ('B002','N004','Single Bed Room','750.00'),
    -> ('B003','N004','Double Bed Room ','1500.00'),
    -> ('B004','N004','Double Bed Room ','1500.00'),
    -> ('B005','N004','Quad Bed Room','4000.00'),
    -> ('B006','N004','Quad Bed Room','4000.00'),
    -> ('B007','N004','Queen Bed Room','7500.00'),
    -> ('B008','N004','Queen Bed Room','7500.00'),
    -> ('B009','N004','King Bed Room','8500.00'),
    -> ('B010','N004','King Bed Room','8500.00'),
    -> ('B011','N004','Executive Suite Bed Room','12500.00'),
    -> ('B012','N004','Executive Suite Bed Room','12500.00'),
    -> ('B013','N004','President Suit Bed Room','13000.00'),
    -> ('B014','N004','President Suit Bed Room','13000.00'),
    -> ('B015','N004','Cabana Bed Room','4000.00'),
    -> ('B016','N004','Cabana Bed Room','4000.00'),
    -> ('B017','N004','Poolside view Bed Room','6500.00'),
    -> ('B018','N004','Poolside view Bed Room','6500.00'),
    -> ('B019','N004','Accessible Bed Room','4500.00'),
    -> ('B020','N004','Accessible Bed Room','4500.00'),
    -> ('B021','N004','Murphy Bed Room','1500.00'),
    -> ('B022','N004','Murphy Bed Room','1500.00'),
    -> ('B023','N004','Smoking Bed Room','850.00'),
    -> ('B024','N004','Smoking Bed Room','850.00');
Query OK, 48 rows affected (0.006 sec)
Records: 48  Duplicates: 0  Warnings: 0
```

9.2.6 Booking table

INSERT INTO Booking (Booking_ID, Customer_ID, Room_number, Check_in, Check_out) VALUES
('BK0001','C0000001','A003','2022-01-01','2022-01-07'),
('BK0002','C0000001','A004','2022-01-01','2022-01-07'),
('BK0003','C0000002','B023','2022-02-15','2022-02-20'),
('BK0004','C0000003','A004','2022-04-13','2022-04-16'),
('BK0005','C0000004','B019','2022-04-13','2022-04-20'),
('BK0006','C0000005','A014','2022-04-13','2022-04-30'),
('BK0007','C0000006','A013','2022-06-03','2022-06-05'),
('BK0008','C0000007','A020','2022-06-10','2022-06-15'),
('BK0009','C0000008','B002','2022-08-17','2022-08-28'),
('BK0010','C0000009','A019','2022-09-20','2022-10-01'),
('BK0011','C0000010','B004','2022-11-11','2022-11-28'),
('BK0012','C0000011','B013','2022-12-01','2022-12-15'),
('BK0013','C0000001','A003','2022-12-23','2023-01-01'),
('BK0014','C0000001','A004','2022-12-23','2023-01-01');

```
MariaDB [hotel]> INSERT INTO Booking (Booking_ID, Customer_ID, Room_number, Check_in, Check_out) VALUES
    -> ('BK0001', 'C0000001', 'A003', '2022-01-01', '2022-01-07'),
    -> ('BK0002', 'C0000001', 'A004', '2022-01-01', '2022-01-07'),
    -> ('BK0003', 'C0000002', 'B023', '2022-02-15', '2022-02-20'),
    -> ('BK0004', 'C0000003', 'A004', '2022-04-13', '2022-04-16'),
    -> ('BK0005', 'C0000004', 'B019', '2022-04-13', '2022-04-20'),
    -> ('BK0006', 'C0000005', 'A014', '2022-04-13', '2022-04-30'),
    -> ('BK0007', 'C0000006', 'A013', '2022-06-03', '2022-06-05'),
    -> ('BK0008', 'C0000007', 'A020', '2022-06-10', '2022-06-15'),
    -> ('BK0009', 'C0000008', 'B002', '2022-08-17', '2022-08-28'),
    -> ('BK0010', 'C0000009', 'A019', '2022-09-20', '2022-10-01'),
    -> ('BK0011', 'C0000010', 'B004', '2022-11-11', '2022-11-28'),
    -> ('BK0012', 'C0000011', 'B013', '2022-12-01', '2022-12-15'),
    -> ('BK0013', 'C0000001', 'A003', '2022-12-23', '2023-01-01'),
    -> ('BK0014', 'C0000001', 'A004', '2022-12-23', '2023-01-01');
Query OK, 14 rows affected (0.003 sec)
Records: 14  Duplicates: 0  Warnings: 0
```

10. ALGEBRA SQL

10.1. Find the Branch address where the elder employee who works in the Aster hotel which required more that 60 years old.

SELECT Branch_Address
FROM Branch
INNER JOIN Employee ON Branch.Branch_ID = Employee.Branch_ID
WHERE Employee.Date_of_birth < '1961-01-01';

```
MariaDB [hotel]> SELECT Branch_Address
    -> FROM Branch
    -> INNER JOIN Employee ON Branch.Branch_ID = Employee.Branch_ID
    -> WHERE Employee.Date_of_birth < '1961-01-01';
+------------------------------+
| Branch_Address               |
+------------------------------+
| 1542 Ramkhamhaeng Road Bangkok |
| 42/17 Charoen Sub Road Krabi   |
+------------------------------+
2 rows in set (0.003 sec)
```

10.2. Find the customer name that reserve the room that have price less than 2,000 at branch number is N004

SELECT C_Name
FROM Customer
INNER JOIN Booking ON Booking.Customer_ID = Customer.Customer_ID
INNER JOIN Room ON Room.Room_number = Booking.Room_number
WHERE Room.Price < 2000.00 AND Room.Branch_ID = 'N004';

```
MariaDB [hotel]> SELECT C_Name
    -> FROM Customer
    -> INNER JOIN Booking ON Booking.Customer_ID = Customer.Customer_ID
    -> INNER JOIN Room ON Room.Room_number = Booking.Room_number
    -> WHERE Room.Price < 2000.00 AND Room.Branch_ID = 'N004';
+---------+
| C_Name  |
+---------+
| Nin     |
| Chalerm |
| Choochai |
+---------+
3 rows in set (0.001 sec)
```

10.3. Find the type of room that has been reserved by Customer ID = C0000005

SELECT Type_room
FROM Room
INNER JOIN Booking ON Room.Room_number = Booking.Room_number
WHERE Booking.Customer_ID = 'C0000005';

```
MariaDB [hotel]> SELECT Type_room
    -> FROM Room
    -> INNER JOIN Booking ON Room.Room_number = Booking.Room_number
    -> WHERE Booking.Customer_ID = 'C0000005';
+----------------------+
| Type_room            |
+----------------------+
| President Suit Bedroom |
+----------------------+
1 row in set (0.002 sec)
```

10.4. Find the customer name and surname that reserve the President Suit Bedroom at branch number is C001

```
SELECT C_Name, C_Surname
FROM Customer
INNER JOIN Booking ON Booking.Customer_ID = Customer.Customer_ID
INNER JOIN Room ON Room.Room_number = Booking.Room_number
WHERE Room.Type_room = 'President Suit Bedroom' AND Room.Branch_ID = 'C001';
```

```
MariaDB [hotel]> SELECT C_Name, C_Surname
    -> FROM Customer
    -> INNER JOIN Booking ON Booking.Customer_ID = Customer.Customer_ID
    -> INNER JOIN Room ON Room.Room_number = Booking.Room_number
    -> WHERE Room.Type_room = 'President Suit Bedroom' AND Room.Branch_ID = 'C001';
+----------+------------+
| C_Name   | C_Surname  |
+----------+------------+
| Kulpat   | Benjawan   |
| Somwang  | Klinpraneet|
+----------+------------+
2 rows in set (0.002 sec)
```

11. SUBQUERY SQL

This command is use to find the employee name and surname that work at the same place as customer name "Nin Kasamsun" had reserves.

```
SELECT E_Name, E_Surname
FROM Employee
WHERE Branch_ID = (SELECT Branch_ID FROM Room
WHERE Room_number = (SELECT Room_number FROM Booking
WHERE Customer_ID = (SELECT Customer_ID FROM Customer WHERE C_Name = "Nin" AND C_Surname = "Kasamsun")));
```

```
MariaDB [hotel]> SELECT E_Name, E_Surname
    -> FROM Employee
    -> WHERE Branch_ID = (SELECT Branch_ID FROM Room
    -> WHERE Room_number = (SELECT Room_number FROM Booking
    -> WHERE Customer_ID = (SELECT Customer_ID FROM Customer WHERE C_Name = "Nin" AND C_Surname = "Kasamsun")));
+----------+-----------+
| E_Name   | E_Surname |
+----------+-----------+
| Rungroat | Markmee   |
+----------+-----------+
1 row in set (0.000 sec)
```

This command is use to find the customer name and surname that reserved room more than 3 times.

```
SELECT C_Name, C_Surname
FROM Customer
WHERE Customer_ID = (SELECT Customer_ID FROM Booking
GROUP BY Customer_ID HAVING COUNT(Customer_ID) > 3);
```

```
MariaDB [hotel]> SELECT C_Name, C_Surname
    -> FROM Customer
    -> WHERE Customer_ID = (SELECT Customer_ID FROM Booking
    -> GROUP BY Customer_ID HAVING COUNT(Customer_ID) > 3);
+--------+-----------+
| C_Name | C_Surname |
+--------+-----------+
| Suda   | Aromdee   |
+--------+-----------+
1 row in set (0.000 sec)
```

This command is use to find the customer name and surname that had reserved "Smoking Bed Room"

SELECT C_Name, C_Surname
FROM Customer
WHERE Customer_ID = (SELECT Customer_ID FROM Booking
WHERE Room_number IN (SELECT Room_number FROM Room
WHERE Type_room = "Smoking Bed Room"));

```
MariaDB [hotel]> SELECT C_Name, C_Surname
    -> FROM Customer
    -> WHERE Customer_ID = (SELECT Customer_ID FROM Booking
    -> WHERE Room_number IN (SELECT Room_number FROM Room
    -> WHERE Type_room = "Smoking Bed Room"));
+---------+-----------+
| C_Name  | C_Surname |
+---------+-----------+
| Nin     | Kasamsun  |
+---------+-----------+
1 row in set (0.000 sec)
```