

CSCE – 5350 – Fundamentals of Database System
(Oracle DB System)
Group – 18

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 make a cohesive and efficient database management system, we create entities and entity-relationships in the first step. We may add more relationships in future depending on the requirements we assume.

Entities:

- ***hotel chain*** – QUEHOTEL is the HotelChain and is defined by -
 - **id, name, Phone number, website**
- ***hotel*** - Each hotel in the hotel chain is defined by the following attributes -
 - **id, Hotel type, name, Phone number, email, website, address_line_1, address_line_2, city, state, country, zipcode, description, capacity, rating, hotelchain_id**
- ***employee*** – Each employee working in the hotel is defined by the following attributes -
 - **id, name, designation, supervisorid, address, Phone number, email, department, education, certification, criminal_record, hotel_id, age**
- ***guest*** - Each guest in the hotel is defined by the following attributes -
 - **id, name, Phone number, email, idproof, address, card number, room_number**
- ***room*** – The rooms in the hotel are defined by:
 - **number, floornumber, hotel_id**
- ***Room type*** – each room will have a specific room type -
 - **roomtype, cost, description, no_of_beds, room_number**
- ***bookings*** – Bookings in the hotel are defined by the following attributes -
 - **id, bookingdate, check_in, check_out, rooms_booked, amount, room_number, hotel_id, guest_id, employee_id.**
- ***discount*** – Guests might sometimes be up for discount offers and they are defined by the following attributes -
 - **Discount rate, reward points, guest id, hotel id**
- ***Star ratings*** – The guests will have to give ratings for the services they received.
 - **id, rating, feedback, guest_id, hotel_id**
- ***amenities*** – All the amenities and facilities provided at the hotels are defined by -

- **id, pool, gym, steam_room, parking, child_day_care, medical_facilities, restaurants, auditorium, game_room, meeting_rooms, playarea, hotel_id**
- ***maintenance*** – The maintenance provided by hotel is defined by -
 - **id, type, hotel_id, employee_id, room_number**
- ***hotel_service*** – The additional services provided by the hotel to the guest is defined as-
 - **id, cost, name, hotel_id**,
- ***Employee rating*** – Each guest will give a rating to the employee who served them. This is defined by -
 - **guest id, emp id, rating**
- ***package*** – The packages chosen with the room are defined by -
 - **id, name, description, duration, cost, validity, hotel_id**
- ***promotions*** – promotions will have the following attributes.
 - **id, name, details, start_date, end_date, hotel_id**
- ***payments*** – The payment methods of guests are defined by -
 - **id, conformation_id, booking_id, guest_id**
- ***Attendance*** – The no.of hours worked by an employee is defined by -
 - **Emp id, date, hours**

Relationships:

Unary Relationships:

- Each supervisor **supervises** many employees. And supervisor is an employee.

Binary Relationships:

- ***One-to-One:***
 - one booking has at least one room.
- ***One-to-Many:***
 - One hotel chain has multiple hotels.
 - One hotel provides multiple amenities.
 - One hotel has many employees.
 - One hotel has rooms.
 - One hotel provides many room types.
 - One employee has many ratings.
 - One employee handles many bookings.
 - One employee has many maintenance duties.
 - One guest does multiple bookings.
 - One room can ask for multiple maintenance duties performed on it.
 - One Employee can work for multiple hours.
- ***Many-to-Many***
 - Many guests give rating to many employees.
 - Many employees provide many hotel services.
 - Many guests give rating to many employees.

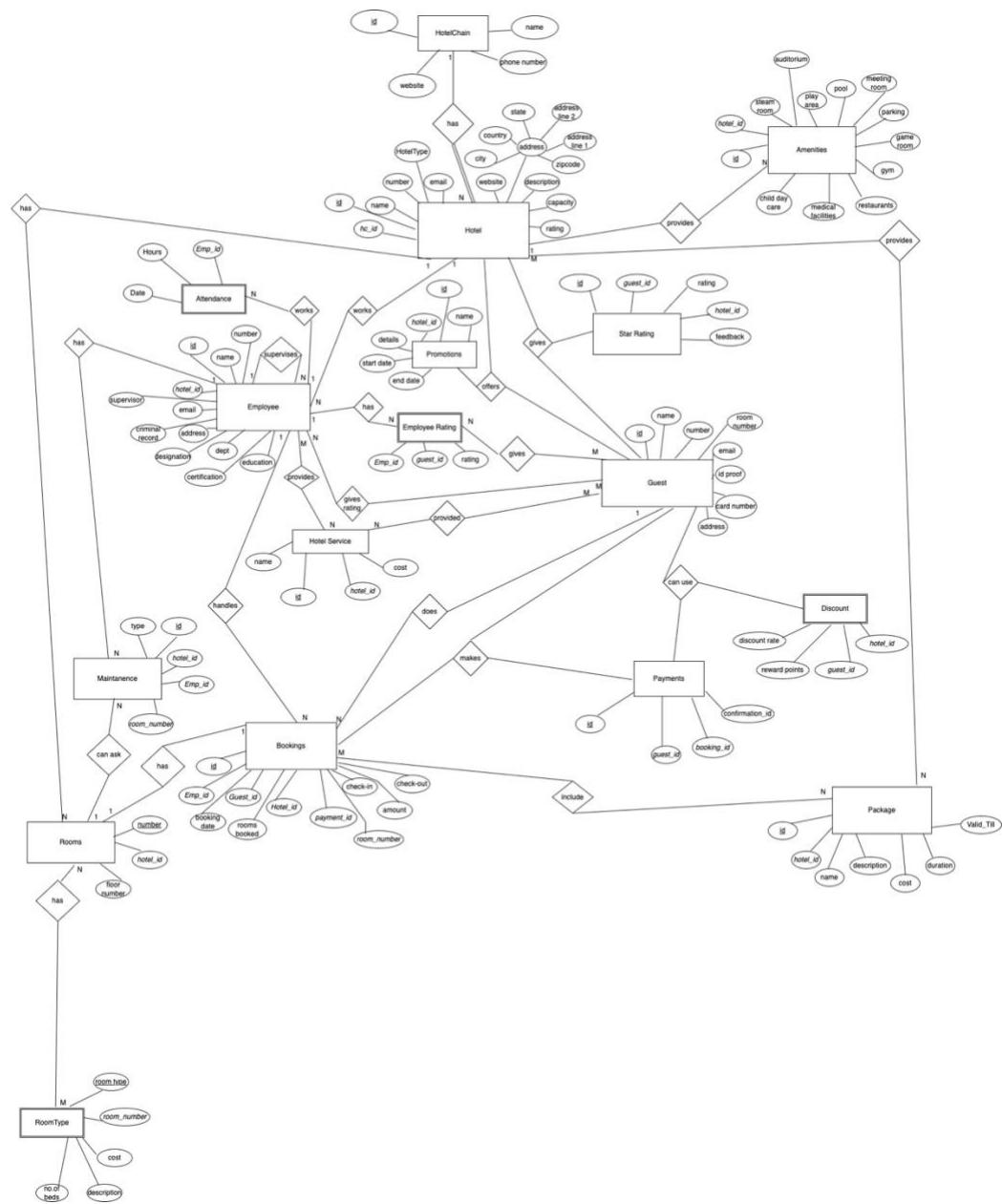
- Multiple services are provided to many guests.
- Hotel provides multiple packages.
- Many bookings have multiple packages.
- Multiple rooms have multiple roomtype.

Ternary Relationships:

- One guest gives rating to hotel.
- Guest books by making a payment.
- Guest can use discount while payments.
- One hotel offers promotion to guests.

ER – Diagram:

<https://drive.google.com/file/d/1OHCEk3P5LJpniXUTi6EJIMSQW0CKi-gn/view?usp=sharing>



Transformation of E-R diagram into relations:

Hotel Chain (**id**, name, Phone Number, website)

Hotel (**id**, Hotel Type, name, Phone Number, email, website, address_line_1, address_line_2, city, state, country, zipcode, description, capacity, rating, **Hotelchain_id**)

Employee (**id**, name, designation, **supervisorid**, address, Phone Number, email, department, education, certification, criminalrecord, **Hotel_id**, age)

Guest (**id**, name, Phone Number, email, idproof, address, cardnumber, **Room_number**)

Rooms (**number**, floornumber, **Hotel_id**)

Room type (**roomtype**, cost, description, no_of_beds, **Room_number**) – weak entity

Bookings (**id**, bookingdate, check_in, check_out, rooms_booked, amount, **Room_number**, **Hotel_id**, **Guest_id**, **Employee_id**)

Discount (discount rate, rewardpoints, **Guest_id**, **Hotel_id**)

Star rating (**id**, rating, feedback, **Guest_id**, **Hotel_id**)

Amenities (**id**, pool, gym, steam_room, parking, child_day_care, medical_facilities, restaurants, auditorium, game_room, meeting_rooms, playarea, **Hotel_id**)

Maintenance (**id**, type, **Hotel_id**, **Employee_id**, **Room_number**)

Hotel_service (**id**, cost, name, **Hotel_id**)

Emp rating (**Guest_id**, **Emp_id**, **Rating**)

Package (**id**, name, description, duration, cost, valid_till, **Hotel_id**)

Promotions (**id**, name, details, start_date, end_date, **Hotel_id**)

Payments (**id**, conformation_id, **Booking_id**, **Guest_id**)

Attendance (**date**, hours, **Emp_id**) – weak entity

Include(**pack id**, **booking id**)

Room_has_roomtype (**room number**, **room type**)

emp_provides_hotservice (**emp id**, **hotservice id**)

gives_rating (emp id, guest id)

Provided (guest id, hotelservice id)

hotel_provides_package (hotel id, pack id)

ASSUMPTIONS:

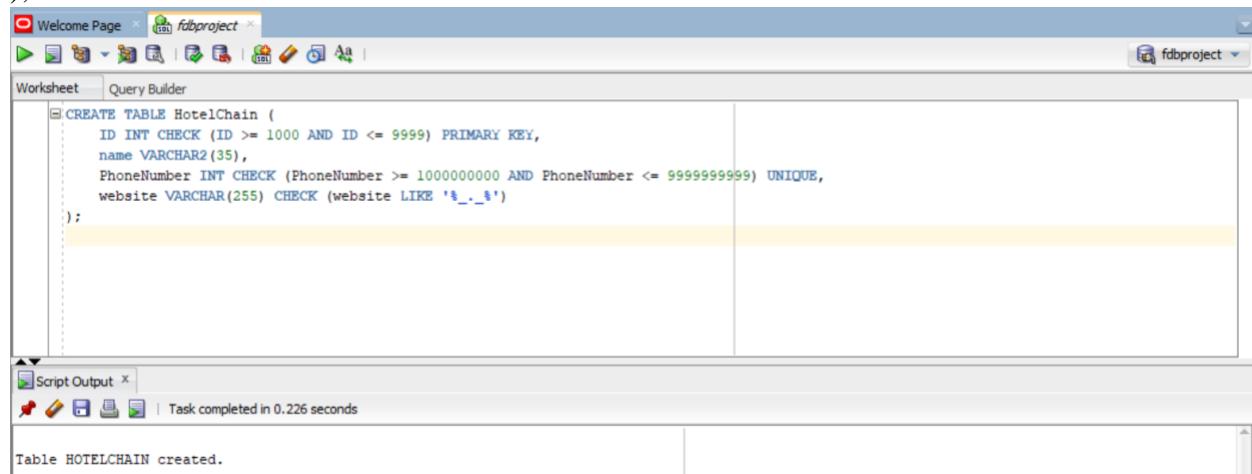
- We assume the hotel services provided by hotel to be - Room service, Laundry Service, Spa Massage, Airport Shuttle, Concierge Service, Fitness centre, car rental, guided tour and Valet parking and In-room dining.
- All the hotels in our database come under Quehotel, which is the only HotelChain in the database. We don't have to add anymore records for HotelChain table.
- Employees are employed in the hotel if they don't have any criminal record.
- Emp_Id, Guest_id, Hotel_id should be unique.
- Emails of each hotel in the hotel chain must be unique.
- Rooms will come under 3 room types - Single Bed, Double Bed and Deluxe.
- Idproof of guest can have alphanumeric values.

CREATION OF TABLES:

- **HotelChain table:**

```
CREATE TABLE HotelChain (
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name VARCHAR2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <=
    9999999999) UNIQUE,
    website VARCHAR(255) CHECK (website LIKE '%_.%')
```

);



The screenshot shows the Oracle SQL Developer interface. The top menu bar has 'Welcome Page' and 'fdbproject'. The main window is titled 'Worksheet' and contains a 'Query Builder' tab. The code in the worksheet pane is:

```
CREATE TABLE HotelChain (
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name VARCHAR2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999) UNIQUE,
    website VARCHAR(255) CHECK (website LIKE '%_.%')
);
```

The bottom pane, 'Script Output', shows the message: 'Table HOTELCHAIN created.' and 'Task completed in 0.226 seconds'.

➤ Guest table:

CREATE TABLE Guest(

```
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name varchar2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <=
9999999999),
    roomNumber INT CHECK (roomNumber >= 1000 AND roomNumber <= 9999),
    email Varchar2(255) CHECK (email LIKE '%_@_%._%'),
    IdProof VARCHAR2(30) CHECK (REGEXP_LIKE(IdProof, '^[a-zA-Z0-9]*$')),
    CardNumber CHAR(16),
    CONSTRAINT check_card_number CHECK (LENGTH(CardNumber) = 16 AND
REGEXP_LIKE(CardNumber, '^[0-9]{16}$')),
    Address Varchar2(255)
);
```

```

CREATE TABLE Guest(
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name varchar2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999),
    roomNumber INT CHECK (roomNumber >= 1000 AND roomNumber <= 9999),
    email Varchar2(255) CHECK (email LIKE '%_@_%._%'),
    IdProof VARCHAR2(30) CHECK (REGEXP_LIKE(IdProof, '^[a-zA-Z0-9]*$')),
    CardNumber CHAR(16),
    CONSTRAINT check_card_number CHECK (LENGTH(CardNumber) = 16 AND REGEXP_LIKE(CardNumber, '^[0-9]{16}$'))
);

```

Script Output | Task completed in 0.075 seconds

Table HOTELCHAIN created.

Table GUEST created.

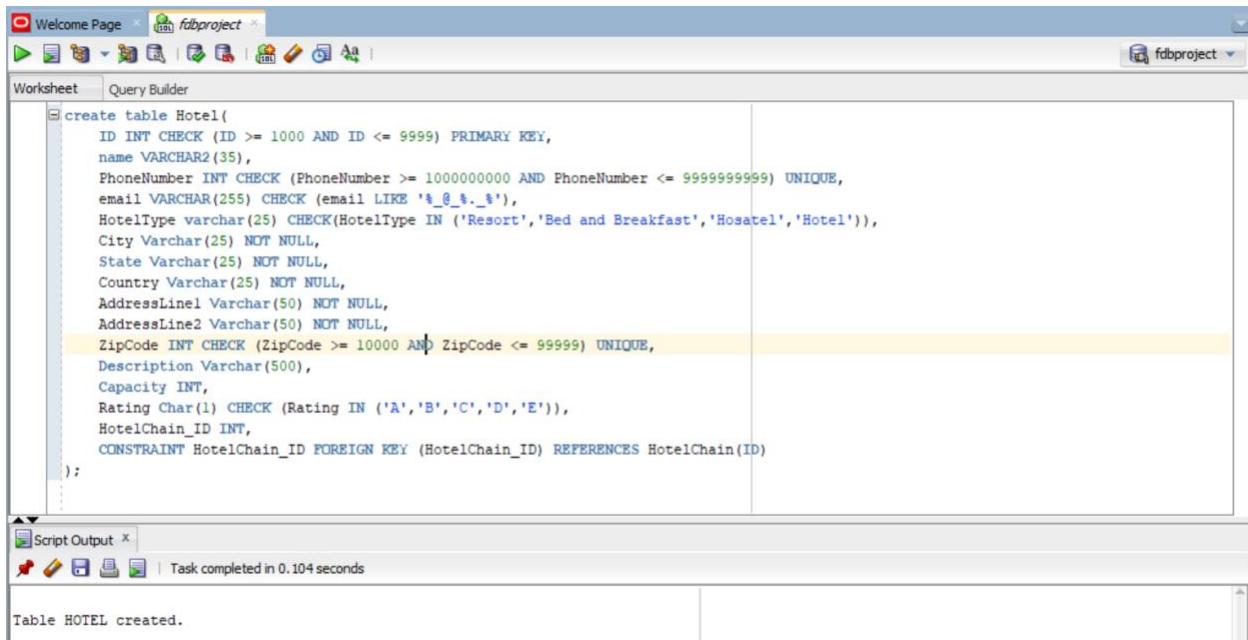
➤ Hotel table:

create table Hotel(

```

        ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
        name VARCHAR2(35),
        PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <=
        9999999999) UNIQUE,
        email VARCHAR(255) CHECK (email LIKE '%_@_%._%'),
        HotelType varchar(25) CHECK(HotelType IN ('Resort','Bed and Breakfast','Hosatel','Hotel')),
        City Varchar(25) NOT NULL,
        State Varchar(25) NOT NULL,
        Country Varchar(25) NOT NULL,
        AddressLine1 Varchar(50) NOT NULL,
        AddressLine2 Varchar(50) NOT NULL,
        ZipCode INT CHECK (ZipCode >= 10000 AND ZipCode <= 99999) UNIQUE,
        Description Varchar(500),
        Capacity INT,
        Rating Char(1) CHECK (Rating IN ('A','B','C','D','E')),
        HotelChain_ID INT,
        CONSTRAINT HotelChain_ID FOREIGN KEY (HotelChain_ID) REFERENCES
        HotelChain(ID)
    
```

);



The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is selected, displaying the SQL code for creating the 'Hotel' table. The code includes constraints for ID, name, PhoneNumber, email, HotelType, City, State, Country, AddressLine1, AddressLine2, ZipCode, Description, Capacity, Rating, HotelChain_ID, and a foreign key constraint for HotelChain_ID. The 'Script Output' tab at the bottom shows the message 'Table HOTEL created.'.

```
create table Hotel(
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name VARCHAR2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999) UNIQUE,
    email VARCHAR(255) CHECK (email LIKE '%_@_%._%'),
    HotelType varchar(25) CHECK(HotelType IN ('Resort','Bed and Breakfast','Hosatel','Hotel')),
    City Varchar(25) NOT NULL,
    State Varchar(25) NOT NULL,
    Country Varchar(25) NOT NULL,
    AddressLine1 Varchar(50) NOT NULL,
    AddressLine2 Varchar(50) NOT NULL,
    ZipCode INT CHECK (ZipCode >= 10000 AND ZipCode <= 99999) UNIQUE,
    Description Varchar(500),
    Capacity INT,
    Rating Char(1) CHECK (Rating IN ('A','B','C','D','E')),
    HotelChain_ID INT,
    CONSTRAINT HotelChain_ID FOREIGN KEY (HotelChain_ID) REFERENCES HotelChain(ID)
);
```

Table HOTEL created.

➤ Employee Table:

Create table Employee(

 ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,

 name VARCHAR2(35),

 PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999) UNIQUE,

 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,

 CONSTRAINT Hotel_ID FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID),

 SupervisorID INT,

 CONSTRAINT SupervisorID FOREIGN KEY (SupervisorID) REFERENCES Employee(ID)

);

The screenshot shows a database management application window. The top bar includes tabs for 'Welcome Page' and 'fabproject'. The main area is a 'Worksheet' tab showing a SQL script for creating the 'Employee' table. The script includes constraints for ID (primary key, check range 1000-9999), name (VARCHAR(35)), phone number (INT CHECK, range 1000000000-9999999999, unique), email (VARCHAR(255) CHECK, like '%_@%.%'), address (VARCHAR(250)), designation (VARCHAR(25)), criminal record (VARCHAR(50)), education (VARCHAR(50)), certifications (VARCHAR(50)), department (VARCHAR(25) NOT NULL), hotel ID (INT), supervisor ID (INT), and foreign keys for Hotel_ID and SupervisorID. The bottom window, 'Script Output', shows the message 'Table EMPLOYEE created.' and a completion time of 0.128 seconds.

```
Create table Employee(
    ID INT CHECK (ID >= 1000 AND ID <= 9999) PRIMARY KEY,
    name VARCHAR2(35),
    PhoneNumber INT CHECK (PhoneNumber >= 1000000000 AND PhoneNumber <= 9999999999) UNIQUE,
    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,
    CONSTRAINT Hotel_ID FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID),
    SupervisorID INT,
    CONSTRAINT SupervisorID FOREIGN KEY (SupervisorID) REFERENCES Employee(ID)
);
Table EMPLOYEE created.
```

➤ Attendance table:

Create table Attendance(

```
    Emp_ID INT,
    Attend_Date DATE,
    Hours INT CHECK (Hours>=0 AND Hours<=8),
    PRIMARY KEY (Emp_ID,Attend_Date),
    FOREIGN KEY (Emp_ID) REFERENCES Employee(ID)
);

```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL code for creating the ATTENDANCE table:

```

CREATE TABLE Attendance (
    Emp_ID INT,
    Attend_Date DATE,
    Hours INT CHECK (Hours >= 0 AND Hours <= 8),
    PRIMARY KEY (Emp_ID, Attend_Date),
    FOREIGN KEY (Emp_ID) REFERENCES Employee(ID)
);

```

In the bottom right corner of the worksheet, there is a message: "Table ATTENDANCE created." Below the worksheet, the 'Script Output' tab is visible, showing the message "Task completed in 0.077 seconds".

➤ Amenities table:

```

create table Amenities(
    Hotel_ID INT,
    ID INT PRIMARY KEY,
    SteamRoom Varchar(3) CHECK (SteamRoom IN ('YES','NO')),
    Auditorium Varchar(3) CHECK (Auditorium IN ('YES','NO')),
    PlayArea Varchar(3) CHECK (PlayArea IN ('YES','NO')),
    Pool Varchar(3) CHECK (Pool IN ('YES','NO')),
    MeetingRoom Varchar(3) CHECK (MeetingRoom IN ('YES','NO')),
    Parking Varchar(3) CHECK (Parking IN ('YES','NO')),
    GameRoom Varchar(3) CHECK (GameRoom IN ('YES','NO')),
    Gym Varchar(3) CHECK (Gym IN ('YES','NO')),
    Restaurants Varchar(3) CHECK (Restaurants IN ('YES','NO')),
    MedicalFacilities Varchar(3) CHECK (MedicalFacilities IN ('YES','NO')),
    ChildDayCare Varchar(3) CHECK (ChildDayCare IN ('YES','NO')),
    FOREIGN KEY (Hotel_ID) References Hotel(ID)
);

```

The screenshot shows the Oracle SQL Developer interface with the 'Worksheet' tab selected. The code in the worksheet pane is:

```

create table Amenities(
    Hotel_ID INT,
    ID INT PRIMARY KEY,
    SteamRoom Varchar(3) CHECK (SteamRoom IN ('YES','NO')),
    Auditorium Varchar(3) CHECK (Auditorium IN ('YES','NO')),
    PlayArea Varchar(3) CHECK (PlayArea IN ('YES','NO')),
    Pool Varchar(3) CHECK (Pool IN ('YES','NO')),
    MeetingRoom Varchar(3) CHECK (MeetingRoom IN ('YES','NO')),
    Parking Varchar(3) CHECK (Parking IN ('YES','NO')),
    GameRoom Varchar(3) CHECK (GameRoom IN ('YES','NO')),
    Gym Varchar(3) CHECK (Gym IN ('YES','NO')),
    Restaurants Varchar(3) CHECK (Restaurants IN ('YES','NO')),
    MedicalFacilities Varchar(3) CHECK (MedicalFacilities IN ('YES','NO')),
    ChildDayCare Varchar(3) CHECK (ChildDayCare IN ('YES','NO')),
    FOREIGN KEY (Hotel_ID) References Hotel(ID)
);

```

The 'Script Output' pane at the bottom shows the message: "Table AMENITIES created.".

➤ Rooms table:

```

create table Rooms(
    Hotel_ID INT,
    RoomNumber INT PRIMARY KEY Check (RoomNumber>=1000 AND
    RoomNumber<=9999),
    FloorNumber INT,
    FOREIGN KEY (Hotel_ID) References Hotel(ID)
);

```

The screenshot shows the Oracle SQL Developer interface with the 'Worksheet' tab selected. The code in the worksheet pane is:

```

create table Rooms(
    Hotel_ID INT,
    RoomNumber INT PRIMARY KEY Check (RoomNumber>=1000 AND RoomNumber<=9999),
    FloorNumber INT,
    FOREIGN KEY (Hotel_ID) References Hotel(ID)
);

```

The 'Script Output' pane at the bottom shows the message: "Table ROOMS created.".

➤ StarRating table:

```

create table StarRating(

```

```

Hotel_ID INT,
ID INT PRIMARY KEY,
Rating Char(1) Check (Rating IN ('A','B','C','D','E')),
Guest_ID INT,
Feedback Varchar(500),
FOREIGN KEY (Hotel_ID) References Hotel(ID),
FOREIGN KEY (Guest_ID) References Guest(ID)
);

```

The screenshot shows a database management tool window titled 'fdbproject'. The 'Worksheet' tab is active, displaying the SQL code for creating the 'StarRating' table. The code is identical to the one provided above. Below the worksheet, the 'Script Output' tab shows the result of the execution: 'Table STARRATING created.' A note indicates the task completed in 0.086 seconds.

```

create table StarRating(
    Hotel_ID INT,
    ID INT PRIMARY KEY,
    Rating Char(1) Check (Rating IN ('A','B','C','D','E')),
    Guest_ID INT,
    Feedback Varchar(500),
    FOREIGN KEY (Hotel_ID) References Hotel(ID),
    FOREIGN KEY (Guest_ID) References Guest(ID)
);

```

Script Output | Task completed in 0.086 seconds

Table STARRATING created.

➤ EmployeeRating table:

```

create table EmployeeRating(
    Emp_ID INT,
    Rating Char(1) Check (Rating IN ('A','B','C','D','E')),
    Guest_ID INT,
    PRIMARY KEY(Emp_ID,Guest_ID),
    FOREIGN KEY (Emp_ID) References Employee(ID),
    FOREIGN KEY (Guest_ID) References Guest(ID)
);

```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL code for creating the 'EmployeeRating' table:

```
create table EmployeeRating(
    Emp_ID INT,
    Rating Char(1) Check (Rating IN ('A','B','C','D','E')),
    Guest_ID INT,
    PRIMARY KEY(Emp_ID,Guest_ID),
    FOREIGN KEY (Emp_ID) References Employee(ID),
    FOREIGN KEY (Guest_ID) References Guest(ID)
);
```

The 'Script Output' tab at the bottom shows the result of the execution:

```
Table EMPLOYEERATING created.
```

➤ Discount table:

```
create table Discount(
    Hotel_ID INT,
    RewardPoints INT,
    DiscountRate INT,
    Guest_ID INT,
    PRIMARY KEY(Hotel_ID,Guest_ID),
    FOREIGN KEY (Hotel_ID) References Hotel(ID),
    FOREIGN KEY (Guest_ID) References Guest(ID)
);
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL code for creating the 'Discount' table:

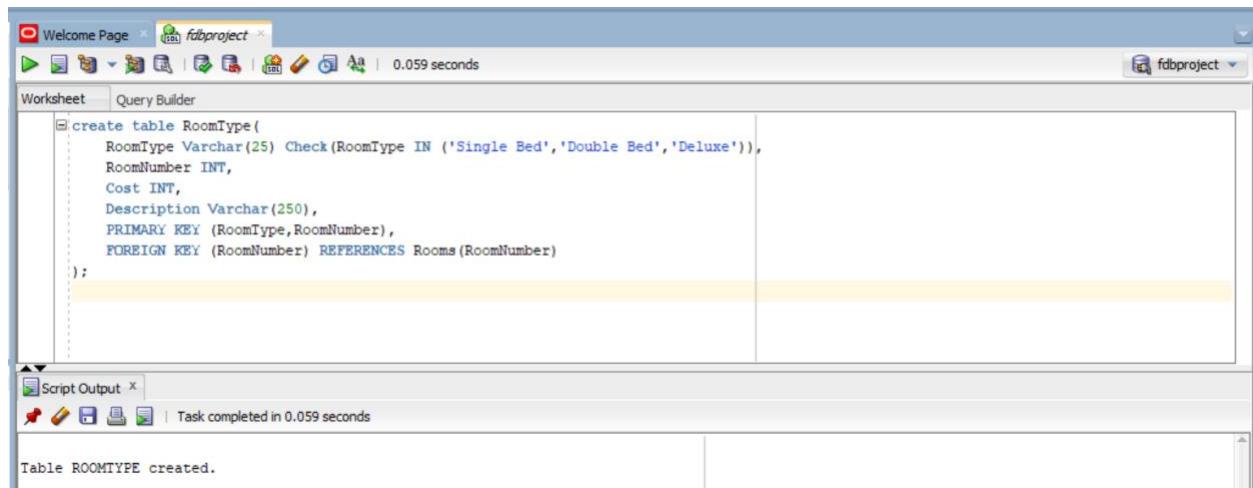
```
create table Discount(
    Hotel_ID INT,
    RewardPoints INT,
    DiscountRate INT,
    Guest_ID INT,
    PRIMARY KEY(Hotel_ID,Guest_ID),
    FOREIGN KEY (Hotel_ID) References Hotel(ID),
    FOREIGN KEY (Guest_ID) References Guest(ID)
);
```

The 'Script Output' tab at the bottom shows the result of the execution:

```
Table DISCOUNT created.
```

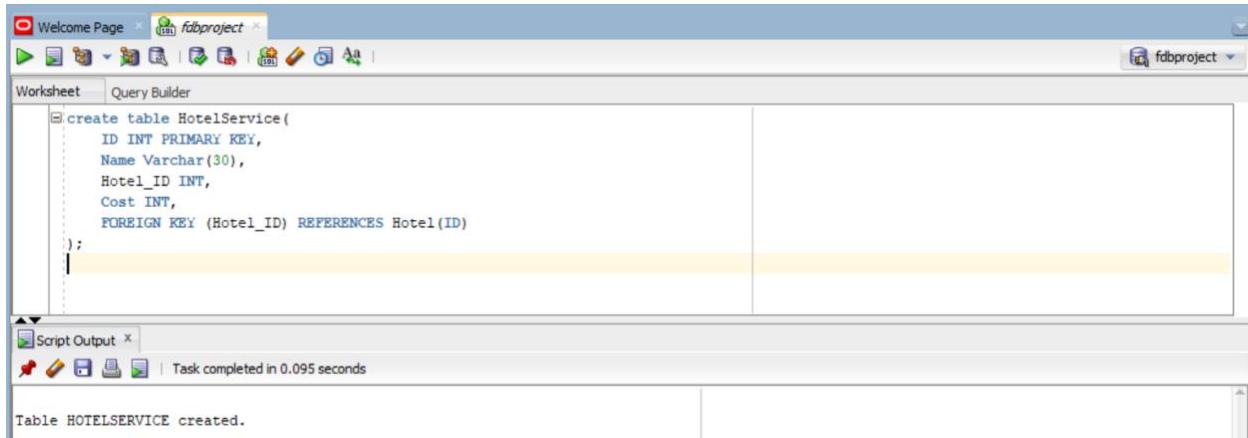
➤ **RoomType table:**

```
create table RoomType(
    RoomType Varchar(25) Check(RoomType IN ('Single Bed','Double Bed','Deluxe')),
    RoomNumber INT,
    Cost INT,
    Description Varchar(250),
    PRIMARY KEY (RoomType,RoomNumber),
    FOREIGN KEY (RoomNumber) REFERENCES Rooms(RoomNumber)
);
```



➤ **HotelService table:**

```
create table HotelService(
    ID INT PRIMARY KEY,
    Name Varchar(30),
    Hotel_ID INT,
    Cost INT,
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
);
```



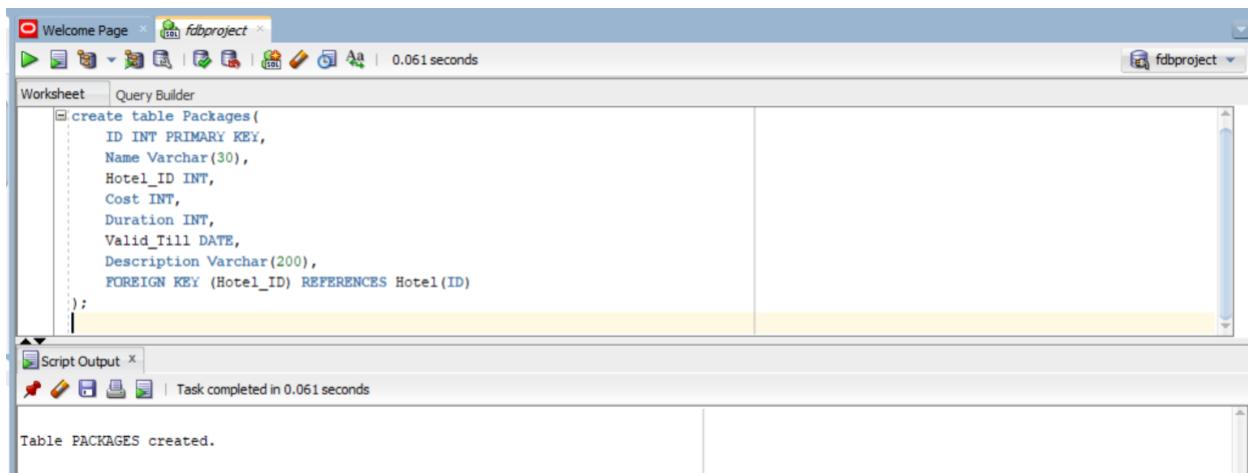
The screenshot shows a database management interface with a toolbar at the top. Below the toolbar, the title bar says "Welcome Page" and "fdbproject". The main area is titled "Worksheet" and contains a "Query Builder" tab. A SQL script is being typed into the worksheet:

```
create table HotelService(
    ID INT PRIMARY KEY,
    Name Varchar(30),
    Hotel_ID INT,
    Cost INT,
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
);
```

Below the worksheet is a "Script Output" window. It shows a green icon followed by the text "Task completed in 0.095 seconds". Underneath that, it says "Table HOTELSERVICE created."

➤ Packages table:

```
create table Packages(
    ID INT PRIMARY KEY,
    Name Varchar(30),
    Hotel_ID INT,
    Cost INT,
    Duration INT,
    Valid_Till DATE,
    Description Varchar(200),
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
);
```



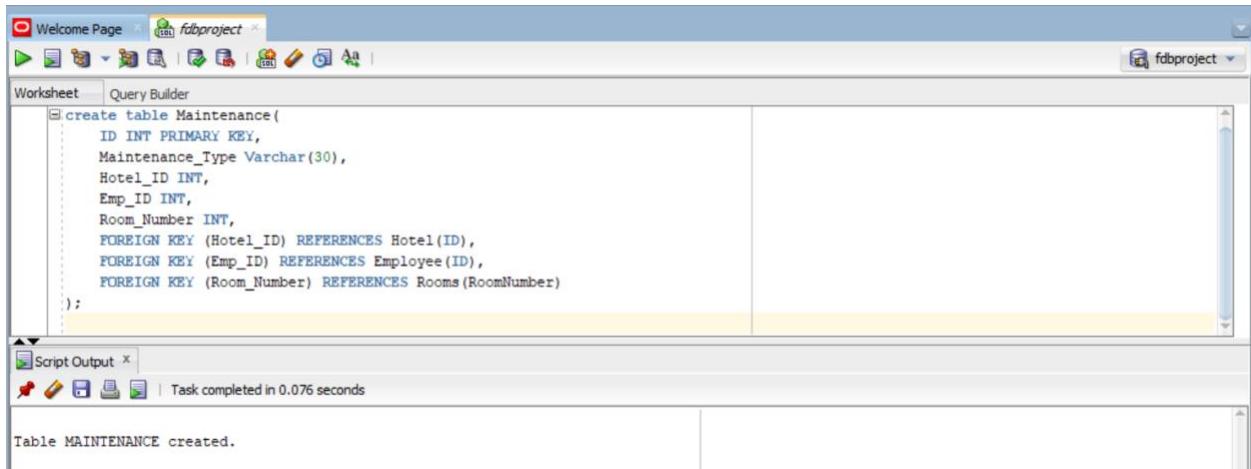
The screenshot shows a database management interface with a toolbar at the top. Below the toolbar, the title bar says "Welcome Page" and "fdbproject". The main area is titled "Worksheet" and contains a "Query Builder" tab. A SQL script is being typed into the worksheet:

```
create table Packages(
    ID INT PRIMARY KEY,
    Name Varchar(30),
    Hotel_ID INT,
    Cost INT,
    Duration INT,
    Valid_Till DATE,
    Description Varchar(200),
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
);
```

Below the worksheet is a "Script Output" window. It shows a green icon followed by the text "0.061seconds". Underneath that, it says "Table PACKAGES created."

➤ Maintenance table:

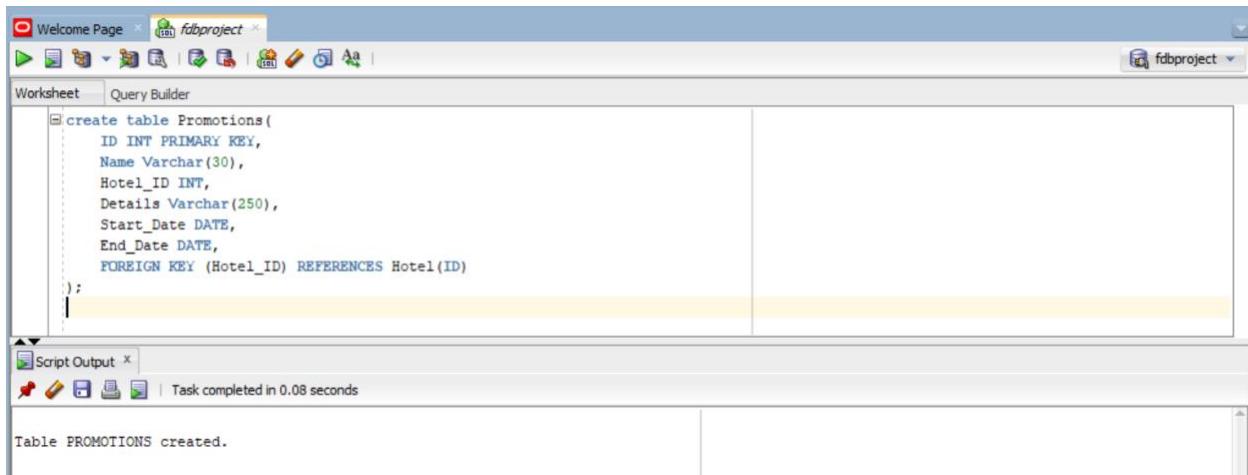
```
create table Maintenance(  
    ID INT PRIMARY KEY,  
    Maintenance_Type Varchar(30),  
    Hotel_ID INT,  
    Emp_ID INT,  
    Room_Number INT,  
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID),  
    FOREIGN KEY (Emp_ID) REFERENCES Employee(ID),  
    FOREIGN KEY (Room_Number) REFERENCES Rooms(RoomNumber)  
);
```



➤ Promotions table:

```
create table Promotions(  
    ID INT PRIMARY KEY,  
    Name Varchar(30),  
    Hotel_ID INT,  
    Details Varchar(250),  
    Start_Date DATE,  
    End_Date DATE,  
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
```

);



The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is selected, displaying the SQL code for creating the 'Promotions' table. The code includes columns for ID (Primary Key), Name (Varchar(30)), Hotel_ID (INT), Details (Varchar(250)), Start_Date (DATE), and End_Date (DATE), with a foreign key constraint linking Hotel_ID to the Hotel table. Below the worksheet, the 'Script Output' tab shows the message 'Table PROMOTIONS created.' and a note indicating the task completed in 0.08 seconds.

```
create table Promotions(
    ID INT PRIMARY KEY,
    Name Varchar(30),
    Hotel_ID INT,
    Details Varchar(250),
    Start_Date DATE,
    End_Date DATE,
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID)
);
```

Table PROMOTIONS created.
Task completed in 0.08 seconds

➤ Bookings table:

```
create table Bookings(
    ID INT PRIMARY KEY,
    RoomsBooked Varchar(100),
    Booking_Date DATE,
    Checkin DATE,
    Checkout DATE,
    Hotel_ID INT,
    Emp_ID INT,
    Room_Number INT,
    Guest_ID INT,
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID),
    FOREIGN KEY (Guest_ID) REFERENCES Guest(ID),
    FOREIGN KEY (Emp_ID) REFERENCES Employee(ID),
    FOREIGN KEY (Room_Number) REFERENCES Rooms(RoomNumber)
);
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL code for creating the 'Bookings' table. The code includes columns for ID (primary key), RoomsBooked, Booking Date, Checkin Date, Checkout Date, Hotel ID, Employee ID, Room Number, and Guest ID, with various foreign key constraints referencing other tables like 'Hotel', 'Guest', 'Employee', and 'Rooms'. Below the worksheet, the 'Script Output' tab shows the message 'Table BOOKINGS created.' indicating the successful execution of the query.

```

create table Bookings(
    ID INT PRIMARY KEY,
    RoomsBooked Varchar(100),
    Booking_Date DATE,
    Checkin DATE,
    Checkout DATE,
    Hotel_ID INT,
    Emp_ID INT,
    Room_Number INT,
    Guest_ID INT,
    FOREIGN KEY (Hotel_ID) REFERENCES Hotel(ID),
    FOREIGN KEY (Guest_ID) REFERENCES Guest(ID),
    FOREIGN KEY (Emp_ID) REFERENCES Employee(ID),
    FOREIGN KEY (Room_Number) REFERENCES Rooms(RoomNumber)
);

```

Script Output | Task completed in 0.128 seconds

Table BOOKINGS created.

➤ Payments table:

```

create table Payments(
    ID INT PRIMARY KEY,
    Guest_ID INT,
    Booking_ID INT,
    Confirmation_ID INT,
    FOREIGN KEY (Guest_ID) REFERENCES Guest(ID),
    FOREIGN KEY (Booking_ID) REFERENCES Bookings(ID)
);

```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL code for creating the 'Payments' table. The code includes columns for ID (primary key), Guest ID, Booking ID, and Confirmation ID, with foreign key constraints referencing the 'Guest' and 'Bookings' tables. Below the worksheet, the 'Script Output' tab shows the message 'Table PAYMENTS created.' indicating the successful execution of the query.

```

create table Payments(
    ID INT PRIMARY KEY,
    Guest_ID INT,
    Booking_ID INT,
    Confirmation_ID INT,
    FOREIGN KEY (Guest_ID) REFERENCES Guest(ID),
    FOREIGN KEY (Booking_ID) REFERENCES Bookings(ID)
);

```

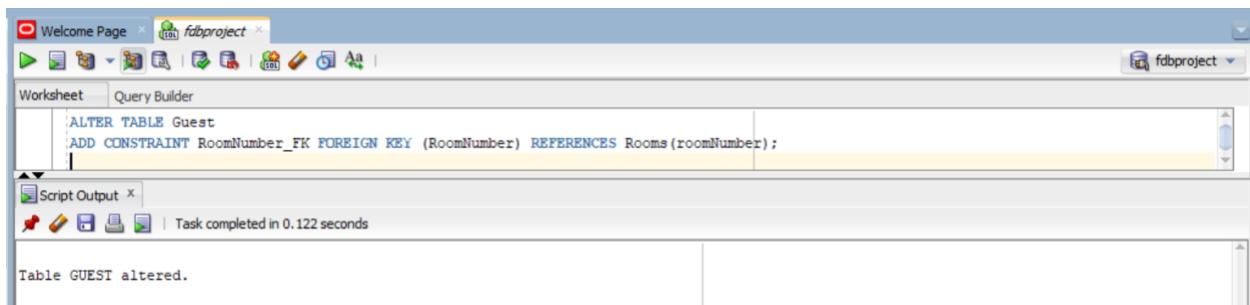
Script Output | Task completed in 0.052 seconds

Table PAYMENTS created.

➤ Alter table Guest:

ALTER TABLE Guest

ADD CONSTRAINT RoomNumber_FK FOREIGN KEY (RoomNumber) REFERENCES Rooms(roomNumber);



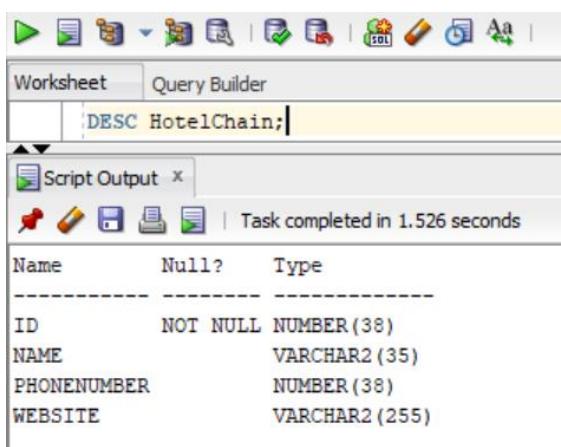
The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL command:

```
ALTER TABLE Guest
ADD CONSTRAINT RoomNumber_FK FOREIGN KEY (RoomNumber) REFERENCES Rooms(roomNumber);
```

Below the worksheet, the 'Script Output' window shows the message "Task completed in 0.122 seconds". In the bottom left corner of the output window, there is a status message: "Table GUEST altered."

TABLE DESCRIPTION

HotelChain



The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying the SQL command:

```
DESC HotelChain;
```

Below the worksheet, the 'Script Output' window shows the message "Task completed in 1.526 seconds". The output window displays the table structure:

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(35)
PHONE NUMBER		NUMBER(38)
WEBSITE		VARCHAR2(255)

Hotel

```
DESC Hotel;
```

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(35)
PHONENUMBER		NUMBER(38)
EMAIL		VARCHAR2(255)
HOTELTYPE		VARCHAR2(25)
CITY	NOT NULL	VARCHAR2(25)
STATE	NOT NULL	VARCHAR2(25)
COUNTRY	NOT NULL	VARCHAR2(25)
ADDRESSLINE1	NOT NULL	VARCHAR2(50)
ADDRESSLINE2	NOT NULL	VARCHAR2(50)
ZIPCODE		NUMBER(38)
DESCRIPTION		VARCHAR2(500)
CAPACITY		NUMBER(38)
RATING		CHAR(1)
HOTELCHAIN_ID		NUMBER(38)

Guest

```
DESC Guest;
```

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(35)
PHONENUMBER		NUMBER(38)
ROOMNUMBER		NUMBER(38)
EMAIL		VARCHAR2(255)
IDPROOF		VARCHAR2(30)
CARDNUMBER		CHAR(16)
ADDRESS		VARCHAR2(255)

Employee

The screenshot shows the Oracle SQL Developer interface. The title bar has two tabs: "Welcome Page" and "fdoproject". The toolbar contains various icons for database operations. The main area has two tabs: "Worksheet" and "Query Builder", with "Worksheet" selected. The worksheet pane contains the following SQL command:

```
DESC Employee;
```

Below the worksheet is a "Script Output" pane with the message "Task completed in 0.585 seconds". The output pane displays the column definitions for the "Employee" table:

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(35)
PHONE NUMBER		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)
SUPERVISORID		NUMBER(38)

Attendance

The screenshot shows the Oracle SQL Developer interface. The title bar has two tabs: "Welcome Page" and "fdoproject". The toolbar contains various icons for database operations. The main area has two tabs: "Worksheet" and "Query Builder", with "Worksheet" selected. The worksheet pane contains the following SQL command:

```
DESC Attendance;
```

Below the worksheet is a "Script Output" pane with the message "Task completed in 0.761 seconds". The output pane displays the column definitions for the "Attendance" table:

Name	Null?	Type
EMP_ID	NOT NULL	NUMBER(38)
ATTEND_DATE	NOT NULL	DATE
HOURS		NUMBER(38)

Amenities

The screenshot shows the Oracle SQL Developer interface. The title bar says "Welcome Page" and "fdbproject". The toolbar has various icons for database operations. The "Worksheet" tab is selected, showing the query "DESC Amenities;". Below it, the "Script Output" tab shows the results of the query:

Name	Null?	Type
HOTEL_ID		NUMBER (38)
ID	NOT NULL	NUMBER (38)
STEAMROOM		VARCHAR2 (3)
AUDITORIUM		VARCHAR2 (3)
PLAYAREA		VARCHAR2 (3)
POOL		VARCHAR2 (3)
MEETINGROOM		VARCHAR2 (3)
PARKING		VARCHAR2 (3)
GAMEROOM		VARCHAR2 (3)
GYM		VARCHAR2 (3)
RESTAURANTS		VARCHAR2 (3)
MEDICALFACILITIES		VARCHAR2 (3)
CHILDDAYCARE		VARCHAR2 (3)

The output also includes a note: "Task completed in 0.254 seconds".

Rooms

The screenshot shows the Oracle SQL Developer interface. The title bar says "Welcome Page" and "fdbproject". The toolbar has various icons for database operations. The "Worksheet" tab is selected, showing the query "DESC Rooms;". Below it, the "Script Output" tab shows the results of the query:

Name	Null?	Type
HOTEL_ID		NUMBER (38)
ROOMNUMBER	NOT NULL	NUMBER (38)
FLOORNUMBER		NUMBER (38)

The output also includes a note: "Task completed in 0.924 seconds".

StarRating

The screenshot shows the SQL Developer interface with a project named "fabproject". In the Worksheet tab, the command `DESC StarRating;` is entered. Below the worksheet, the Script Output window shows the execution results:

Name	Null?	Type
HOTEL_ID		NUMBER (38)
ID	NOT NULL	NUMBER (38)
RATING		CHAR (1)
GUEST_ID		NUMBER (38)
FEEDBACK		VARCHAR2 (500)

EmployeeRating

The screenshot shows the SQL Developer interface with a project named "fabproject". In the Worksheet tab, the command `DESC EmployeeRating;` is entered. Below the worksheet, the Script Output window shows the execution results:

Name	Null?	Type
EMP_ID	NOT NULL	NUMBER (38)
RATING		CHAR (1)
GUEST_ID	NOT NULL	NUMBER (38)

Discount

The screenshot shows the SQL Developer interface with a project named "fabproject". In the Worksheet tab, the command `DESC Discount;` is entered. Below the worksheet, the Script Output window shows the execution results:

Name	Null?	Type
HOTEL_ID	NOT NULL	NUMBER (38)
REWARDPOINTS		NUMBER (38)
DISCOUNTRATE		NUMBER (38)
GUEST_ID	NOT NULL	NUMBER (38)

RoomType

The screenshot shows the fdbproject interface with the 'Worksheet' tab selected. In the query editor, the command `DESC RoomType;` is entered. The results show the following table structure:

Name	Null?	Type
ROOMTYPE	NOT NULL	VARCHAR2 (25)
ROOMNUMBER	NOT NULL	NUMBER (38)
COST		NUMBER (38)
DESCRIPTION		VARCHAR2 (250)

HotelService

The screenshot shows the fdbproject interface with the 'Worksheet' tab selected. In the query editor, the command `DESC HotelService;` is entered. The results show the following table structure:

Name	Null?	Type
ID	NOT NULL	NUMBER (38)
NAME		VARCHAR2 (30)
HOTEL_ID		NUMBER (38)
COST		NUMBER (38)

Packages

The screenshot shows the Oracle SQL Developer interface. The title bar has two tabs: "Welcome Page" and "fabproject". The toolbar includes icons for running queries, saving, and connecting. The main menu bar has "Worksheet" and "Query Builder" selected. The central workspace contains a query editor with the text "DESC Packages;". Below it is a "Script Output" window showing the results of the query. The output table has columns "Name", "Null?", and "Type". The data rows are:

Name	Null?	Type
ID	NOT NULL	NUMBER (38)
NAME		VARCHAR2 (30)
HOTEL_ID		NUMBER (38)
COST		NUMBER (38)
DURATION		NUMBER (38)
VALID_TILL		DATE
DESCRIPTION		VARCHAR2 (200)

The "Script Output" window also displays a message: "Task completed in 0.574 seconds".

Maintenance:

The screenshot shows the Oracle SQL Developer interface. The title bar has two tabs: "Welcome Page" and "fabproject". The toolbar includes icons for running queries, saving, and connecting. The main menu bar has "Worksheet" and "Query Builder" selected. The central workspace contains a query editor with the text "DESC Maintenance;". Below it is a "Script Output" window showing the results of the query. The output table has columns "Name", "Null?", and "Type". The data rows are:

Name	Null?	Type
ID	NOT NULL	NUMBER (38)
MAINTENANCE_TYPE		VARCHAR2 (30)
HOTEL_ID		NUMBER (38)
EMP_ID		NUMBER (38)
ROOM_NUMBER		NUMBER (38)

The "Script Output" window also displays a message: "Task completed in 0.587 seconds".

Promotions:

The screenshot shows the Oracle SQL Developer interface. The title bar has tabs for 'Welcome Page' and 'fdbproject'. The toolbar includes icons for running queries, saving, and navigating. The main area has tabs for 'Worksheet' and 'Query Builder', with 'Worksheet' selected. The query window contains the command: 'DESC Promotions;'. Below it, the 'Script Output' window shows the results: 'Task completed in 0.557 seconds'. The output window displays the table structure:

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
NAME		VARCHAR2(30)
HOTEL_ID		NUMBER(38)
DETAILS		VARCHAR2(250)
START_DATE		DATE
END_DATE		DATE

Bookings

The screenshot shows the Oracle SQL Developer interface. The title bar has tabs for 'Welcome Page' and 'fdbproject'. The toolbar includes icons for running queries, saving, and navigating. The main area has tabs for 'Worksheet' and 'Query Builder', with 'Worksheet' selected. The query window contains the command: 'DESC Bookings;'. Below it, the 'Script Output' window shows the results: 'Task completed in 0.598 seconds'. The output window displays the table structure:

Name	Null?	Type
ID	NOT NULL	NUMBER(38)
ROOMSBOOKED		VARCHAR2(100)
BOOKING_DATE		DATE
CHECKIN		DATE
CHECKOUT		DATE
HOTEL_ID		NUMBER(38)
EMP_ID		NUMBER(38)
ROOM_NUMBER		NUMBER(38)
GUEST_ID		NUMBER(38)

Payments

```

Welcome Page × fdbproject ×
Worksheet Query Builder
DESC Payments;

Script Output ×
Task completed in 0.604 seconds

Name Null? Type
-----
ID NOT NULL NUMBER (38)
GUEST_ID NUMBER (38)
BOOKING_ID NUMBER (38)
CONFIRMATION_ID NUMBER (38)

```

INSERTION OF TUPLES:

- **HotelChain:** QUEHOTEL is the HotelChain in the database. So, It just have one value as entry.

```
INSERT INTO HotelChain values(2024,'QueHotel',9376452848,'www.QueHotelChain.com');
```

```

Welcome Page × fdbproject ×
Worksheet Query Builder
0.20999999 seconds
INSERT INTO HotelChain values(2024,'QueHotel',9376452848,'www.QueHotelChain.com');

Script Output ×
Task completed in 0.21 seconds

1 row inserted.

```

- **Hotel:**

```
-- Insert 1
```

```
INSERT INTO Hotel VALUES(4567, 'Grand Resort', 9876543210,
'grandresort@quehotelchain.com', 'Resort', 'Miami', 'Florida', 'USA', '123 Ocean Avenue', 'Suite
1001', 33139, 'Luxurious beachfront resort', 200, 'A', 2024);
```

```
-- Insert 2
```

```
INSERT INTO Hotel VALUES(5678, 'Cozy Inn', 7543209876, 'cozyinn@quehotelchain.com',
'Bed and Breakfast', 'Aspen', 'Colorado', 'USA', '456 Mountain Road', 'Apt 202', 81611, 'Quaint
bed and breakfast in the mountains', 15, 'B', 2024);
```

-- Insert 3

```
INSERT INTO Hotel VALUES(6789, 'Sunset Lodge', 5412369870,  
'sunsetlodge@quehotelchain.com', 'Hotel', 'Santa Barbara', 'California', 'USA', '789 Seaside  
Boulevard', 'Unit 500', 93101, 'Charming seaside hotel with ocean views', 100, 'C', 2024);
```

-- Insert 4

```
INSERT INTO Hotel VALUES(7890, 'Mountain View Retreat', 7896541230,  
'mountainview@quehotelchain.com', 'Resort', 'Lake Tahoe', 'Nevada', 'USA', '321 Pinecrest  
Drive', 'Cabin 20', 89452, 'Serenity in the heart of the mountains', 50, 'D', 2024);
```

-- Insert 5

```
INSERT INTO Hotel VALUES(8901, 'City Lights Hotel', 1234876509,  
'citylights@quehotelchain.com', 'Hotel', 'New York', 'New York', 'USA', '987 Broadway Street',  
'Floor 30', 10001, 'Stylish urban hotel in the heart of the city', 150, 'E', 2024);
```

-- Insert 6

```
INSERT INTO Hotel VALUES(9012, 'Sunny Beach Resort', 4567890123,  
'sunnybeach@quehotelchain.com', 'Resort', 'San Diego', 'California', 'USA', '654 Sandy Lane',  
'Suite 300', 92109, 'Fun and relaxation by the beach', 300, 'A', 2024);
```

-- Insert 7

```
INSERT INTO Hotel VALUES(1234, 'Mountain Peak Lodge', 9870123456,  
'mountainpeak@quehotelchain.com', 'Hotel', 'Denver', 'Colorado', 'USA', '321 Summit Avenue',  
'Chalet 10', 80202, 'Escape to the Rockies', 80, 'B', 2024);
```

-- Insert 8

```
INSERT INTO Hotel VALUES(2345, 'Riverside Retreat', 3698745210,  
'riverside@quehotelchain.com', 'Resort', 'Portland', 'Oregon', 'USA', '456 Riverwalk Drive', 'Suite  
201', 97204, 'Tranquil riverside escape', 120, 'C', 2024);
```

-- Insert 9

```
INSERT INTO Hotel VALUES(3456, 'Harbor View Hotel', 6325489701,  
'harborview@quehotelchain.com', 'Hotel', 'Seattle', 'Washington', 'USA', '789 Marina Drive', 'Pier  
15', 98101, 'Scenic views of the harbor', 90, 'D', 2024);
```

-- Insert 10

```
INSERT INTO Hotel VALUES(6790, 'The Vineyard Inn', 9854712360,  
'vineyardinn@quehotelchain.com', 'Bed and Breakfast', 'Napa Valley', 'California', 'USA', '101  
Grapevine Lane', 'Vineyard Cottage', 94558, 'Relax in the heart of wine country', 25, 'E', 2024);
```

```

-- Insert 1
INSERT INTO Hotel VALUES(4567, 'Grand Resort', 9876543210, 'grandresort@quehotelchain.com', 'Resort', 'Miami', 'Florida', 'USA', '123 Ocean Avenue', 'Suite 1001', 33139,
-- Insert 2
INSERT INTO Hotel VALUES(5678, 'Cozy Inn', 7543209876, 'cozyinn@quehotelchain.com', 'Bed and Breakfast', 'Aspen', 'Colorado', 'USA', '456 Mountain Road', 'Apt 202', 816,
-- Insert 3
INSERT INTO Hotel VALUES(6789, 'Sunset Lodge', 5412369870, 'sunsetlodge@quehotelchain.com', 'Hotel', 'Santa Barbara', 'California', 'USA', '789 Seaside Boulevard', 'Unit 101',
-- Insert 4
INSERT INTO Hotel VALUES(7890, 'Mountain View Retreat', 7896541230, 'mountainview@quehotelchain.com', 'Resort', 'Lake Tahoe', 'Nevada', 'USA', '321 Pinecrest Drive', 'Cabin 10',
-- Insert 5
INSERT INTO Hotel VALUES(8901, 'City Lights Hotel', 12345676509, 'citylights@quehotelchain.com', 'Hotel', 'New York', 'New York', 'USA', '987 Broadway Street', 'Floor 30',
-- Insert 6
INSERT INTO Hotel VALUES(9012, 'Sunny Beach Resort', 4567890123, 'sunnybeach@quehotelchain.com', 'Resort', 'San Diego', 'California', 'USA', '654 Sandy Lane', 'Suite 300',
-- Insert 7
INSERT INTO Hotel VALUES(1234, 'Mountain Peak Lodge', 9870123456, 'mountainpeak@quehotelchain.com', 'Hotel', 'Denver', 'Colorado', 'USA', '321 Summit Avenue', 'Chalet 101',
-- Insert 8
INSERT INTO Hotel VALUES(2345, 'Riverside Retreat', 3698745210, 'riverside@quehotelchain.com', 'Resort', 'Portland', 'Oregon', 'USA', '456 Riverwalk Drive', 'Suite 201',
-- Insert 9
INSERT INTO Hotel VALUES(3456, 'Harbor View Hotel', 6325489701, 'harborview@quehotelchain.com', 'Hotel', 'Seattle', 'Washington', 'USA', '789 Marina Drive', 'Pier 15', 'Suite 101',
-- Insert 10
INSERT INTO Hotel VALUES(6790, 'The Vineyard Inn', 9854712360, 'vineyardinn@quehotelchain.com', 'Bed and Breakfast', 'Napa Valley', 'California', 'USA', '101 Grapevine Lane')

```

Script Output | Task completed in 15.93 seconds
1 row inserted.

Script Output | Task completed in 15.93 seconds
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.

➤ Amenities:

-- Insert 1

INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)

VALUES (4567, 11, 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 2

INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)

VALUES (5678, 12, 'NO', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO', 'YES', 'NO', 'NO');

--Insert 3

INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)

VALUES (6789, 13, 'YES', 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 4

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (7890, 14, 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO');

--Insert 5
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (8901, 15, 'NO', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO');

--Insert 6
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (9012, 16, 'YES', 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 7
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (1234, 17, 'NO', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO', 'NO', 'YES', 'NO', 'NO');

--Insert 8
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (2345, 18, 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 9
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (3456, 19, 'NO', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO');

--Insert 10
```

```
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool,
MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (6790, 20, 'YES', 'NO', 'NO', 'YES', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO');
```

```
-- Insert 1
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (4567, 11, 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 2
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (5678, 12, 'NO', 'YES', 'NO', 'YES', 'YES', 'NO', 'YES', 'NO', 'NO');

--Insert 3
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (6789, 13, 'YES', 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 4
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (7890, 14, 'YES', 'NO', 'YES', 'YES', 'NO', 'YES', 'YES', 'NO', 'NO');

--Insert 5
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (8901, 15, 'NO', 'YES', 'YES', 'NO', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO');

--Insert 6
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (9012, 16, 'YES', 'NO', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 7
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (1234, 17, 'NO', 'YES', 'NO', 'NO', 'YES', 'NO', 'NO', 'YES', 'NO', 'NO');

--Insert 8
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
```

Script Output X | Task completed in 0.511 seconds

1 row inserted.

```
-- Insert 6
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (2345, 18, 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'YES', 'NO', 'NO');

--Insert 9
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (3456, 19, 'NO', 'YES', 'YES', 'NO', 'YES', 'NO', 'NO', 'YES', 'YES', 'NO');

--Insert 10
INSERT INTO Amenities (Hotel_ID, ID, SteamRoom, Auditorium, PlayArea, Pool, MeetingRoom, Parking, GameRoom, Gym, Restaurants, MedicalFacilities, ChildDayCare)
VALUES (6790, 20, 'YES', 'NO', 'NO', 'YES', 'YES', 'NO', 'YES', 'NO', 'NO');
```

Script Output X | Task completed in 0.511 seconds

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

➤ Employee:

--Insert 1

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

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, NULL);

--Insert 2

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

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, 1001);

--Insert 3

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

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, 1001);

--Insert 4

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

VALUES (1004, 'Emily Brown', 4567890123, 'emily.brown@example.com', '321 Pinecrest Dr, Lake Tahoe, NV, USA', 'Housekeeping Supervisor', 'None', 'High School Diploma', 'Hospitality Certification', 'Housekeeping', 7890, 1001);

--Insert 5

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

VALUES (1005, 'David Miller', 5678901234, 'david.miller@example.com', '987 Broadway St, New York, NY, USA', 'Concierge', 'None', 'Bachelor of Arts in Hospitality Management', 'Concierge Certification', 'Concierge', 8901, 1001);

--Insert 6

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

VALUES (1006, 'Jessica Davis', 6789012345, 'jessica.davis@example.com', '654 Sandy Ln, San Diego, CA, USA', 'Spa Therapist', 'None', 'Certified Massage Therapist', 'Spa Certification', 'Spa', 9012, 1001);

--Insert 7

INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)

VALUES (1007, 'Christopher Wilson', 7890123456, 'christopher.wilson@example.com', '321 Summit Ave, Denver, CO, USA', 'Maintenance Technician', 'None', 'High School Diploma', 'Maintenance Certification', 'Maintenance', 1234, 1001);

--Insert 8

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,  
CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
```

```
VALUES (1008, 'Sarah Martinez', 8901234567, 'sarah.martinez@example.com', '456 Riverwalk  
Dr, Portland, OR, USA', 'Event Coordinator', 'None', 'Bachelor of Arts in Event Management',  
'Event Planning Certification', 'Events', 2345, 1001);
```

--Insert 9

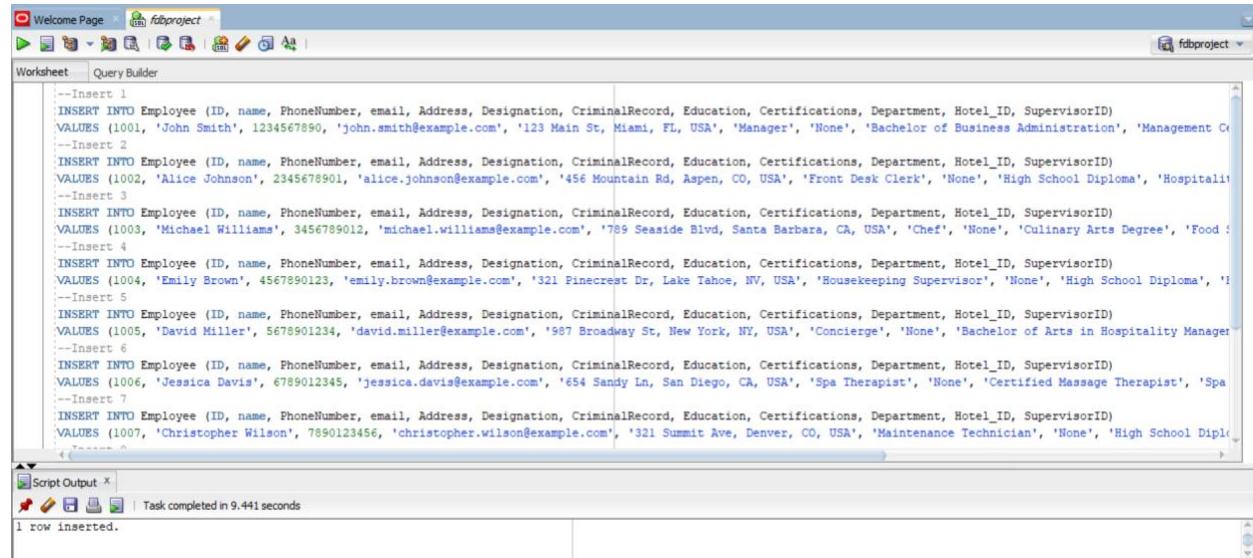
```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,  
CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
```

```
VALUES (1009, 'Daniel Anderson', 9012345678, 'daniel.anderson@example.com', '789 Marina  
Dr, Seattle, WA, USA', 'Security Officer', 'None', 'High School Diploma', 'Security Certification',  
'Security', 3456, 1001);
```

--Insert 10

```
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,  
CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
```

```
VALUES (1010, 'Olivia Garcia', 1234876509, 'olivia.garcia@example.com', '101 Grapevine Ln,  
Napa Valley, CA, USA', 'Wine Sommelier', 'None', 'Certified Sommelier', 'Wine Tasting  
Certification', 'Food and Beverage', 6790, 1001);
```



The screenshot shows the Oracle SQL Developer interface with the 'Worksheet' tab selected. The main pane displays the following SQL script:

```
--Insert 1  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1001, 'John Smith', 1234567890, 'john.smith@example.com', '123 Main St, Miami, FL, USA', 'Manager', 'None', 'Bachelor of Business Administration', 'Management Ce  
--Insert 2  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1002, 'Alice Johnson', 2345678901, 'alice.johnson@example.com', '456 Mountain Rd, Aspen, CO, USA', 'Front Desk Clerk', 'None', 'High School Diploma', 'Hospitalit  
--Insert 3  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1003, 'Michael Williams', 3456789012, 'michael.williams@example.com', '789 Seaside Blvd, Santa Barbara, CA, USA', 'Chef', 'None', 'Culinary Arts Degree', 'Food :  
--Insert 4  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1004, 'Emily Brown', 4567890123, 'emily.brown@example.com', '321 Pinecrest Dr, Lake Tahoe, NV, USA', 'Housekeeping Supervisor', 'None', 'High School Diploma', 'H  
--Insert 5  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1005, 'David Miller', 5678901234, 'david.miller@example.com', '987 Broadway St, New York, NY, USA', 'Concierge', 'None', 'Bachelor of Arts in Hospitality Manager  
--Insert 6  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1006, 'Jessica Davis', 6789012345, 'jessica.davis@example.com', '654 Sandy Ln, San Diego, CA, USA', 'Spa Therapist', 'None', 'Certified Massage Therapist', 'Spa  
--Insert 7  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1007, 'Christopher Wilson', 7890123456, 'christopher.wilson@example.com', '321 Summit Ave, Denver, CO, USA', 'Maintenance Technician', 'None', 'High School Dipl  
--Insert 8  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1008, 'Sarah Martinez', 8901234567, 'sarah.martinez@example.com', '456 Riverwalk  
Dr, Portland, OR, USA', 'Event Coordinator', 'None', 'Bachelor of Arts in Event Management',  
'Event Planning Certification', 'Events', 2345, 1001);  
--Insert 9  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,  
CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1009, 'Daniel Anderson', 9012345678, 'daniel.anderson@example.com', '789 Marina  
Dr, Seattle, WA, USA', 'Security Officer', 'None', 'High School Diploma', 'Security Certification',  
'Security', 3456, 1001);  
--Insert 10  
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation,  
CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)  
VALUES (1010, 'Olivia Garcia', 1234876509, 'olivia.garcia@example.com', '101 Grapevine Ln,  
Napa Valley, CA, USA', 'Wine Sommelier', 'None', 'Certified Sommelier', 'Wine Tasting  
Certification', 'Food and Beverage', 6790, 1001);
```

The bottom status bar indicates "Task completed in 9.441 seconds" and "1 row inserted."

```

--Insert 8
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
VALUES (1008, 'Sarah Martinez', 8901234567, 'sarah.martinez@example.com', '456 Riverwalk Dr, Portland, OR, USA', 'Event Coordinator', 'None', 'Bachelor of Arts in Event
--Insert 9
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
VALUES (1009, 'Daniel Anderson', 9012345678, 'daniel.anderson@example.com', '789 Marina Dr, Seattle, WA, USA', 'Security Officer', 'None', 'High School Diploma', 'Secur
--Insert 10
INSERT INTO Employee (ID, name, PhoneNumber, email, Address, Designation, CriminalRecord, Education, Certifications, Department, Hotel_ID, SupervisorID)
VALUES (1010, 'Olivia Garcia', 1234567890, 'olivia.garcia@example.com', '101 Grapevine Ln, Napa Valley, CA, USA', 'Wine Sommelier', 'None', 'Certified Sommelier', 'Wine

```

Script Output X | Task completed in 9.441 seconds

1 row inserted.

➤ Rooms:

--Insert 1

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (4567, 1001, 1);

--Insert 2

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (4567, 1002, 1);

--Insert 3

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (5678, 2001, 2);

--Insert 4

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (5678, 2002, 2);

--Insert 5

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (6789, 3001, 3);

--Insert 6

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (6789, 3002, 3);

--Insert 7

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (7890, 4001, 4);

--Insert 8

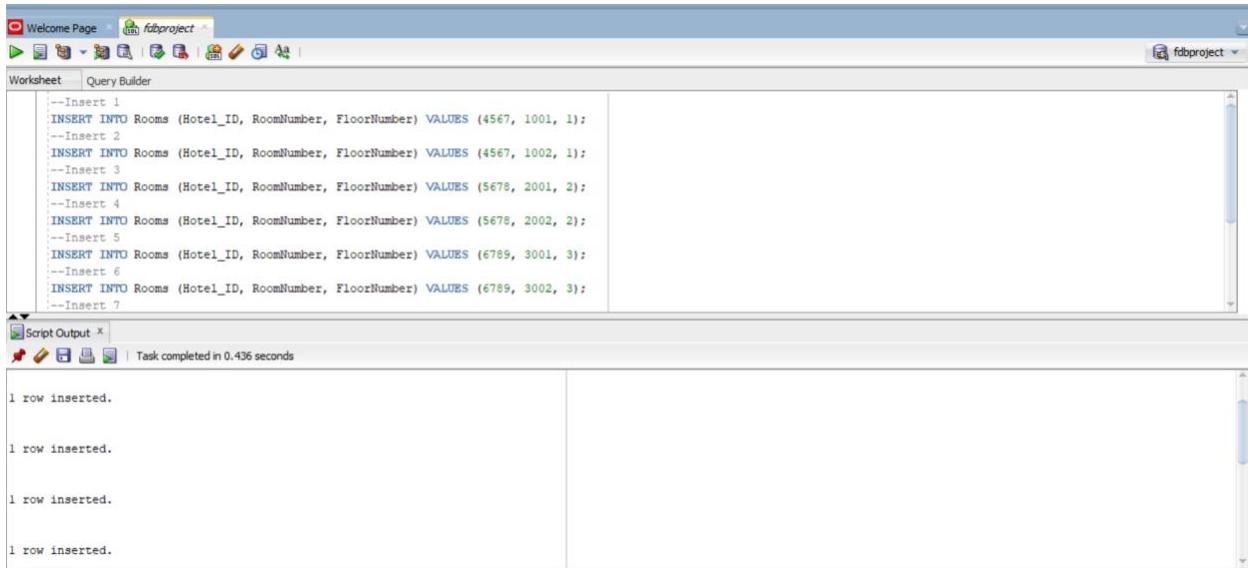
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (7890, 4002, 4);

--Insert 9

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (8901, 5001, 5);

--Insert 10

INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (8901, 5002, 5);



The screenshot shows the Oracle SQL Developer interface with two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab contains the following SQL script:

```
--Insert 1
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (4567, 1001, 1);
--Insert 2
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (4567, 1002, 1);
--Insert 3
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (5678, 2001, 2);
--Insert 4
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (5678, 2002, 2);
--Insert 5
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (6789, 3001, 3);
--Insert 6
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (6789, 3002, 3);
--Insert 7
```

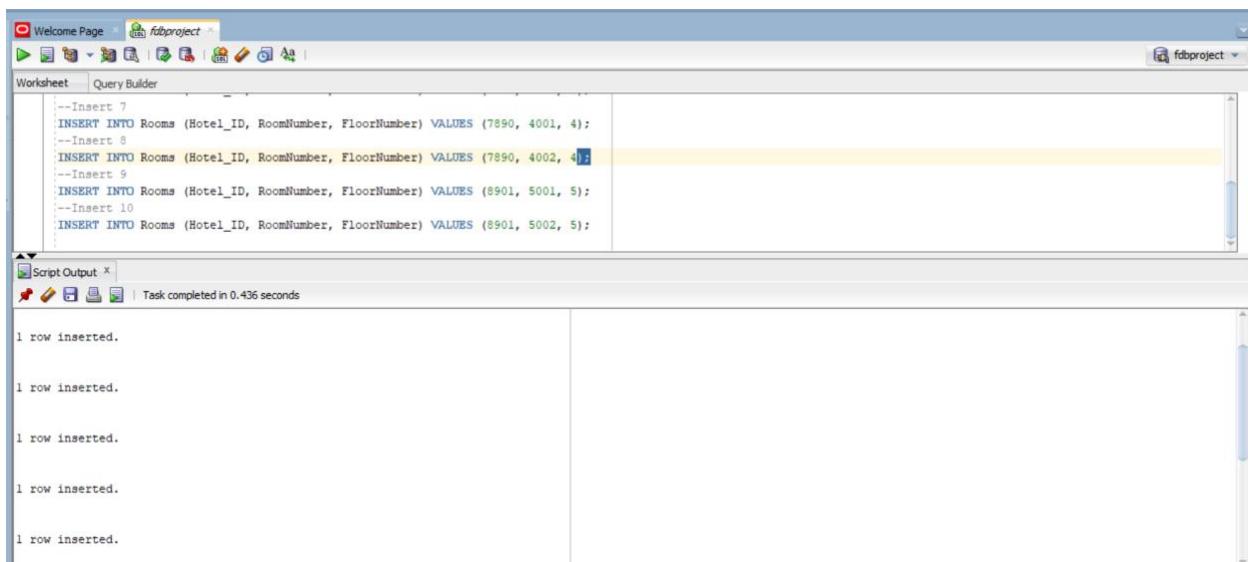
The 'Script Output' tab below shows the results of the execution:

```
1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```



The screenshot shows the Oracle SQL Developer interface with two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab contains the following SQL script:

```
--Insert 7
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (7890, 4001, 4);
--Insert 8
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (7890, 4002, 4);
--Insert 9
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (8901, 5001, 5);
--Insert 10
INSERT INTO Rooms (Hotel_ID, RoomNumber, FloorNumber) VALUES (8901, 5002, 5);
```

The 'Script Output' tab below shows the results of the execution:

```
1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```

➤ Guest:

--Insert 1

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2001, 'Alice Johnson', 1234567890, 1001, 'alice.johnson@gmail.com', 'ABC123XYZ', '1234567890123456', '123 Main St, Miami, FL, USA');

--Insert 2

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2002, 'Bob Smith', 2345678901, 1002, 'bob.smith@gmail.com', 'DEF456WXY', '2345678901234567', '456 Mountain Rd, Aspen, CO, USA');

--Insert 3

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2003, 'Charlie Brown', 3456789012, 2001, 'charlie.brown@gmail.com', 'GHI789UVW', '3456789012345678', '789 Seaside Blvd, Santa Barbara, CA, USA');

--Insert 4

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2004, 'David Miller', 4567890123, 2002, 'david.miller@gmail.com', 'JKL012PQR', '4567890123456789', '321 Pinecrest Dr, Lake Tahoe, NV, USA');

--Insert 5

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2005, 'Emma Davis', 5678901234, 3001, 'emma.davis@gmail.com', 'MNO345STU', '5678901234567890', '987 Broadway St, New York, NY, USA');

--Insert 6

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2006, 'Frank Wilson', 6789012345, 3002, 'frank.wilson@gmail.com', 'VWX678YZA', '6789012345678901', '654 Sandy Ln, San Diego, CA, USA');

--Insert 7

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

VALUES (2007, 'Grace Lee', 7890123456, 4001, 'grace.lee@gmail.com', 'BCD901LMN', '7890123456789012', '321 Summit Ave, Denver, CO, USA');

--Insert 8

INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)

```
VALUES (2008, 'Henry Garcia', 8901234567, 4002, 'henry.garcia@gmail.com', 'EFG234OPQ',  
'8901234567890123', '456 Riverwalk Dr, Portland, OR, USA');
```

--Insert 9

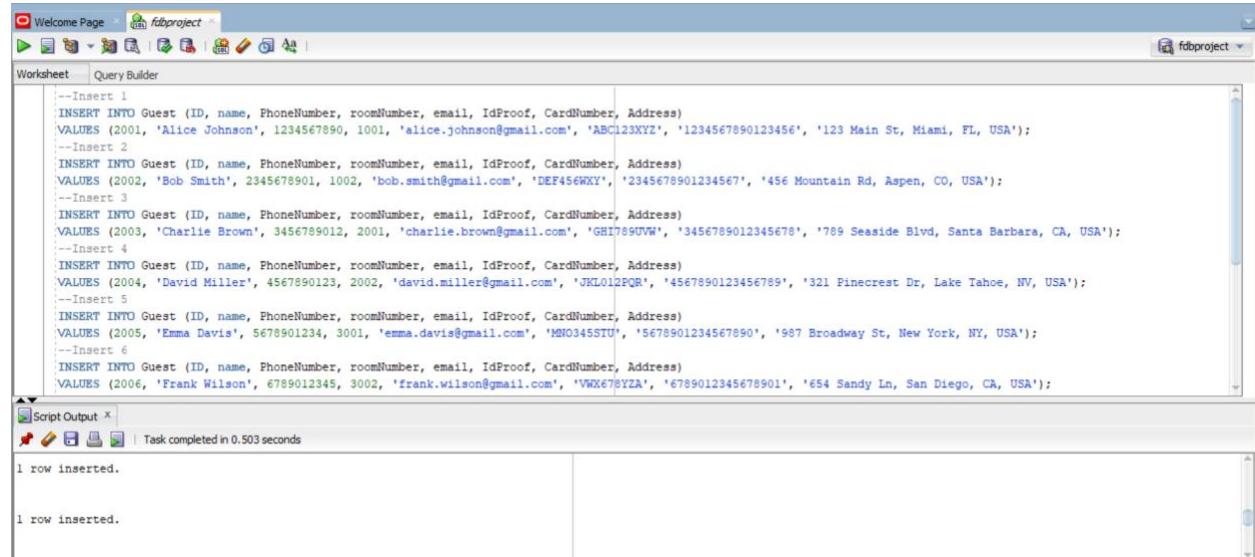
```
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber,  
Address)
```

```
VALUES (2009, 'Ivy Martinez', 9012345678, 5001, 'ivy.martinez@gmail.com', 'RST567BCD',  
'9012345678901234', '789 Marina Dr, Seattle, WA, USA');
```

--Insert 10

```
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber,  
Address)
```

```
VALUES (2010, 'Jack Thompson', 1234876509, 5002, 'jack.thompson@gmail.com',  
'YZA678CDE', '1234876509876543', '101 Grapevine Ln, Napa Valley, CA, USA');
```



The screenshot shows the Oracle SQL Developer interface with a script editor window titled 'fdoproject'. The script contains 10 INSERT statements, each preceded by a comment '--Insert n' where n ranges from 1 to 10. The script inserts data for various guests with IDs from 2001 to 2010, names like Alice Johnson, Bob Smith, etc., and addresses across different cities and states. The bottom panel shows the 'Script Output' tab with the message 'Task completed in 0.503 seconds' and two '1 row inserted.' messages corresponding to the first two insertions.

```
--Insert 1  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2001, 'Alice Johnson', 1234567890, 1001, 'alice.johnson@gmail.com', 'ABC123XYZ', '123 Main St, Miami, FL, USA');  
--Insert 2  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2002, 'Bob Smith', 2345678901, 1002, 'bob.smith@gmail.com', 'DEF456WXY', '2345678901234567', '456 Mountain Rd, Aspen, CO, USA');  
--Insert 3  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2003, 'Charlie Brown', 3456789012, 2001, 'charlie.brown@gmail.com', 'GHI789UUVW', '3456789012345678', '789 Seaside Blvd, Santa Barbara, CA, USA');  
--Insert 4  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2004, 'David Miller', 4567890123, 2002, 'david.miller@gmail.com', 'JKL012PQR', '4567890123456789', '321 Pinecrest Dr, Lake Tahoe, NV, USA');  
--Insert 5  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2005, 'Emma Davis', 5678901234, 3001, 'emma.davis@gmail.com', 'MNO345STU', '5678901234567890', '987 Broadway St, New York, NY, USA');  
--Insert 6  
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)  
VALUES (2006, 'Frank Wilson', 6789012345, 3002, 'frank.wilson@gmail.com', 'VWX678YZA', '6789012345678901', '654 Sandy Ln, San Diego, CA, USA');
```

```

--Insert 7
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)
VALUES (2007, 'Grace Lee', 7890123456, 4001, 'grace.lee@gmail.com', 'BCD901LMN', '7890123456789012', '321 Summit Ave, Denver, CO, USA');

--Insert 8
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)
VALUES (2008, 'Henry Garcia', 8901234567, 4002, 'henry.garcia@gmail.com', 'EFG234OPQ', '8901234567890123', '456 Riverwalk Dr, Portland, OR, USA');

--Insert 9
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)
VALUES (2009, 'Ivy Martinez', 9012345678, 5001, 'ivy.martinez@gmail.com', 'RST567BCD', '9012345678901234', '789 Marina Dr, Seattle, WA, USA');

--Insert 10
INSERT INTO Guest (ID, name, PhoneNumber, roomNumber, email, IdProof, CardNumber, Address)
VALUES (2010, 'Jack Thompson', 1234876509, 5002, 'jack.thompson@gmail.com', 'YZA678CDE', '1234876509876543', '101 Grapevine Ln, Napa Valley, CA, USA');

```

Script Output X | Task completed in 0.503 seconds

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

➤ Discount:

--Insert 1

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (4567, 100, 10, 2001);

--Insert 2

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (5678, 150, 15, 2002);

--Insert 3

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (6789, 200, 20, 2003);

--Insert 4

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (7890, 250, 25, 2004);

--Insert 5

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (8901, 300, 30, 2005);

--Insert 6

INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (4567, 350, 35, 2006);

--Insert 7

```
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (5678, 400, 40, 2007);
```

--Insert 8

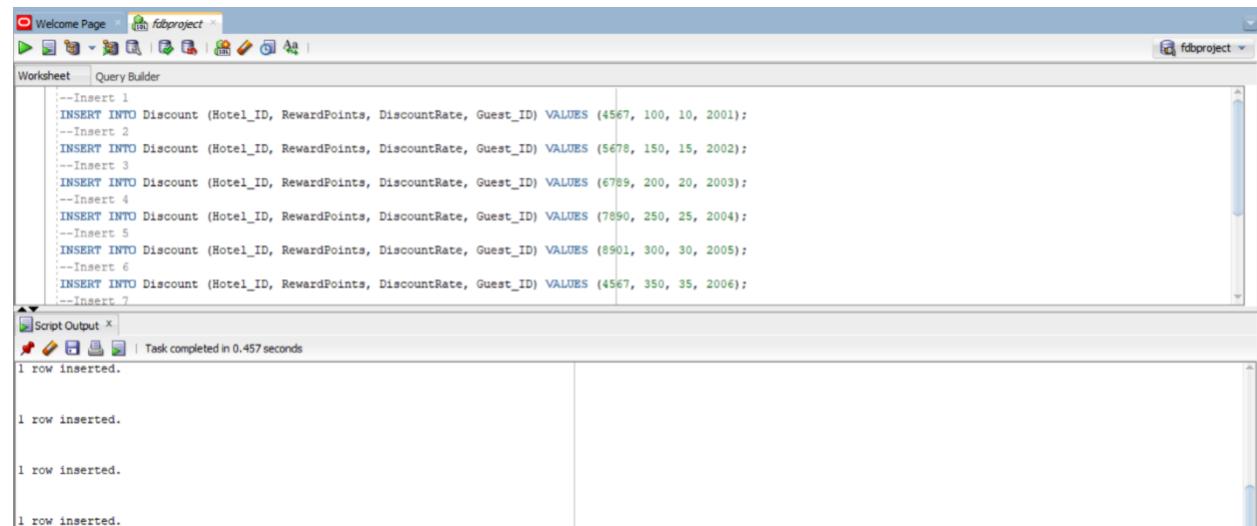
```
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (6789, 450, 45, 2008);
```

--Insert 9

```
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (7890, 500, 50, 2009);
```

--Insert 10

```
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (8901, 550, 55, 2010);
```



The screenshot shows the Oracle SQL Developer interface with the 'fdoproject' workspace selected. The 'Worksheet' tab is active, displaying a script window with 10 INSERT statements labeled --Insert 1 through --Insert 10. The 'Script Output' window below shows the results of each insert operation, indicating '1 row inserted.' for each statement. The status bar at the bottom right of the output window says 'Task completed in 0.457 seconds'.

```
--Insert 1
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (4567, 100, 10, 2001);
--Insert 2
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (5678, 150, 15, 2002);
--Insert 3
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (6789, 200, 20, 2003);
--Insert 4
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (7890, 250, 25, 2004);
--Insert 5
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (8901, 300, 30, 2005);
--Insert 6
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (4567, 350, 35, 2006);
--Insert 7
```

Script Output | Task completed in 0.457 seconds

```
1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying a block of SQL code. The code consists of several INSERT statements into the 'Discount' table, each followed by a comment indicating the insert number (e.g., '--Insert 7'). The SQL statements are:

```
--Insert 7
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (5678, 400, 40, 2007);
--Insert 8
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (6789, 450, 45, 2008);
--Insert 9
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (7890, 500, 50, 2009);
--Insert 10
INSERT INTO Discount (Hotel_ID, RewardPoints, DiscountRate, Guest_ID) VALUES (8901, 550, 55, 2010);
```

The 'Script Output' window below the worksheet shows the results of the execution. It displays five lines, each stating '1 row inserted.', corresponding to the five INSERT statements above.

➤ EmployeeRating:

--Insert 1

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1001, 'A', 2001);
```

--Insert 2

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1002, 'B', 2002);
```

--Insert 3

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1003, 'C', 2003);
```

--Insert 4

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1004, 'D', 2004);
```

--Insert 5

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1005, 'E', 2005);
```

--Insert 6

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1006, 'A', 2006);
```

--Insert 7

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1007, 'B', 2007);
```

--Insert 8

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1008, 'C', 2008);
```

--Insert 9

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1009, 'D', 2009);
```

--Insert 10

```
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1010, 'E', 2010);
```

The screenshot shows two separate sessions in Oracle SQL Developer. Both sessions are titled 'fdoproject' and have tabs for 'Worksheet' and 'Query Builder'. The 'Worksheet' tab is active in both.

Session 1 (Top):

```
--Insert 1
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1001, 'A', 2001);
--Insert 2
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1002, 'B', 2002);
--Insert 3
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1003, 'C', 2003);
--Insert 4
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1004, 'D', 2004);
--Insert 5
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1005, 'E', 2005);
--Insert 6
```

Session 2 (Bottom):

```
--Insert 6
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1006, 'A', 2006);
--Insert 7
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1007, 'B', 2007);
--Insert 8
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1008, 'C', 2008);
--Insert 9
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1009, 'D', 2009);
--Insert 10
INSERT INTO EmployeeRating (Emp_ID, Rating, Guest_ID) VALUES (1010, 'E', 2010);
```

In the 'Script Output' section of each session, the message 'Task completed in 0.698 seconds' is displayed. The output pane shows the confirmation '1 row inserted.' repeated five times for Session 1 and five times for Session 2.

The screenshot shows two separate sessions in Oracle SQL Developer. Both sessions are titled 'fdoproject' and have tabs for 'Worksheet' and 'Query Builder'. The 'Worksheet' tab is active in both.

Session 1 (Top):

```
--Insert 6
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1011, 'Room Service', 4567, 20);
--Insert 7
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1012, 'Laundry Service', 5678, 30);
```

Session 2 (Bottom):

```
--Insert 1
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1011, 'Room Service', 4567, 20);
--Insert 2
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1012, 'Laundry Service', 5678, 30);
```

In the 'Script Output' section of each session, the message 'Task completed in 0.489 seconds' is displayed. The output pane shows the confirmation '1 row inserted.' repeated once for Session 1 and once for Session 2.

➤ HotelService:

--Insert 1

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1011, 'Room Service', 4567, 20);
```

--Insert 2

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1012, 'Laundry Service', 5678, 30);
```

--Insert 3

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1013, 'Spa Massage', 6789, 50);
```

--Insert 4

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1014, 'Airport Shuttle', 7890, 40);
```

--Insert 5

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1015, 'Concierge Service', 8901, 25);
```

--Insert 6

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1016, 'Fitness Center Access', 4567, 15);
```

--Insert 7

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1017, 'Car Rental', 5678, 60);
```

--Insert 8

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1018, 'Guided Tour', 6789, 35);
```

--Insert 9

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1019, 'Valet Parking', 7890, 20);
```

--Insert 10

```
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1020, 'In-Room Dining', 8901, 25);
```

The image shows two separate sessions in Oracle SQL Developer. Both sessions are titled 'fdoproject' and have a 'Worksheet' tab selected.

Session 1 (Top):

```
--Insert 1
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1011, 'Room Service', 4567, 20);
--Insert 2
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1012, 'Laundry Service', 5678, 30);
--Insert 3
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1013, 'Spa Massage', 6789, 50);
--Insert 4
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1014, 'Airport Shuttle', 7890, 40);
--Insert 5
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1015, 'Concierge Service', 8901, 25);
```

Session 2 (Bottom):

```
--Insert 6
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1016, 'Fitness Center Access', 4567, 15);
--Insert 7
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1017, 'Car Rental', 5678, 60);
--Insert 8
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1018, 'Guided Tour', 6789, 35);
--Insert 9
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1019, 'Valet Parking', 7890, 20);
--Insert 10
INSERT INTO HotelService (ID, Name, Hotel_ID, Cost) VALUES (1020, 'In-Room Dining', 8901, 25);
```

In both sessions, the 'Script Output' pane shows the message 'Task completed in 0.424 seconds' and lists five rows inserted for each session.

➤ StarRating:

--Insert 1

INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (4567, 91876, 'A', 2001, 'Excellent service and beautiful accommodations.');

--Insert 2

INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (5678, 37594, 'B', 2002, 'Comfortable stay, friendly staff.');

--Insert 3

INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (6789, 56218, 'C', 2003, 'Average experience, could improve cleanliness.');

--Insert 4

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (7890, 74029, 'D', 2004, 'Disappointed with room condition and service.');
```

--Insert 5

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (8901, 12683, 'E', 2005, 'Terrible experience, would not recommend.');
```

--Insert 6

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (4567, 58340, 'A', 2006, 'Amazing amenities and attentive staff.');
```

--Insert 7

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (5678, 49237, 'B', 2007, 'Enjoyed the location and atmosphere.');
```

--Insert 8

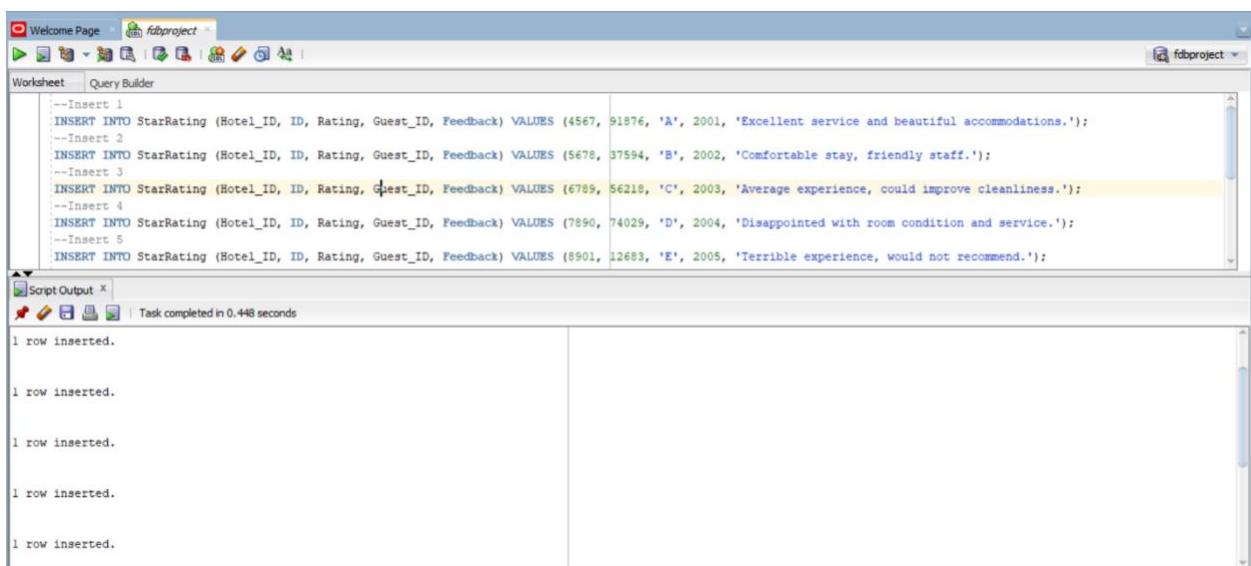
```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (6789, 86541, 'C', 2008, 'Fair service but room was noisy.');
```

--Insert 9

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (7890, 20789, 'D', 2009, 'Poor communication and cleanliness issues.');
```

--Insert 10

```
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (8901, 63452, 'E', 2010, 'Horrible stay, rude staff and unclean rooms.');
```



The screenshot shows the Oracle SQL Developer interface with the 'fdoproject' workspace selected. The 'Worksheet' tab is active, displaying the 10 INSERT statements. The 'Script Output' tab at the bottom shows the confirmation '1 row inserted.' repeated 10 times, indicating successful execution of each statement.

```
--Insert 1
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (4567, 58340, 'A', 2006, 'Amazing amenities and attentive staff.');
--Insert 2
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (5678, 49237, 'B', 2007, 'Enjoyed the location and atmosphere.');
--Insert 3
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (6789, 86541, 'C', 2008, 'Fair service but room was noisy.');
--Insert 4
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (7890, 20789, 'D', 2009, 'Poor communication and cleanliness issues.');
--Insert 5
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (7890, 74029, 'D', 2004, 'Disappointed with room condition and service.');
--Insert 6
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (8901, 12683, 'E', 2005, 'Terrible experience, would not recommend.');
--Insert 7
--Insert 8
--Insert 9
--Insert 10
```

Script Output X | Task completed in 0.448 seconds

1 row inserted.	

```
--Insert 6
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (4567, 58340, 'A', 2006, 'Amazing amenities and attentive staff.');
--Insert 7
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (5678, 49237, 'B', 2007, 'Enjoyed the location and atmosphere.');
--Insert 8
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (6789, 86541, 'C', 2008, 'Fair service but room was noisy.');
--Insert 9
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (7890, 20789, 'D', 2009, 'Poor communication and cleanliness issues.');
--Insert 10
INSERT INTO StarRating (Hotel_ID, ID, Rating, Guest_ID, Feedback) VALUES (8901, 63452, 'E', 2010, 'Horrible stay, rude staff and unclean rooms.');

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```

➤ Maintenance:

-- Insert 1

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (1, 'Plumbing', 4567, 1007, 1001);
```

-- Insert 2

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (2, 'Electrical', 5678, 1008, 2002);
```

-- Insert 3

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (3, 'HVAC', 6789, 1009, 3001);
```

-- Insert 4

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (4, 'Painting', 7890, 1010, 4002);
```

-- Insert 5

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (5, 'Carpentry', 8901, 1007, 5001);
```

-- Insert 6

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (6, 'Plumbing', 4567, 1008, 1002);
```

-- Insert 7

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (7, 'Electrical', 5678, 1009, 2001);
```

-- Insert 8

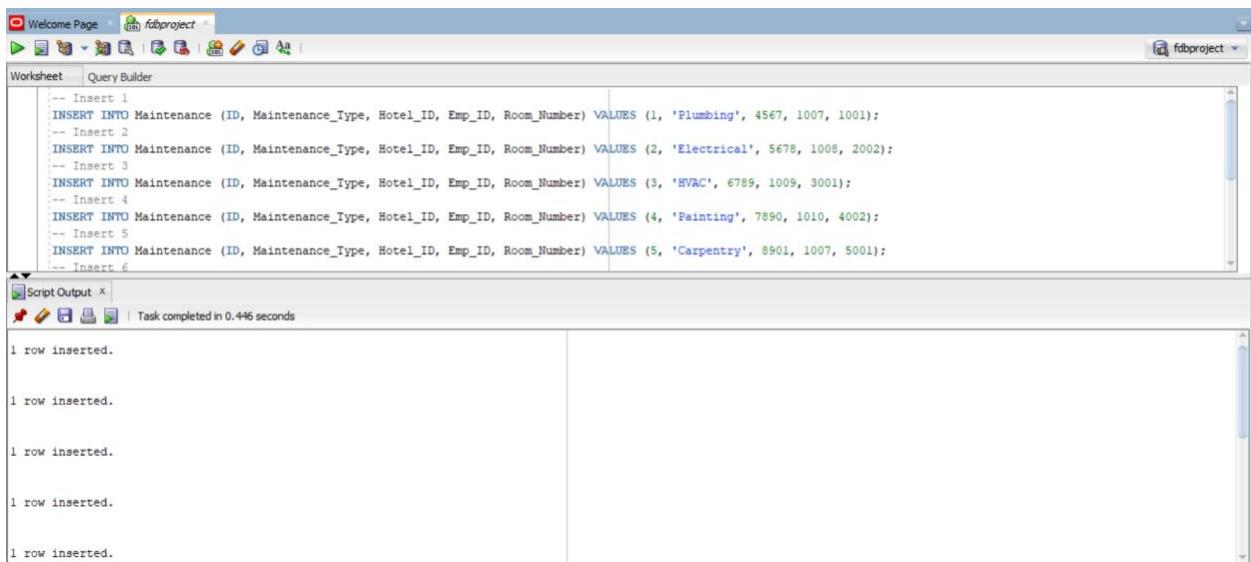
```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (8, 'HVAC', 6789, 1010, 3002);
```

-- Insert 9

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (9, 'Painting', 7890, 1007, 4001);
```

-- Insert 10

```
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number)
VALUES (10, 'Carpentry', 8901, 1008, 5002);
```

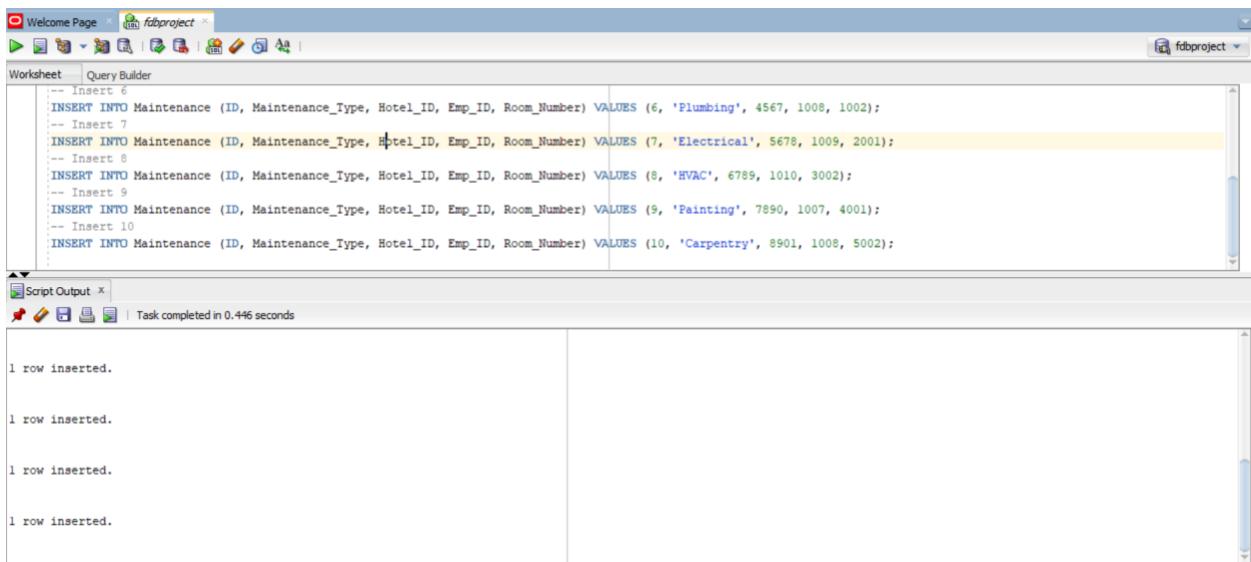


The screenshot shows the Oracle SQL Developer interface with two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab contains the following SQL script:

```
-- Insert 1
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number) VALUES (1, 'Plumbing', 4567, 1007, 1001);
-- Insert 2
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number) VALUES (2, 'Electrical', 5678, 1008, 2002);
-- Insert 3
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number) VALUES (3, 'HVAC', 6789, 1009, 3001);
-- Insert 4
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number) VALUES (4, 'Painting', 7890, 1010, 4002);
-- Insert 5
INSERT INTO Maintenance (ID, Maintenance_Type, Hotel_ID, Emp_ID, Room_Number) VALUES (5, 'Carpentry', 8901, 1007, 5001);
-- Insert 6
-- Insert 7
-- Insert 8
-- Insert 9
-- Insert 10
```

The 'Script Output' pane below the worksheet shows the results of the execution:

```
1 row inserted.
```



The screenshot shows the Oracle SQL Developer interface with two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab contains the same SQL script as the previous screenshot, but the 7th line is highlighted in yellow:

```
-- Insert 6
-- Insert 7
-- Insert 8
-- Insert 9
-- Insert 10
-- Insert 11
-- Insert 12
-- Insert 13
-- Insert 14
-- Insert 15
```

The 'Script Output' pane below the worksheet shows the results of the execution:

```
1 row inserted.
1 row inserted.
1 row inserted.
1 row inserted.
```

➤ **RoomType:**

-- Insert 1

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Single Bed', 1001, 100, 'Comfortable room with a single bed, suitable for solo travelers.');

-- Insert 2

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Single Bed', 1002, 100, 'Comfortable room with a single bed, suitable for solo travelers.');

-- Insert 3

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Double Bed', 2001, 150, 'Spacious room with a double bed, perfect for couples.');

-- Insert 4

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Double Bed', 2002, 150, 'Spacious room with a double bed, perfect for couples.');

-- Insert 5

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Deluxe', 3001, 200, 'Luxurious deluxe room with modern amenities.');

-- Insert 6

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Deluxe', 3002, 200, 'Luxurious deluxe room with modern amenities.');

-- Insert 7

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Single Bed', 4001, 100, 'Comfortable room with a single bed, suitable for solo travelers.');

-- Insert 8

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Single Bed', 4002, 100, 'Comfortable room with a single bed, suitable for solo travelers.');

-- Insert 9

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Double Bed', 5001, 150, 'Spacious room with a double bed, perfect for couples.');

-- Insert 10

INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)

VALUES ('Double Bed', 5002, 150, 'Spacious room with a double bed, perfect for couples.');

The screenshot shows two separate sessions in Oracle SQL Developer. The top session contains the first five inserts (1-5) for room types with single beds. The bottom session contains the last five inserts (6-10) for room types with double beds. Both sessions show the execution results in the 'Script Output' tab, indicating '1 row inserted.' for each query.

```
-- Insert 1
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 1001, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 2
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 1002, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 3
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 2001, 150, 'Spacious room with a double bed, perfect for couples.');
-- Insert 4
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 2002, 150, 'Spacious room with a double bed, perfect for couples.');
-- Insert 5
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Deluxe', 3001, 200, 'Luxurious deluxe room with modern amenities.');

-- Insert 6
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Deluxe', 3002, 200, 'Luxurious deluxe room with modern amenities.');
-- Insert 7
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 4001, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 8
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 4002, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 9
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 5001, 150, 'Spacious room with a double bed, perfect for couples.');
-- Insert 10
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 5002, 150, 'Spacious room with a double bed, perfect for couples.');
```

This screenshot shows the continuation of the room type insertions. It includes the remaining five inserts (6-10) for room types with double beds. The execution results in the 'Script Output' tab show '1 row inserted.' for each query.

```
-- Insert 6
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Deluxe', 3002, 200, 'Luxurious deluxe room with modern amenities.');
-- Insert 7
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 4001, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 8
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Single Bed', 4002, 100, 'Comfortable room with a single bed, suitable for solo travelers.');
-- Insert 9
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 5001, 150, 'Spacious room with a double bed, perfect for couples.');
-- Insert 10
INSERT INTO RoomType (RoomType, RoomNumber, Cost, Description)
VALUES ('Double Bed', 5002, 150, 'Spacious room with a double bed, perfect for couples.');
```

➤ Packages:

-- Insert 1

INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)

```
VALUES (9831, 'Beach Getaway', 4567, 500, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Enjoy a relaxing beach vacation at our luxurious resort.');
```

-- Insert 2

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (7542, 'Mountain Escape', 7890, 400, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Experience the tranquility of the mountains with our special package.');
```

-- Insert 3

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (6218, 'City Adventure', 8901, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Explore the vibrant city life with our exciting package.');
```

-- Insert 4

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (8379, 'Wine Tasting Tour', 6790, 250, 1, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Indulge in exquisite wines with a guided tour of local vineyards.');
```

-- Insert 5

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (5487, 'Spa Retreat', 9012, 350, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Pamper yourself with rejuvenating spa treatments and relaxation.');
```

-- Insert 6

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (2903, 'Family Fun Package', 6789, 450, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Create lasting memories with our family-friendly activities and amenities.');
```

-- Insert 7

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (6310, 'Romantic Getaway', 5678, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Celebrate love and romance with our intimate getaway package.');
```

-- Insert 8

```
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
```

```
VALUES (4126, 'Adventure Excursion', 1234, 400, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Embark on thrilling outdoor adventures in the heart of nature.');
```

-- Insert 9

```

INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (8765, 'Culinary Experience', 2345, 350, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Savor gourmet delights and culinary masterpieces with our exclusive package.');

-- Insert 10

INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (3591, 'Golf Retreat', 3456, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Hit the greens and enjoy a golfing getaway amidst picturesque landscapes.');

```

```

-- Insert 1
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (9631, 'Beach Getaway', 4567, 500, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Enjoy a relaxing beach vacation at our luxurious resort.');

-- Insert 2
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (7542, 'Mountain Escape', 7890, 400, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Experience the tranquility of the mountains with our special package.');

-- Insert 3
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (6218, 'City Adventure', 8901, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Explore the vibrant city life with our exciting package.');

-- Insert 4
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (8379, 'Wine Tasting Tour', 6790, 250, 1, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Indulge in exquisite wines with a guided tour of local vineyards.');

-- Insert 5
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (5487, 'Spa Retreat', 9012, 350, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Pamper yourself with rejuvenating spa treatments and relaxation.');

```

1 row inserted.
1 row inserted.
1 row inserted.

```

-- Insert 6
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (2903, 'Family Fun Package', 6789, 450, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Create lasting memories with our family-friendly activities and amenities.');

-- Insert 7
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (6310, 'Romantic Getaway', 5678, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Celebrate love and romance with our intimate getaway package.');

-- Insert 8
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (4126, 'Adventure Excursion', 1234, 400, 3, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Embark on thrilling outdoor adventures in the heart of nature.');

-- Insert 9
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (8765, 'Culinary Experience', 2345, 350, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Savor gourmet delights and culinary masterpieces with our exclusive package.');

-- Insert 10
INSERT INTO Packages (ID, Name, Hotel_ID, Cost, Duration, Valid_Till, Description)
VALUES (3591, 'Golf Retreat', 3456, 300, 2, TO_DATE('2024-12-31', 'YYYY-MM-DD'), 'Hit the greens and enjoy a golfing getaway amidst picturesque landscapes.');

```

1 row inserted.
1 row inserted.
1 row inserted.

➤ Promotions:

```

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

```

VALUES (7392, 'Summer Special', 4567, 'Enjoy our exclusive summer offer with discounted rates!', TO_DATE('2024-06-01','yyyy-mm-dd'), TO_DATE('2024-08-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (3157, 'Weekend Getaway', 5678, 'Escape for the weekend and indulge in luxury at unbeatable prices.', TO_DATE('2024-03-01','yyyy-mm-dd'), TO_DATE('2024-03-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (8741, 'Family Fun Package', 6789, 'Treat your family to an unforgettable vacation with our family-friendly package.', TO_DATE('2024-05-15','yyyy-mm-dd'), TO_DATE('2024-09-15','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (1692, 'Honeymoon Bliss', 7890, 'Celebrate your love with our romantic honeymoon package.', TO_DATE('2024-04-01','yyyy-mm-dd'), TO_DATE('2024-12-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (8294, 'Business Traveler Deal', 8901, 'Ideal for business travelers, enjoy exclusive perks during your stay.', TO_DATE('2024-02-15','yyyy-mm-dd'), TO_DATE('2024-12-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (4057, 'Spa Retreat Special', 9012, 'Relax and rejuvenate with our spa retreat special package.', TO_DATE('2024-03-15','yyyy-mm-dd'), TO_DATE('2024-11-30','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (5736, 'Gourmet Experience', 1234, 'Savor culinary delights with our gourmet experience package.', TO_DATE('2024-04-15','yyyy-mm-dd'), TO_DATE('2024-10-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (9823, 'Adventure Seeker Deal', 2345, 'Embark on thrilling adventures with our exclusive package.', TO_DATE('2024-06-01','yyyy-mm-dd'), TO_DATE('2024-08-31','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

VALUES (7421, 'Ski Vacation Package', 3456, 'Hit the slopes and enjoy a winter wonderland getaway.', TO_DATE('2024-06-15','yyyy-mm-dd'), TO_DATE('2024-09-30','yyyy-mm-dd'));

INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)

```
VALUES (6198, 'Last Minute Escape', 3456, 'Book now and enjoy great discounts on last-minute getaways.', TO_DATE('2024-02-27','yyyy-mm-dd'), TO_DATE('2024-06-30','yyyy-mm-dd'));
```

```
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (7392, 'Summer Special', 4567, 'Enjoy our exclusive summer offer with discounted rates!', TO_DATE('2024-06-01','yyyy-mm-dd'), TO_DATE('2024-08-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (3157, 'Weekend Getaway', 5678, 'Escape for the weekend and indulge in luxury at unbeatable prices.', TO_DATE('2024-03-01','yyyy-mm-dd'), TO_DATE('2024-03-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (8741, 'Family Fun Package', 6789, 'Treat your family to an unforgettable vacation with our family-friendly package.', TO_DATE('2024-05-15','yyyy-mm-dd'), TO_DATE('2024-07-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (1692, 'Honeymoon Bliss', 7890, 'Celebrate your love with our romantic honeymoon package.', TO_DATE('2024-04-01','yyyy-mm-dd'), TO_DATE('2024-12-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (8294, 'Business Traveler Deal', 8901, 'Ideal for business travelers, enjoy exclusive perks during your stay.', TO_DATE('2024-02-15','yyyy-mm-dd'), TO_DATE('2024-04-30','yyyy-mm-dd'));
```

Script Output X | Task completed in 0.545 seconds

1 row inserted.

```
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (4057, 'Spa Retreat Special', 9012, 'Relax and rejuvenate with our spa retreat special package.', TO_DATE('2024-03-15','yyyy-mm-dd'), TO_DATE('2024-11-30','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (5736, 'Gourmet Experience', 1234, 'Savor culinary delights with our gourmet experience package.', TO_DATE('2024-04-15','yyyy-mm-dd'), TO_DATE('2024-10-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (9623, 'Adventure Seeker Deal', 2345, 'Embark on thrilling adventures with our exclusive package.', TO_DATE('2024-06-01','yyyy-mm-dd'), TO_DATE('2024-08-31','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (7421, 'Ski Vacation Package', 3456, 'Hit the slopes and enjoy a winter wonderland getaway.', TO_DATE('2024-06-15','yyyy-mm-dd'), TO_DATE('2024-09-30','yyyy-mm-dd'));
INSERT INTO Promotions (ID, Name, Hotel_ID, Details, Start_Date, End_Date)
VALUES (6198, 'Last Minute Escape', 3456, 'Book now and enjoy great discounts on last-minute getaways.', TO_DATE('2024-02-27','yyyy-mm-dd'), TO_DATE('2024-06-30','yyyy-mm-dd'));
```

Script Output X | Task completed in 0.545 seconds

1 row inserted.

➤ Bookings:

-- Insert 1

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10001, '1001,1002', TO_DATE('2024-02-27', 'YYYY-MM-DD'),
TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'),
4567, 1001, 1001, 2001);
```

-- Insert 2

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10002, '2001,2002', TO_DATE('2024-02-27', 'YYYY-MM-DD'),
TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'),
5678, 1002, 2002);
```

```
-- Insert 3
```

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10003, '3001', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-
01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 6789, 1003, 3001,
2003);
```

```
-- Insert 4
```

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10004, '4002', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-
01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 7890, 1004, 4002,
2004);
```

```
-- Insert 5
```

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10005, '5001', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-
01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 8901, 1005, 5001,
2005);
```

```
-- Insert 6
```

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10006, '1001,1002', TO_DATE('2024-02-28', 'YYYY-MM-DD'),
TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'),
4567, 1006, 1001, 2006);
```

```
-- Insert 7
```

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID,
Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10007, '2001', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 5678, 1007, 2001, 2007);
```

-- Insert 8

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10008, '3002', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 6789, 1008, 3002, 2008);
```

-- Insert 9

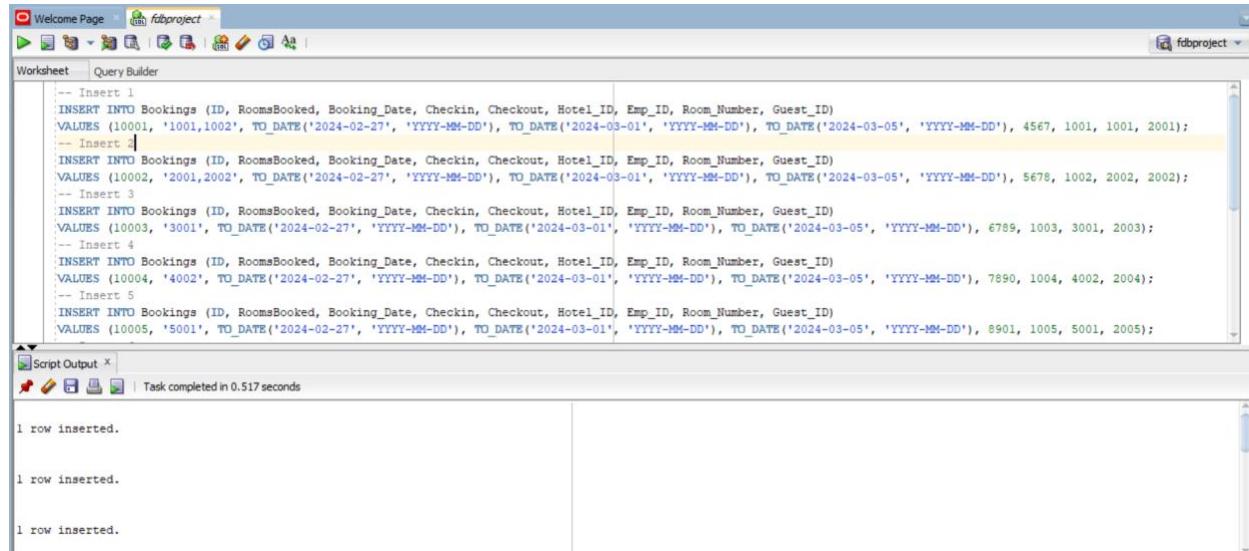
```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10009, '4001', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 7890, 1009, 4001, 2009);
```

-- Insert 10

```
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
```

```
VALUES (10010, '5002', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 8901, 1010, 5002, 2010);
```



The screenshot shows the Oracle SQL Developer interface with the 'Worksheet' tab active. The code area contains 10 INSERT statements, each followed by a comment indicating the insert number (e.g., '-- Insert 1' through '-- Insert 10'). The 'Script Output' tab at the bottom displays three rows of output, each showing '1 row inserted.' This indicates that all 10 statements were successfully executed.

```
-- Insert 1
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10001, '1001', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 4567, 1001, 1001, 2001);
-- Insert 2
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10002, '2002', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 5678, 1002, 2002, 2002);
-- Insert 3
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10003, '3001', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 6789, 1003, 3001, 2003);
-- Insert 4
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10004, '4002', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 7890, 1004, 4002, 2004);
-- Insert 5
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10005, '5001', TO_DATE('2024-02-27', 'YYYY-MM-DD'), TO_DATE('2024-03-01', 'YYYY-MM-DD'), TO_DATE('2024-03-05', 'YYYY-MM-DD'), 8901, 1005, 5001, 2005);
```

Script Output x
Task completed in 0.517 seconds

```
1 row inserted.  
1 row inserted.  
1 row inserted.
```

The screenshot shows a database management tool window. The top bar includes tabs for 'Welcome Page' and 'fdbsproject'. The main area has tabs for 'Worksheet' and 'Query Builder', with 'Worksheet' selected. The 'Worksheet' tab contains several SQL INSERT statements for the 'Bookings' table, each followed by a comment indicating the insert number (e.g., '-- Insert 6'). The 'Script Output' tab below shows the results of the execution: '1 row inserted.' repeated three times.

```

-- Insert 6
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10006, '1001,1002', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 4567, 1006, 1001, 2006);
-- Insert 7
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10007, '2001', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 5678, 1007, 2001, 2007);
-- Insert 8
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10008, '3002', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 6789, 1008, 3002, 2008);
-- Insert 9
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10009, '4001', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 7890, 1009, 4001, 2009);
-- Insert 10
INSERT INTO Bookings (ID, RoomsBooked, Booking_Date, Checkin, Checkout, Hotel_ID, Emp_ID, Room_Number, Guest_ID)
VALUES (10010, '5002', TO_DATE('2024-02-28', 'YYYY-MM-DD'), TO_DATE('2024-03-02', 'YYYY-MM-DD'), TO_DATE('2024-03-06', 'YYYY-MM-DD'), 8901, 1010, 5002, 2010);

1 row inserted.

1 row inserted.

1 row inserted.

```

➤ Payments:

-- Insert 1

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (12345, 2001, 10001, 50001);

-- Insert 2

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (23456, 2002, 10002, 50002);

-- Insert 3

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (34567, 2003, 10003, 50003);

-- Insert 4

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (45678, 2004, 10004, 50004);

-- Insert 5

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (56789, 2005, 10005, 50005);

-- Insert 6

INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (67890, 2006, 10006, 50006);

-- Insert 7

```
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (78901, 2007, 10007, 50007);
```

-- Insert 8

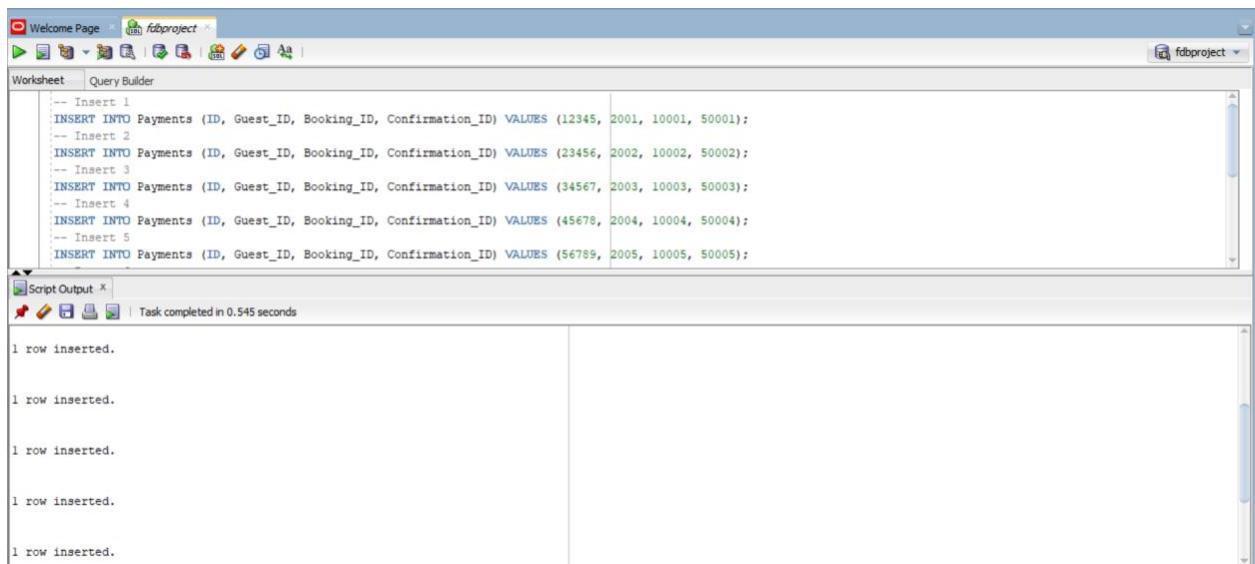
```
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (89012, 2008, 10008, 50008);
```

-- Insert 9

```
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (90123, 2009, 10009, 50009);
```

-- Insert 10

```
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (12340, 2010, 10010, 50010);
```



The screenshot shows the Oracle SQL Developer interface. The top navigation bar includes 'Welcome Page', 'fdoproject', and a toolbar with various icons. Below the toolbar is a tab bar with 'Worksheet' and 'Query Builder'. The main area has two panes. The left pane, titled 'Worksheet', contains the following SQL script:

```
-- Insert 1
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (12345, 2001, 10001, 50001);
-- Insert 2
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (23456, 2002, 10002, 50002);
-- Insert 3
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (34567, 2003, 10003, 50003);
-- Insert 4
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (45678, 2004, 10004, 50004);
-- Insert 5
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (56789, 2005, 10005, 50005);
```

The right pane, titled 'Script Output', displays the results of the execution:

```
1 row inserted.
```

A status message at the bottom of the output pane says 'Task completed in 0.545 seconds'.

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, displaying a block of SQL code. The code consists of several 'INSERT INTO' statements for the 'Payments' table, each with specific values for columns like ID, Guest_ID, Booking_ID, and Confirmation_ID. Below the worksheet, the 'Script Output' tab is visible, showing the results of the execution: four rows were inserted successfully, each followed by the message '1 row inserted.'

```

-- Insert 6
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (67890, 2006, 10006, 50006);
-- Insert 7
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (78901, 2007, 10007, 50007);
-- Insert 8
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (89012, 2008, 10008, 50008);
-- Insert 9
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (90123, 2009, 10009, 50009);
-- Insert 10
INSERT INTO Payments (ID, Guest_ID, Booking_ID, Confirmation_ID) VALUES (12340, 2010, 10010, 50010);

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

```

➤ Attendance:

```

insert into Attendance values(1001,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(1002,TO_DATE('2023-02-01','YYYY-MM-DD'),5);
insert into Attendance values(1003,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(1004,TO_DATE('2023-02-01','YYYY-MM-DD'),6);
insert into Attendance values(1005,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(1006,TO_DATE('2023-02-01','YYYY-MM-DD'),5);
insert into Attendance values(1007,TO_DATE('2023-02-01','YYYY-MM-DD'),6);
insert into Attendance values(1008,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(1009,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(1010,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(1001,TO_DATE('2023-02-02','YYYY-MM-DD'),7);
insert into Attendance values(1002,TO_DATE('2023-02-02','YYYY-MM-DD'),6);
insert into Attendance values(1003,TO_DATE('2023-02-02','YYYY-MM-DD'),5);
insert into Attendance values(1002,TO_DATE('2023-08-31','YYYY-MM-DD'),8);
insert into Attendance values(1003,TO_DATE('2023-08-31','YYYY-MM-DD'),5);
insert into Attendance values(1004,TO_DATE('2023-08-31','YYYY-MM-DD'),4);
insert into Attendance values(1005,TO_DATE('2023-08-31','YYYY-MM-DD'),7);
insert into Attendance values(1006,TO_DATE('2023-08-31','YYYY-MM-DD'),5);

```

around 3000 more inserts have been added. Populated few for reference.

The screenshot shows a database interface with a "Worksheet" tab selected. The query builder contains the following SQL code:

```
insert into Attendance values(101,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(102,TO_DATE('2023-02-01','YYYY-MM-DD'),5);
insert into Attendance values(103,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(104,TO_DATE('2023-02-01','YYYY-MM-DD'),6);
insert into Attendance values(105,TO_DATE('2023-02-01','YYYY-MM-DD'),5);
insert into Attendance values(106,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(107,TO_DATE('2023-02-01','YYYY-MM-DD'),6);
insert into Attendance values(108,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(109,TO_DATE('2023-02-01','YYYY-MM-DD'),4);
insert into Attendance values(110,TO_DATE('2023-02-01','YYYY-MM-DD'),8);
insert into Attendance values(101,TO_DATE('2023-02-02','YYYY-MM-DD'),7);
insert into Attendance values(102,TO_DATE('2023-02-02','YYYY-MM-DD'),6);
insert into Attendance values(103,TO_DATE('2023-02-02','YYYY-MM-DD'),5);
insert into Attendance values(104,TO_DATE('2023-02-02','YYYY-MM-DD'),5);
insert into Attendance values(105,TO_DATE('2023-02-02','YYYY-MM-DD'),5);
insert into Attendance values(106,TO_DATE('2023-02-02','YYYY-MM-DD'),5);
insert into Attendance values(107,TO_DATE('2023-02-02','YYYY-MM-DD'),7);
insert into Attendance values(108,TO_DATE('2023-02-02','YYYY-MM-DD'),4);
insert into Attendance values(109,TO_DATE('2023-02-02','YYYY-MM-DD'),8);
```

The "Script Output" pane below shows the results of the execution:

```
1 row inserted.

1 row inserted.
```

The screenshot shows a database interface with a "Worksheet" tab selected. The query builder contains the following SQL code:

```
insert into Attendance values(101,TO_DATE('2023-05-15','YYYY-MM-DD'),4);
insert into Attendance values(102,TO_DATE('2023-05-16','YYYY-MM-DD'),4);
insert into Attendance values(103,TO_DATE('2023-05-16','YYYY-MM-DD'),8);
insert into Attendance values(104,TO_DATE('2023-05-16','YYYY-MM-DD'),4);
insert into Attendance values(105,TO_DATE('2023-05-16','YYYY-MM-DD'),8);
insert into Attendance values(106,TO_DATE('2023-05-16','YYYY-MM-DD'),6);
insert into Attendance values(107,TO_DATE('2023-05-16','YYYY-MM-DD'),8);
insert into Attendance values(108,TO_DATE('2023-05-16','YYYY-MM-DD'),7);
insert into Attendance values(109,TO_DATE('2023-05-16','YYYY-MM-DD'),6);
insert into Attendance values(110,TO_DATE('2023-05-16','YYYY-MM-DD'),4);
insert into Attendance values(101,TO_DATE('2023-05-17','YYYY-MM-DD'),6);
insert into Attendance values(102,TO_DATE('2023-05-17','YYYY-MM-DD'),4);
insert into Attendance values(103,TO_DATE('2023-05-17','YYYY-MM-DD'),5);
insert into Attendance values(104,TO_DATE('2023-05-17','YYYY-MM-DD'),7);
insert into Attendance values(105,TO_DATE('2023-05-17','YYYY-MM-DD'),4);
```

The "Script Output" pane below shows the results of the execution:

```
1 row inserted.

1 row inserted.

1 row inserted.
```

RETRIEVING ALL RECORDS:

Screenshot 1: Retrieving all records from HotelChain

```
select * from HotelChain;
```

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0.047 seconds

ID	NAME	PHONENUMBER	WEBSITE
1	2024 QueHotel	9376452848	www.QueHotelChain.com

Screenshot 2: Retrieving all records from Hotel

```
select * from Hotel;
```

Script Output | Query Result | SQL | All Rows Fetched: 10 in 0.056 seconds

ID	NAME	PHONENUMBER	EMAIL	HOTELTYPE	CITY	STATE	COUNTRY	ADDRESSLINE1	ADDRESSLINE2	ZIPCODE
1	4567 Grand Resort	9876543210	grandresort@quehotelchain.com	Resort	Miami	Florida	USA	123 Ocean Avenue	Suite 1001	33139
2	5678 Cozy Inn	7543209876	cozyinn@quehotelchain.com	Bed and Breakfast	Aspen	Colorado	USA	456 Mountain Road	Apt 202	81611
3	6789 Sunset Lodge	5412369870	sunsetlodge@quehotelchain.com	Hotel	Santa Barbara	California	USA	789 Seaside Boulevard	Unit 500	93101
4	7890 Mountain View Retreat	7896541230	mountainview@quehotelchain.com	Resort	Lake Tahoe	Nevada	USA	321 Pinecrest Drive	Cabin 20	89452
5	8901 City Lights Hotel	1234876509	citylights@quehotelchain.com	Hotel	New York	New York	USA	987 Broadway Street	Floor 30	10001
6	9012 Sunny Beach Resort	4567890123	sunnybeach@quehotelchain.com	Resort	San Diego	California	USA	654 Sandy Lane	Suite 300	92109
7	1234 Mountain Peak Lodge	9870123456	mountainpeak@quehotelchain.com	Hotel	Denver	Colorado	USA	321 Summit Avenue	Chalet 10	80202
8	2345 Riverside Retreat	3698745210	riverside@quehotelchain.com	Resort	Portland	Oregon	USA	456 Riverwalk Drive	Suite 201	97204
9	3456 Harbor View Hotel	6325489701	harborview@quehotelchain.com	Hotel	Seattle	Washington	USA	789 Marina Drive	Pier 15	98101
10	6790 The Vineyard Inn	9854712360	vineyardinn@quehotelchain.com	Bed and Breakfast	Napa Valley	California	USA	101 Grapevine Lane	Vineyard Cottage	94558

Screenshot 3: Retrieving all records from Amenities

```
SELECT * FROM Amenities;
```

Query Result | SQL | All Rows Fetched: 10 in 0.062 seconds

HOTEL_ID	ID	STEAMROOM	AUDITORIUM	PLAYAREA	POOL	MEETINGROOM	PARKING	GAMEROOM	GYM	RESTAURANTS	MEDICALFACILITIES	CHILDDAYCARE
1	4567	11	YES	NO	YES	YES	YES	YES	YES	YES	NO	NO
2	5678	12	NO	YES	NO	NO	YES	YES	NO	YES	NO	NO
3	6789	13	YES	YES	NO	YES	YES	YES	YES	YES	NO	NO
4	7890	14	YES	NO	YES	YES	YES	NO	YES	YES	NO	NO
5	8901	15	NO	YES	YES	NO	YES	YES	NO	YES	YES	NO
6	9012	16	YES	YES	NO	YES	YES	YES	YES	YES	NO	NO
7	1234	17	NO	YES	NO	NO	YES	YES	NO	YES	NO	NO
8	2345	18	YES	YES	YES	YES	YES	YES	YES	YES	NO	NO
9	3456	19	NO	YES	YES	NO	YES	YES	NO	YES	YES	NO
10	6790	20	YES	NO	NO	YES	YES	YES	NO	YES	YES	NO

fdoproject

Worksheet | Query Builder

```
SELECT * FROM Employee;
```

Query Result | All Rows Fetched: 10 in 0.066 seconds

ID	NAME	PHONENUMBER	EMAIL	ADDRESS	DESIGNATION	CRIMINALRECORD	EDUCATION
1	1001 John Smith	1234567890	john.smith@example.com	123 Main St, Miami, FL, USA	Manager	None	Bachelor of Business Administration
2	1002 Alice Johnson	2345678901	alice.johnson@example.com	456 Mountain Rd, Aspen, CO, USA	Front Desk Clerk	None	High School Diploma
3	1003 Michael Williams	3456789012	michael.williams@example.com	789 Seaside Blvd, Santa Barbara, CA, USA	Chef	None	Culinary Arts Degree
4	1004 Emily Brown	4567890123	emily.brown@example.com	321 Pinecrest Dr, Lake Tahoe, NV, USA	Housekeeping Supervisor	None	High School Diploma
5	1005 David Miller	5678901234	david.miller@example.com	987 Broadway St, New York, NY, USA	Concierge	None	Bachelor of Arts in Hotel Management
6	1006 Jessica Davis	6789012345	jessica.davis@example.com	654 Sandy Ln, San Diego, CA, USA	Spa Therapist	None	Certified Massage Therapist
7	1007 Christopher Wilson	7890123456	christopher.wilson@example.com	321 Summit Ave, Denver, CO, USA	Maintenance Technician	None	High School Diploma
8	1008 Sarah Martinez	8901234567	sarah.martinez@example.com	456 Riverwalk Dr, Portland, OR, USA	Event Coordinator	None	Bachelor of Arts in Event Planning
9	1009 Daniel Anderson	9012345678	daniel.anderson@example.com	789 Marina Dr, Seattle, WA, USA	Security Officer	None	High School Diploma
10	1010 Olivia Garcia	1234567850	olivia.garcia@example.com	101 Grapevine Ln, Napa Valley, CA, USA	Wine Sommelier	None	Certified Sommelier

fdoproject

Worksheet | Query Builder

```
SELECT * FROM Employee;
```

Query Result | All Rows Fetched: 10 in 0.066 seconds

DESIGNATION	CRIMINALRECORD	EDUCATION	CERTIFICATIONS	DEPARTMENT	HOTEL_ID	SUPERVISORID
1, USA Manager	None	Bachelor of Business Administration	Management Certification	Management	4567	(null)
2, CO, USA Front Desk Clerk	None	High School Diploma	Hospitality Certification	Front Desk	5678	1001
3, a Barbara, CA, USA Chef	None	Culinary Arts Degree	Food Safety Certification	Kitchen	6789	1001
4, Tahoe, NV, USA Housekeeping Supervisor	None	High School Diploma	Hospitality Certification	Housekeeping	7890	1001
5, ork, NY, USA Concierge	None	Bachelor of Arts in Hospitality Management	Concierge Certification	Concierge	8901	1001
6, o, CA, USA Spa Therapist	None	Certified Massage Therapist	Spa Certification	Spa	9012	1001
7, CO, USA Maintenance Technician	None	High School Diploma	Maintenance Certification	Maintenance	1234	1001
8, land, OR, USA Event Coordinator	None	Bachelor of Arts in Event Management	Event Planning Certification	Events	2345	1001
9, WA, USA Security Officer	None	High School Diploma	Security Certification	Security	3456	1001
10, Valley, CA, USA Wine Sommelier	None	Certified Sommelier	Wine Tasting Certification	Food and Beverage	6790	1001

Welcome Page fdbproject

Worksheet Query Builder

```
SELECT * FROM Rooms;
```

Query Result All Rows Fetched: 10 in 0.058 seconds

	HOTEL_ID	ROOMNUMBER	FLOORNUMBER
1	4567	1001	1
2	4567	1002	1
3	5678	2001	2
4	5678	2002	2
5	6789	3001	3
6	6789	3002	3
7	7890	4001	4
8	7890	4002	4
9	8901	5001	5
10	8901	5002	5

Welcome Page fdbproject

Run Statement (Ctrl+Enter)

```
SELECT * FROM Guest;
```

Query Result | All Rows Fetched: 10 in 0.08 seconds

ID	NAME	PHONENUMBER	ROOMNUMBER	EMAIL	IDPROOF	CARDNUMBER	ADDRESS
1	2001 Alice Johnson	1234567890	1001	alice.johnson@gmail.com	ABC123XYZ	1234567890123456	123 Main St, Miami, FL, USA
2	2002 Bob Smith	2345678901	1002	bob.smith@gmail.com	DEF456WXY	2345678901234567	456 Mountain Rd, Aspen, CO, USA
3	2003 Charlie Brown	3456789012	2001	charlie.brown@gmail.com	GHI789UVW	3456789012345678	789 Seaside Blvd, Santa Barbara, CA, USA
4	2004 David Miller	4567890123	2002	david.miller@gmail.com	JKL012PQR	4567890123456789	321 Pinecrest Dr, Lake Tahoe, NV, USA
5	2005 Emma Davis	5678901234	3001	emma.davis@gmail.com	MNO345STU	5678901234567890	987 Broadway St, New York, NY, USA
6	2006 Frank Wilson	6789012345	3002	frank.wilson@gmail.com	VWX678YZA	6789012345678901	654 Sandy Ln, San Diego, CA, USA
7	2007 Grace Lee	7890123456	4001	grace.lee@gmail.com	BCD901LMN	7890123456789012	321 Summit Ave, Denver, CO, USA
8	2008 Henry Garcia	8901234567	4002	henry.garcia@gmail.com	EFG234OPQ	8901234567890123	456 Riverwalk Dr, Portland, OR, USA
9	2009 Ivy Martinez	9012345678	5001	ivy.martinez@gmail.com	RST567BCD	9012345678901234	789 Marina Dr, Seattle, WA, USA
10	2010 Jack Thompson	1234876509	5002	jack.thompson@gmail.com	YZA678CDE	1234876509876543	101 Grapevine Ln, Napa Valley, CA, USA

Welcome Page fdbproject

Worksheet | Query Builder

```
SELECT * FROM Guest;
```

Query Result | All Rows Fetched: 10 in 0.063 seconds

ID	NAME	PHONENUMBER	ROOMNUMBER	EMAIL	IDPROOF	CARDNUMBER	ADDRESS
1	2001 Alice Johnson	1234567890	1001	alice.johnson@gmail.com	ABC123XYZ	1234567890123456	123 Main St, Miami, FL, USA
2	2002 Bob Smith	2345678901	1002	bob.smith@gmail.com	DEF456WXY	2345678901234567	456 Mountain Rd, Aspen, CO, USA
3	2003 Charlie Brown	3456789012	2001	charlie.brown@gmail.com	GHI789UVW	3456789012345678	789 Seaside Blvd, Santa Barbara, CA, USA
4	2004 David Miller	4567890123	2002	david.miller@gmail.com	JKL012PQR	4567890123456789	321 Pinecrest Dr, Lake Tahoe, NV, USA
5	2005 Emma Davis	5678901234	3001	emma.davis@gmail.com	MNO345STU	5678901234567890	987 Broadway St, New York, NY, USA
6	2006 Frank Wilson	6789012345	3002	frank.wilson@gmail.com	VWX678YZA	6789012345678901	654 Sandy Ln, San Diego, CA, USA
7	2007 Grace Lee	7890123456	4001	grace.lee@gmail.com	BCD901LMN	7890123456789012	321 Summit Ave, Denver, CO, USA
8	2008 Henry Garcia	8901234567	4002	henry.garcia@gmail.com	EFG234OPQ	8901234567890123	456 Riverwalk Dr, Portland, OR, USA
9	2009 Ivy Martinez	9012345678	5001	ivy.martinez@gmail.com	RST567BCD	9012345678901234	789 Marina Dr, Seattle, WA, USA
10	2010 Jack Thompson	1234876509	5002	jack.thompson@gmail.com	YZA678CDE	1234876509876543	101 Grapevine Ln, Napa Valley, CA, USA

Welcome Page fdbproject

Worksheet Query Builder

```
SELECT * FROM EmployeeRating;
```

Query Result All Rows Fetched: 10 in 0.056 seconds

	EMP_ID	RATING	GUEST_ID
1	1001	A	2001
2	1002	B	2002
3	1003	C	2003
4	1004	D	2004
5	1005	E	2005
6	1006	A	2006
7	1007	B	2007
8	1008	C	2008
9	1009	D	2009
10	1010	E	2010

Welcome Page × fabproject ×

Worksheet Query Builder

```
SELECT * FROM HotelService;
```

Query Result ×

All Rows Fetched: 10 in 0.062 seconds

ID	NAME	HOTEL_ID	COST
1	Room Service	4567	20
2	Laundry Service	5678	30
3	Spa Massage	6789	50
4	Airport Shuttle	7890	40
5	Concierge Service	8901	25
6	Fitness Center Access	4567	15
7	Car Rental	5678	60
8	Guided Tour	6789	35
9	Valet Parking	7890	20
10	In-Room Dining	8901	25

Welcome Page fabproject

Worksheet Query Builder

```
SELECT * FROM StarRating;
```

Query Result

All Rows Fetched: 10 in 0.06 seconds

	HOTEL_ID	ID	RATING	GUEST_ID	FEEDBACK
1	4567	91876	A		2001 Excellent service and beautiful accommodations.
2	5678	37594	B		2002 Comfortable stay, friendly staff.
3	6789	56218	C		2003 Average experience, could improve cleanliness.
4	7890	74029	D		2004 Disappointed with room condition and service.
5	8901	12683	E		2005 Terrible experience, would not recommend.
6	4567	58340	A		2006 Amazing amenities and attentive staff.
7	5678	49237	B		2007 Enjoyed the location and atmosphere.
8	6789	86541	C		2008 Fair service but room was noisy.
9	7890	20789	D		2009 Poor communication and cleanliness issues.
10	8901	63452	E		2010 Horrible stay, rude staff and unclean rooms.

Workshop

fdbsqlproject

Worksheet Query Builder

```
SELECT * FROM Maintenance;
```

Query Result

All Rows Fetched: 10 in 0.058 seconds

ID	MAINTENANCE_TYPE	HOTEL_ID	EMP_ID	ROOM_NUMBER
1	1 Plumbing	4567	1007	1001
2	2 Electrical	5678	1008	2002
3	3 HVAC	6789	1009	3001
4	4 Painting	7890	1010	4002
5	5 Carpentry	8901	1007	5001
6	6 Plumbing	4567	1008	1002
7	7 Electrical	5678	1009	2001
8	8 HVAC	6789	1010	3002
9	9 Painting	7890	1007	4001
10	10 Carpentry	8901	1008	5002

Welcome Page > fdbproject

Worksheet | Query Builder

```
SELECT * FROM RoomType;
```

Query Result x

All Rows Fetched: 10 in 0.056 seconds

	ROOMTYPE	ROOMNUMBER	COST	DESCRIPTION
1	Single Bed	1001	100	Comfortable room with a single bed, suitable for solo travelers.
2	Single Bed	1002	100	Comfortable room with a single bed, suitable for solo travelers.
3	Double Bed	2001	150	Spacious room with a double bed, perfect for couples.
4	Double Bed	2002	150	Spacious room with a double bed, perfect for couples.
5	Deluxe	3001	200	Luxurious deluxe room with modern amenities.
6	Deluxe	3002	200	Luxurious deluxe room with modern amenities.
7	Single Bed	4001	100	Comfortable room with a single bed, suitable for solo travelers.
8	Single Bed	4002	100	Comfortable room with a single bed, suitable for solo travelers.
9	Double Bed	5001	150	Spacious room with a double bed, perfect for couples.
10	Double Bed	5002	150	Spacious room with a double bed, perfect for couples.

Welcome Page > fdbproject

Worksheet | Query Builder

```
SELECT * FROM Packages;
```

Query Result x

All Rows Fetched: 10 in 0.055 seconds

	ID	NAME	HOTEL_ID	COST	DURATION	VALID_TILL	DESCRIPTION
1	9831	Beach Getaway	4567	500	3	31-DEC-24	Enjoy a relaxing beach vacation at our luxurious resort.
2	7542	Mountain Escape	7890	400	2	31-DEC-24	Experience the tranquility of the mountains with our special package.
3	6218	City Adventure	8901	300	2	31-DEC-24	Explore the vibrant city life with our exciting package.
4	8379	Wine Tasting Tour	6790	250	1	31-DEC-24	Indulge in exquisite wines with a guided tour of local vineyards.
5	5487	Spa Retreat	9012	350	2	31-DEC-24	Pamper yourself with rejuvenating spa treatments and relaxation.
6	2903	Family Fun Package	6789	450	3	31-DEC-24	Create lasting memories with our family-friendly activities and amenities.
7	6310	Romantic Getaway	5678	300	2	31-DEC-24	Celebrate love and romance with our intimate getaway package.
8	4126	Adventure Excursion	1234	400	3	31-DEC-24	Embark on thrilling outdoor adventures in the heart of nature.
9	8765	Culinary Experience	2345	350	2	31-DEC-24	Savor gourmet delights and culinary masterpieces with our exclusive package.
10	3591	Golf Retreat	3456	300	2	31-DEC-24	Hit the greens and enjoy a golfing getaway amidst picturesque landscapes.

Welcome Page × fdbproject ×

Worksheet Query Builder

```
SELECT * FROM Promotions;
```

Query Result ×

SQL | All Rows Fetched: 10 in 0.082 seconds

ID	NAME	HOTEL_ID	DETAILS	START_DATE	END_DATE
1	7392 Summer Special	4567	Enjoy our exclusive summer offer with discounted rates!	01-JUN-24	31-AUG-24
2	3157 Weekend Getaway	5678	Escape for the weekend and indulge in luxury at unbeatable prices.	01-MAR-24	31-MAR-24
3	8741 Family Fun Package	6789	Treat your family to an unforgettable vacation with our family-friendly package.	15-MAY-24	15-SEP-24
4	1692 Honeymoon Bliss	7890	Celebrate your love with our romantic honeymoon package.	01-APR-24	31-DEC-24
5	8294 Business Traveler Deal	8901	Ideal for business travelers, enjoy exclusive perks during your stay.	15-FEB-24	31-DEC-24
6	4057 Spa Retreat Special	9012	Relax and rejuvenate with our spa retreat special package.	15-MAR-24	30-NOV-24
7	5736 Gourmet Experience	1234	Savor culinary delights with our gourmet experience package.	15-APR-24	31-OCT-24
8	9823 Adventure Seeker Deal	2345	Embark on thrilling adventures with our exclusive package.	01-JUN-24	31-AUG-24
9	7421 Ski Vacation Package	3456	Hit the slopes and enjoy a winter wonderland getaway.	15-JUN-24	30-SEP-24
10	6198 Last Minute Escape	3456	Book now and enjoy great discounts on last-minute getaways.	27-FEB-24	30-JUN-24

Welcome Page × fdbproject ×

Worksheet Query Builder

```
SELECT * FROM Bookings;
```

Query Result ×

SQL | All Rows Fetched: 10 in 0.061 seconds

ID	ROOMSBOOKED	BOOKING_DATE	CHECKIN	CHECKOUT	HOTEL_ID	EMP_ID	ROOM_NUMBER	GUEST_ID
1	10001 1001,1002	27-FEB-24	01-MAR-24	05-MAR-24	4567	1001	1001	2001
2	10002 2001,2002	27-FEB-24	01-MAR-24	05-MAR-24	5678	1002	2002	2002
3	10003 3001	27-FEB-24	01-MAR-24	05-MAR-24	6789	1003	3001	2003
4	10004 4002	27-FEB-24	01-MAR-24	05-MAR-24	7890	1004	4002	2004
5	10005 5001	27-FEB-24	01-MAR-24	05-MAR-24	8901	1005	5001	2005
6	10006 1001,1002	28-FEB-24	02-MAR-24	06-MAR-24	4567	1006	1001	2006
7	10007 2001	28-FEB-24	02-MAR-24	06-MAR-24	5678	1007	2001	2007
8	10008 3002	28-FEB-24	02-MAR-24	06-MAR-24	6789	1008	3002	2008
9	10009 4001	28-FEB-24	02-MAR-24	06-MAR-24	7890	1009	4001	2009
10	10010 5002	28-FEB-24	02-MAR-24	06-MAR-24	8901	1010	5002	2010

Welcome Page fdbproject

Worksheet Query Builder

```
SELECT * FROM Payments;
```

Query Result All Rows Fetched: 10 in 0.099 seconds

	ID	GUEST_ID	BOOKING_ID	CONFIRMATION_ID	
1	12345	2001	10001	50001	
2	23456	2002	10002	50002	
3	34567	2003	10003	50003	
4	45678	2004	10004	50004	
5	56789	2005	10005	50005	
6	67890	2006	10006	50006	
7	78901	2007	10007	50007	
8	89012	2008	10008	50008	
9	90123	2009	10009	50009	
10	12340	2010	10010	50010	

Worksheet | Query Builder

```
Select * from Attendance;
```

Query Result | SQL | Fetched 50 rows in 0.064 seconds

	EMP_ID	ATTEND_DATE	HOURS
1	1006	17-APR-23	4
2	1007	17-APR-23	6
3	1008	17-APR-23	6
4	1009	17-APR-23	4
5	1010	17-APR-23	7
6	1001	18-APR-23	7
7	1002	18-APR-23	5
8	1003	18-APR-23	6
9	1004	18-APR-23	5
10	1005	18-APR-23	8
11	1006	18-APR-23	5
12	1007	18-APR-23	6
13	1008	18-APR-23	4
14	1009	18-APR-23	8
15	1010	18-APR-23	8
16	1001	19-APR-23	4
17	1002	19-APR-23	6
18	1003	19-APR-23	7

Query Result | SQL | Fetched 50 rows in 0.064 seconds

	EMP_ID	ATTEND_DATE	HOURS
19	1004	19-APR-23	6
20	1005	19-APR-23	7
21	1006	19-APR-23	7
22	1007	19-APR-23	5
23	1008	19-APR-23	8
24	1009	19-APR-23	8
25	1010	19-APR-23	8
26	1001	20-APR-23	8
27	1002	20-APR-23	8
28	1003	20-APR-23	5
29	1004	20-APR-23	8
30	1005	20-APR-23	5
31	1006	20-APR-23	8
32	1007	20-APR-23	6
33	1008	20-APR-23	6
34	1009	20-APR-23	8
35	1010	20-APR-23	5
36	1001	21-APR-23	7

Individual Contributions:

For part 3 of the project, I helped with writing the entity transformations. I also assisted in writing the SQL Oracle statements and created tables and inserted values into those tables. I also wrote additional assumptions that we used to create the database for QueHotel.

I also assisted in updating the ER diagram and adding additional entities like Attendance entity, that we previously missed. This entity will be used to calculate the no.of hours of work an employee is assigned.

I also assisted with editing the documentation and making appropriate changes. I also organized timely and productive meetings to discuss the project and update the project wherever necessary.