

## CSCE – 5350

### Fundamentals of Database Systems

#### Group 18

#### PART 6: DB REFINEMENT(NORMALIZATION).

Name	Student Email
Drusya Chandra	DrusyaChandra@my.unt.edu
Fayaz Shareef Mohammad	FayazMohammad@my.unt.edu
Sai Swathi Paruchuri	sai-swathiparuchuri@my.unt.edu
Pravallika Bollavaram	pravallikabollavaram@my.unt.edu
Sweatha Subramanian	sweathasubramanian@my.unt.edu

#### Normalization:

Normalization is a technique we implement to remove or reduce redundancy from a table and avoid anomalies. Three types of anomalies arise when our database is not normalized -

1. Insertion Anomaly: Not being able to add data to the database without adding the unrelated data.
2. Update Anomaly: When the records are not updated properly and the database becomes inconsistent, it leads to update anomaly.
3. Deletion Anomaly: An unintended loss of data in the database when we wish to delete certain records.

All these anomalies mainly happen due to functional dependency. FD defines the relationship between attributes in a table. An attribute A is functionally dependent on another attribute B (in the same table) if the value of B is determined by A.

To reduce FDs in our database, we normalize the tables and reduce them into third normal form (3NF) or Boyce-Codd normal form (BCNF). Reducing FDs will help in making the database more efficient and functional.

#### Initial Database Schema:

Initially, the tables and attributes of our database is as follows:

1. **HotelChain**(ID, name, PhoneNumber, website)
2. **Guest**(ID, name, PhoneNumber, roomNumber, email, IDProof, CardNumber, Address)
3. **Hotel**(ID, name, PhoneNumber, email, HotelType, City, State, Country, AddressLine1, AddressLine2, ZipCode, Description, Capacity, Rating, HotelChain\_ID)
4. **Employee**(ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID, SupervisorID)
5. **Attendance**(Emp\_ID, Attend\_Date, Hours)

6. **Amenities**(ID, Hotel\_ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
7. **Rooms**(Hotel\_ID, RoomNumber, FloorNumber)
8. **StarRating**(ID, Hotel\_ID, Rating, Guest\_ID, Feedback)
9. **EmployeeRating**(Emp\_ID, Guest\_ID, Rating)
10. **Discount**(Hotel\_ID, Guest\_ID, RewardPoints, DiscountRate)
11. **RoomType**(RoomType, RoomNumber, Cost, Description)
12. **HotelService**(ID, Name, Hotel\_ID, Cost)
13. **Packages**(ID, Name, Hotel, Cost, Duration, Valid\_Till, Description)
14. **Maintenance**(ID, Maintanence\_Type, Hotel\_ID, Emp\_ID, Room\_Number)
15. **Promotions**(ID, Name, Hotel\_ID, Details, Start\_Date, End\_Date)
16. **Bookings**(ID, RoomsBooked, Booking\_Date, Checkin, Checkout, Hotel\_ID, Emp\_ID, Room\_Number, Guest\_ID)
17. **Payments**(ID, Guest\_ID, Booking\_ID, Confirmation\_ID)

## Identifying the Functional Dependencies:

### HotelChain

#### Functional Dependencies:

ID → name, PhoneNumber, website

#### GuestFunctional Dependencies:

ID → name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address

### Hotel

#### Functional Dependencies:

ID → name, PhoneNumber, email, HotelType, City, State, Country, AddressLine1, AddressLine2, ZipCode, Description, Capacity, Rating, HotelChain\_ID

ZipCode → City, State, Country, AddressLine1, AddressLine2

### Employee

#### Functional Dependencies:

ID → name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID, SupervisorID

SupervisorID → (If SupervisorID uniquely identifies another employee, it could potentially determine all attributes of that employee. This is often the case but would require additional context to confirm.)

### Attendance

#### Functional Dependencies:

(Emp\_ID, Attend\_Date) → Hours

### Amenities

#### Functional Dependencies:

Hotel\_ID  $\rightarrow$  ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare

### **Rooms**

#### **Functional Dependencies:**

RoomNumber  $\rightarrow$  Hotel\_ID, FloorNumber

### **StarRating**

#### **Functional Dependencies:**

(Hotel\_ID, ID)  $\rightarrow$  Rating, Guest\_ID, Feedback

### **EmployeeRating**

#### **Functional Dependencies:**

(Emp\_ID, Guest\_ID)  $\rightarrow$  Rating

### **Discount**

#### **Functional Dependencies:**

(Hotel\_ID, Guest\_ID)  $\rightarrow$  RewardPoints, DiscountRate

### **RoomType**

#### **Functional Dependencies:**

(RoomType, RoomNumber)  $\rightarrow$  Cost, Description

### **HotelService**

#### **Functional Dependencies:**

ID  $\rightarrow$  Name, Hotel\_ID, Cost

### **Packages**

#### **Functional Dependencies:**

ID  $\rightarrow$  Name, Hotel\_ID, Cost, Duration, Valid\_Till, Description

### **Maintenance**

#### **Functional Dependencies:**

ID  $\rightarrow$  Maintenance\_Type, Hotel\_ID, Emp\_ID, Room\_Number

### **Promotions**

#### **Functional Dependencies:**

ID  $\rightarrow$  Name, Hotel\_ID, Details, Start\_Date, End\_Date

### **Bookings**

#### **Functional Dependencies:**

ID  $\rightarrow$  RoomsBooked, Booking\_Date, Checkin, Checkout, Hotel\_ID, Emp\_ID, Room\_Number, Guest\_ID

## Payments

### Functional Dependencies:

ID  $\rightarrow$  Guest\_ID, Booking\_ID, Confirmation\_ID

## Decomposition

From the descriptions of the tables and their functional dependencies, we can identify the tables that may not be in BCNF due to violations where a non-superkey determines other attributes. Here's a list of such tables and their potential violations:

### Hotel

Potential Violation: **ZipCode**  $\rightarrow$  City, State, Country, AddressLine1, AddressLine2. If **ZipCode** is not a candidate key, this represents a violation because it is a non-superkey determining other attributes.

### Employee

Potential Violation: **SupervisorID** potentially determining supervisor attributes (if SupervisorID uniquely identifies another employee, it could potentially determine all attributes of that employee). This would be a transitive dependency, which violates BCNF.

These tables have dependencies that involve non-candidate keys determining other attributes, which is a direct violation of BCNF conditions. To resolve these issues, further decomposition of these tables might be necessary, as outlined in the normalization process described earlier.

## Hotel Table Decomposition

### Original Schema:

Hotel(ID, name, PhoneNumber, email, HotelType, City, State, Country, AddressLine1, AddressLine2, ZipCode, Description, Capacity, Rating, HotelChain\_ID).

### Potential BCNF Violation:

ZipCode  $\rightarrow$  City, State, Country, AddressLine1, AddressLine2.

### Decomposition to Resolve BCNF Violation:

#### Hotel:

1. Attributes: ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID.
2. Primary Key: ID

3. Dependencies:  $ID \rightarrow \text{name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID}$ .
4. This table is in BCNF as all attributes are functionally dependent on the primary key, which is a superkey.

#### **HotelAddress:**

1. Attributes: ZipCode, City, State, Country, AddressLine1, AddressLine2.
2. Primary Key: ZipCode
3. Dependencies:  $\text{ZipCode} \rightarrow \text{City, State, Country, AddressLine1, AddressLine2}$ .
4. This table is in BCNF because ZipCode, the primary key, determines all other attributes in the table.

### **Employee Table Decomposition**

#### **Original Schema:**

Employee(ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID, SupervisorID).

#### **Potential BCNF Violation:**

$\text{SupervisorID} \rightarrow \text{Supervisor attributes (if SupervisorID determines attributes of the supervisor)}$ .

### **Decomposition to Resolve BCNF Violation:**

#### **Employee:**

1. Attributes: ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID.
2. Primary Key: ID
3. Dependencies:  $ID \rightarrow \text{name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID}$ .
4. This table is in BCNF as ID is a superkey.

#### **Supervision:**

1. Attributes: EmployeeID, SupervisorID.
2. Primary Key: EmployeeID
3. Dependencies:  $\text{EmployeeID} \rightarrow \text{SupervisorID}$ .
4. This table is in BCNF because the primary key determines all attributes in the table.

These decompositions remove the violations by isolating the dependencies where a non-superkey determines other attributes. Each table in the new schema satisfies the conditions of BCNF, ensuring that for every non-trivial functional dependency, the left side is a superkey.

## **Updated Tables (hotel and hotel address)**

### **Updates Hotel table.**

```
CREATE TABLE Hotel (  
    ID INT PRIMARY KEY CHECK (ID >= 1000 AND ID <= 9999),  
    name VARCHAR2(35),  
    PhoneNumber INT UNIQUE CHECK (PhoneNumber >= 1000000000 AND  
    PhoneNumber <= 9999999999),  
    email VARCHAR(255) CHECK (email LIKE '%_@_%._%'),  
    HotelType VARCHAR(25) CHECK (HotelType IN ('Resort', 'Bed & Breakfast', 'Hostel',  
    'Hotel')),  
    ZipCode INT NOT NULL, -- Foreign key relation to be defined if necessary  
    Description VARCHAR(500),  
    Capacity INT,  
    Rating CHAR(1) CHECK (Rating IN ('A', 'B', 'C', 'D', 'E')),  
    HotelChain_ID INT,  
    FOREIGN KEY (HotelChain_ID) REFERENCES HotelChain(ID)  
);
```

### **Updated Hotel Address Table**

```
CREATE TABLE HotelAddress (  
    ZipCode INT PRIMARY KEY CHECK (ZipCode >= 10000 AND ZipCode <= 99999),  
    City VARCHAR(25) NOT NULL,  
    State VARCHAR(25) NOT NULL,  
    Country VARCHAR(25) NOT NULL,  
    AddressLine1 VARCHAR(50) NOT NULL,  
    AddressLine2 VARCHAR(50)  
);
```

## **Updated Tables (employee and supervision)**

### **Updated Employee Table**

```
CREATE TABLE Employee (  
    ID INT PRIMARY KEY CHECK (ID >= 1000 AND ID <= 9999),  
    name VARCHAR2(35),  
    PhoneNumber INT UNIQUE CHECK (PhoneNumber >= 1000000000 AND  
    PhoneNumber <= 9999999999),  
    email VARCHAR(255) CHECK (email LIKE '%_@_%._%'),  
    Address VARCHAR(250),  
    Designation VARCHAR(25),  
    CriminalRecord VARCHAR(50),  
    Education VARCHAR(50),  
    Certifications VARCHAR(50),  
    Department VARCHAR(25) NOT NULL,  
    Hotel_ID INT,  
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)  
);
```

### **Updated Supervisor Table**

```
CREATE TABLE Supervision (  
    EmployeeID INT,  
    SupervisorID INT,  
    PRIMARY KEY (EmployeeID),  
    FOREIGN KEY (EmployeeID) REFERENCES Employee(ID),  
    FOREIGN KEY (SupervisorID) REFERENCES Employee(ID)  
);
```

## **Insertions for Each Table:**

### **Hotel Table Inserts:**

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (4567, 'Grand Resort', 9876543210, 'grandresort@quehotelchain.com', 'Resort', 33139, 'Luxurious beachfront resort', 200, 'A', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (5678, 'Cozy Inn', 7543209876, 'cozyinn@quehotelchain.com', 'Bed & Breakfast', 81611, 'Quaint bed and breakfast in the mountains', 15, 'B', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (6789, 'Sunset Lodge', 5412369870, 'sunsetlodge@quehotelchain.com', 'Hotel', 93101, 'Charming seaside hotel with ocean views', 100, 'C', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (7890, 'Mountain View Retreat', 7896541230, 'mountainview@quehotelchain.com', 'Resort', 89452, 'Serenity in the heart of the mountains', 50, 'D', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (8901, 'City Lights Hotel', 1234876509, 'citylights@quehotelchain.com', 'Hotel', 10001, 'Stylish urban hotel in the heart of the city', 150, 'E', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (9012, 'Sunny Beach Resort', 4567890123, 'sunnybeach@quehotelchain.com', 'Resort', 92109, 'Fun and relaxation by the beach', 300, 'A', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (1234, 'Mountain Peak Lodge', 9870123456, 'mountainpeak@quehotelchain.com', 'Hotel', 80202, 'Escape to the Rockies', 80, 'B', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (2345, 'Riverside Retreat', 3698745210, 'riverside@quehotelchain.com', 'Resort', 97204, 'Tranquil riverside escape', 120, 'C', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (3456, 'Harbor View Hotel', 6325489701, 'harborview@quehotelchain.com', 'Hotel', 98101, 'Scenic views of the harbor', 90, 'D', 2024);

INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID) VALUES (6790, 'The Vineyard Inn', 9854712360, 'vineyardinn@quehotelchain.com', 'Bed & Breakfast', 94558, 'Relax in the heart of wine country', 25, 'E', 2024);

**HotelAddress Table Inserts:**



```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (33139, 'Miami', 'Florida', 'USA', '123 Ocean Avenue', 'Suite 1001');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (81611, 'Aspen', 'Colorado', 'USA', '456 Mountain Road', 'Apt 202');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (93101, 'Santa Barbara', 'California', 'USA', '789 Seaside Boulevard', 'Unit 500');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (89452, 'Lake Tahoe', 'Nevada', 'USA', '321 Pinecrest Drive', 'Cabin 20');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (10001, 'New York', 'New York', 'USA', '987 Broadway Street', 'Floor 30');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (92109, 'San Diego', 'California', 'USA', '654 Sandy Lane', 'Suite 300');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (80202, 'Denver', 'Colorado', 'USA', '321 Summit Avenue', 'Chalet 10');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (97204, 'Portland', 'Oregon', 'USA', '456 Riverwalk Drive', 'Suite 201');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (98101, 'Seattle', 'Washington', 'USA', '789 Marina Drive', 'Pier 15');
```

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (94558, 'Napa Valley', 'California', 'USA', '101 Grapevine Lane', 'Vineyard
Cottage');
```

### **Employee Table Inserts:**

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,
CriminalRecord, Education, Certifications, Department, Hotel_ID) VALUES (1001, 'John
Smith', 1234567890, 'john.smith@example.com', '123 Main St Miami FL USA', 'Manager',
'None', 'Bachelor of Business Administration', 'Management Certification', 'Management',
4567);
```

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,
CriminalRecord, Education, Certifications, Department, Hotel_ID) VALUES (1002, 'Alice
Johnson', 2345678901, 'alice.johnson@example.com', '456 Mountain Rd Aspen CO USA',
'Front Desk Clerk', 'None', 'High School Diploma', 'Hospitality Certification', 'Front Desk',
5678);
```

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,
CriminalRecord, Education, Certifications, Department, Hotel_ID) VALUES (1003, 'Michael
Williams', 3456789012, 'michael.williams@example.com', '789 Seaside Blvd Santa Barbara
CA USA', 'Chef', 'None', 'Culinary Arts Degree', 'Food Safety Certification', 'Kitchen', 6789);
```

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,
CriminalRecord, Education, Certifications, Department, Hotel_ID) VALUES (1004, 'Emily
```

Brown', 4567890123, '[emily.brown@example.com](mailto:emily.brown@example.com)', '321 Pinecrest Dr Lake Tahoe NV USA', 'Housekeeping Supervisor', 'None', 'High School Diploma', 'Hospitality Certification', 'Housekeeping', 7890);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1005, 'David Miller', 5678901234, '[david.miller@example.com](mailto:david.miller@example.com)', '987 Broadway St New York NY USA', 'Concierge', 'None', 'Bachelor of Arts in Hospitality Management', 'Concierge Certification', 'Concierge', 8901);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1006, 'Jessica Davis', 6789012345, '[jessica.davis@example.com](mailto:jessica.davis@example.com)', '654 Sandy Ln San Diego CA USA', 'Spa Therapist', 'None', 'Certified Massage Therapist', 'Spa Certification', 'Spa', 9012);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1007, 'Christopher Wilson', 7890123456, '[christopher.wilson@example.com](mailto:christopher.wilson@example.com)', '321 Summit Ave Denver CO USA', 'Maintenance Technician', 'None', 'High School Diploma', 'Maintenance Certification', 'Maintenance', 1234);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1008, 'Sarah Martinez', 8901234567, '[sarah.martinez@example.com](mailto:sarah.martinez@example.com)', '456 Riverwalk Dr Portland OR USA', 'Event Coordinator', 'None', 'Bachelor of Arts in Event Management', 'Event Planning Certification', 'Events', 2345);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1009, 'Daniel Anderson', 9012345678, '[daniel.anderson@example.com](mailto:daniel.anderson@example.com)', '789 Marina Dr Seattle WA USA', 'Security Officer', 'None', 'High School Diploma', 'Security Certification', 'Security', 3456);

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID) VALUES (1010, 'Olivia Garcia', 1234876509, '[olivia.garcia@example.com](mailto:olivia.garcia@example.com)', '101 Grapevine Ln Napa Valley CA USA', 'Wine Sommelier', 'None', 'Certified Sommelier', 'Wine Tasting Certification', 'Food & Beverage', 6790);

### **Supervision Table Inserts:**

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1002, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1003, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1004, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1005, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1006, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1007, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1008, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1009, 1001);

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1010, 1001);

## Updated Project Description

### Project Description:

QUEHOTEL is a well-known international hotel organization which requires a well-defined and precise database to manage its resorts and lodging facilities throughout the world.

ROOMWISE (Resort Operations and Management Workflow Integration System), is a system which helps the management to store the details of all the resorts, employees, guests, amenities etc., all over the world. To ensure efficient database management, the system has been updated to meet normalization standards, specifically to Boyce-Codd Normal Form (BCNF), to enhance data integrity and reduce redundancy.

In order to meet these requirements, entities have been carefully constructed, with an emphasis on decomposing tables like Hotel and Employee to achieve BCNF and guarantee that each determinant is a superkey. To make this adjustment, employee supervision was managed using a separate Supervision database, and the address information from hotels was divided into a separate HotelAddress table.

### Updated Database Schema:

1. **HotelChain**(ID, name, PhoneNumber, website)
2. **Guest**(ID, name, PhoneNumber, roomNumber, email, IDProof, CardNumber, Address)
3. **Hotel**(ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain\_ID)
4. **HotelAddress**(ZipCode, City, State, Country, AddressLine1, AddressLine2)
5. **Employee**(ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel\_ID)
6. **Supervision**(EmployeeID, SupervisorID)
7. **Attendance**(Emp\_ID, Attend\_Date, Hours)
8. **Amenities**(ID, Hotel\_ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
9. **Rooms**(Hotel\_ID, RoomNumber, FloorNumber)
10. **StarRating**(ID, Hotel\_ID, Rating, Guest\_ID, Feedback)
11. **EmployeeRating**(Emp\_ID, Guest\_ID, Rating)
12. **Discount**(Hotel\_ID, Guest\_ID, RewardPoints, DiscountRate)
13. **RoomType**(RoomType, RoomNumber, Cost, Description)
14. **HotelService**(ID, Name, Hotel\_ID, Cost)
15. **Packages**(ID, Name, Hotel, Cost, Duration, Valid\_Till, Description)
16. **Maintenance**(ID, Maintenance\_Type, Hotel\_ID, Emp\_ID, Room\_Number)

17. **Promotions**(ID, Name, Hotel\_ID, Details, Start\_Date, End\_Date)
18. **Bookings**(ID, RoomsBooked, Booking\_Date, Checkin, Checkout, Hotel\_ID, Emp\_ID, Room\_Number, Guest\_ID)
19. **Payments**(ID, Guest\_ID, Booking\_ID, Confirmation\_ID)

## Updated Table Outputs:

The screenshot shows the SQL Developer interface with the 'Query Builder' tab selected. The SQL script in the main editor is as follows:

```
CREATE TABLE Hotel (
  ID INT PRIMARY KEY CHECK (ID >= 1000 AND ID <= 9999),
  name VARCHAR2(35),
  PhoneNumber INT UNIQUE CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999),
  email VARCHAR(255) CHECK (email LIKE '%_@%._%'),
  HotelType VARCHAR(25) CHECK (HotelType IN ('Resort', 'Bed and Breakfast', 'Hostel', 'Hotel')),
  ZipCode INT NOT NULL, -- Foreign key relation to be defined if necessary
  Description VARCHAR(500),
  Capacity INT,
  Rating CHAR(1) CHECK (Rating IN ('A', 'B', 'C', 'D', 'E')),
  HotelChain_ID INT,
  FOREIGN KEY (HotelChain_ID) REFERENCES HotelChain(ID)
);
```

The 'Script Output' pane at the bottom shows the message: 'Table HOTEL created.' and 'Task completed in 1.082 seconds'.

The screenshot shows the SQL Developer interface with the 'Query Builder' tab selected. The SQL script in the main editor is as follows:

```
CREATE TABLE HotelAddress (
  ZipCode INT PRIMARY KEY CHECK (ZipCode >= 10000 AND ZipCode <= 99999),
  City VARCHAR(25) NOT NULL,
  State VARCHAR(25) NOT NULL,
  Country VARCHAR(25) NOT NULL,
  AddressLine1 VARCHAR(50) NOT NULL,
  AddressLine2 VARCHAR(50)
);
```

The 'Script Output' pane at the bottom shows the message: 'Table HOTELADDRESS created.' and 'Task completed in 1.082 seconds'.

Welcome Page fdb\_project

Worksheet Query Builder

```
CREATE TABLE Employee (  
  ID INT PRIMARY KEY CHECK (ID >= 1000 AND ID <= 9999),  
  name VARCHAR(35),  
  PhoneNumber INT UNIQUE CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999),  
  email VARCHAR(255) CHECK (email LIKE '%_@_%._%'),  
  Address VARCHAR(250),  
  Designation VARCHAR(25),  
  CriminalRecord VARCHAR(50),  
  Education VARCHAR(50),  
  Certifications VARCHAR(50),  
  Department VARCHAR(25) NOT NULL,  
  Hotel_ID INT,  
  FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)  
);
```

Script Output x

Task completed in 1.082 seconds

Table EMPLOYEE created.

Welcome Page fdb\_project

Worksheet Query Builder

```
CREATE TABLE Supervision (  
  EmployeeID INT,  
  SupervisorID INT,  
  PRIMARY KEY (EmployeeID),  
  FOREIGN KEY (EmployeeID) REFERENCES Employee(ID),  
  FOREIGN KEY (SupervisorID) REFERENCES Employee(ID)  
);
```

Script Output x

Task completed in 1.082 seconds

Table SUPERVISION created.

Welcome Page fdb\_project HOTELADDRESS

Worksheet Query Builder

```
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID)  
VALUES (4567, 'Grand Resort', 9876543210, 'grandresort@quehotelchain.com', 'Resort', 33139, 'Luxurious beachfront resort', 200, 'A', 2024);  
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID)  
VALUES (5678, 'Cozy Inn', 7543209876, 'cozyinn@quehotelchain.com', 'Bed and Breakfast', 81611,  
'Quaint bed and breakfast in the mountains', 15, 'B', 2024);  
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID)  
VALUES (6789, 'Sunset Lodge', 5412369870, 'sunsetlodge@quehotelchain.com', 'Hotel', 93101,  
'Charming seaside hotel with ocean views', 100, 'C', 2024);  
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID)  
VALUES (7890, 'Mountain View Retreat', 7896541230, 'mountainview@quehotelchain.com', 'Resort', 89452,  
'Serenity in the heart of the mountains', 50, 'D', 2024);  
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID)  
VALUES (8901, 'City Lights Hotel', 1234876509, 'citylights@quehotelchain.com', 'Hotel', 10001, 'Stylish urban hotel in the heart of the cit  
INSERT INTO Hotel (ID, name, PhoneNumber, email, HotelType, ZipCode, Description, Capacity, Rating, HotelChain_ID) VALUES (9012, 'Sunny Bea
```

Script Output x

Task completed in 309.39 seconds

1 row inserted.

1 row inserted.

Worksheet Query Builder

```
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (33139, 'Miami', 'Florida', 'USA', '123 Ocean Avenue', 'Suite 1001');
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (81611, 'Aspen', 'Colorado', 'USA', '456 Mountain Road', 'Apt 202');
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (93101, 'Santa Barbara', 'California', 'USA', '789 Seaside Boulevard', 'Unit 500');
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (89452, 'Lake Tahoe', 'Nevada', 'USA', '321 Pinecrest Drive', 'Cabin 20');
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
VALUES (10001, 'New York', 'New York', 'USA', '987 Broadway Street', 'Floor 30');
INSERT INTO HotelAddress (ZipCode, City, State, Country, AddressLine1, AddressLine2)
```

Script Output x

Task completed in 309.39 seconds

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

Worksheet Query Builder

```
VALUES (6790, 20, 'YES', 'NO', 'NO', 'YES', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO');
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID)
VALUES (1001, 'John Smith', 1234567890, 'john.smith@example.com', '123 Main St Miami FL USA', 'Manager', 'None',
'Bachelor of Business Administration', 'Management Certification', 'Management', 4567);
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID)
VALUES (1002, 'Alice Johnson', 2345678901, 'alice.johnson@example.com', '456 Mountain Rd Aspen CO USA',
'Front Desk Clerk', 'None', 'High School Diploma', 'Hospitality Certification', 'Front Desk', 5678);
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID)
VALUES (1003, 'Michael Williams', 3456789012, 'michael.williams@example.com', '789 Seaside Blvd Santa Barbara CA USA',
'Chef', 'None', 'Culinary Arts Degree', 'Food Safety Certification', 'Kitchen', 6789);
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID)
VALUES (1004, 'Emily Brown', 4567890123, 'emily.brown@example.com', '321 Pinecrest Dr Lake Tahoe NV USA',
```

Script Output x

Task completed in 309.39 seconds

1 row inserted.

1 row inserted.

1 row inserted.

Welcome Page | fab\_project | HOTELADDRESS

1 of 1

Worksheet | Query Builder

```

INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1002, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1003, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1004, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1005, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1006, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1007, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1008, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1009, 1001);
INSERT INTO Supervision (EmployeeID, SupervisorID) VALUES (1010, 1001);

```

Script Output x

Task completed in 309.39 seconds

```

1 row inserted.

1 row inserted.

1 row inserted.

```

Welcome Page | fab\_project | HOTELADDRESS

1.43200004 seconds

1 of 1

Worksheet | Query Builder

```

desc hotel;

```

Script Output x

Task completed in 1.432 seconds

Name	Null?	Type
ID	NOT NULL	NUMBER (38)
NAME		VARCHAR2 (35)
PHONENUMBER		NUMBER (38)
EMAIL		VARCHAR2 (255)
HOTELTYPE		VARCHAR2 (25)
ZIPCODE	NOT NULL	NUMBER (38)
DESCRIPTION		VARCHAR2 (500)
CAPACITY		NUMBER (38)
RATING		CHAR (1)
HOTELCHAIN_ID		NUMBER (38)

Oracle SQL Developer interface showing the execution of a query to describe the HOTELADDRESS table.

**Top Bar:** Welcome Page, fdb\_project, HOTELADDRESS

**Query Builder:** Worksheet, Query Builder

**Query:** desc hoteladdress;

**Script Output:** Task completed in 0.365 seconds

Name	Null?	Type
ZIPCODE	NOT NULL	NUMBER(38)
CITY	NOT NULL	VARCHAR2(25)
STATE	NOT NULL	VARCHAR2(25)
COUNTRY	NOT NULL	VARCHAR2(25)
ADDRESSLINE1	NOT NULL	VARCHAR2(50)
ADDRESSLINE2		VARCHAR2(50)



Oracle SQL Developer interface showing the Query Builder tab with the query `desc employee;` entered. The Script Output window displays the table structure for the `employee` table.

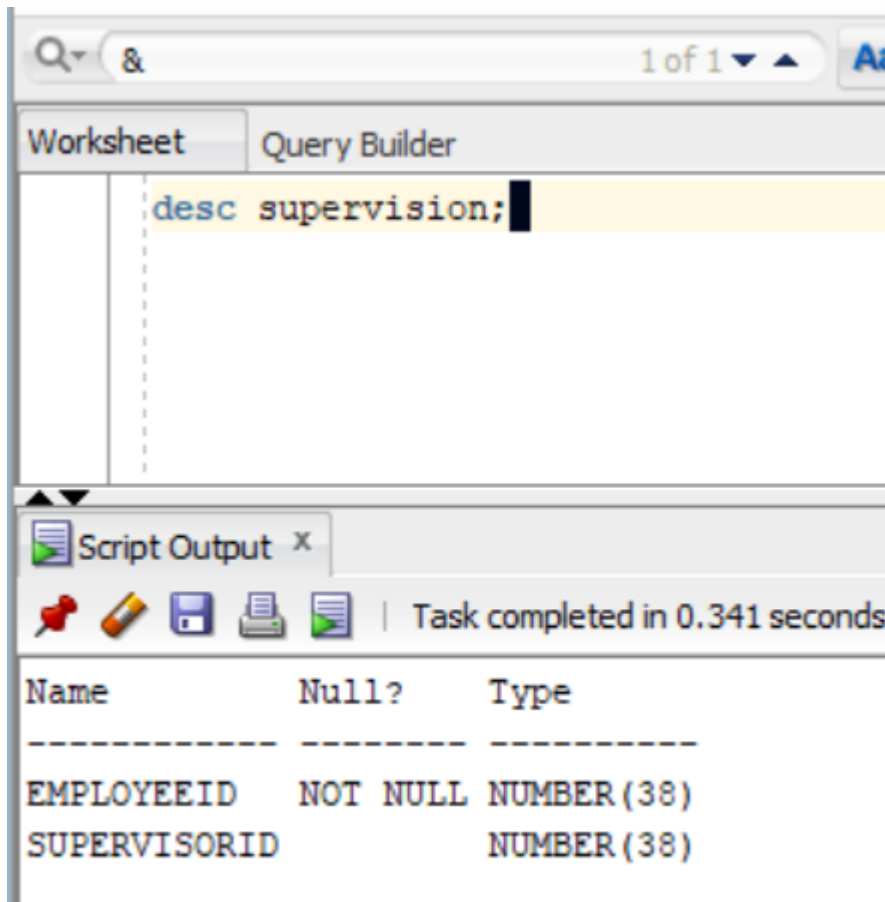
Query Builder

```
desc employee;
```

Script Output

Task completed in 0.342 seconds

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(35)
PHONENUMBER		NUMBER(38)
EMAIL		VARCHAR2(255)
ADDRESS		VARCHAR2(250)
DESIGNATION		VARCHAR2(25)
CRIMINALRECORD		VARCHAR2(50)
EDUCATION		VARCHAR2(50)
CERTIFICATIONS		VARCHAR2(50)
DEPARTMENT	NOT NULL	VARCHAR2(25)
HOTEL_ID		NUMBER(38)



### Individual Contribution:

For this week's project submission, I worked on identifying the functional dependencies. I helped in normalizing, creating and decomposing the tables and inserting the new tuples into the tables and made sure we got the correct outputs. I also assisted in writing the updated schema and updating the project description.

I also assisted with the documentation this week, formatting the document wherever necessary. I also organized timely and productive meetings to discuss the project and update the project wherever necessary, according to the mentioned guidelines.

