

**JAVA SWING BASED- IT'S VACAY TIME-TOUR  
MANAGEMENT SYSTEM- SQL CONNECTIVITY  
USING JDBC**

A

*Report*

*Submitted in partial fulfillment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

Bashetti Sankeerthana

<1602-19-737-099>

Under the guidance of Ms B. Leelavathy



Department of Information Technology  
Vasavi College of Engineering (Autonomous)  
(Affiliated to Osmania University)

ITS VACAY TIME TOUR MANAGEMENT SYSTEM

Ibrahimbagh, Hyderabad-31

B.SANKEERTHANA  
1602-19-737-099

## BONAFIDE CERTIFICATE

This is to certify that this project report titled '**It's Vacay Time-Tour Management System**' is a project work of Ms. Bashetti Sankeerthana bearing roll no. 1602-19-737-099 who carried out the project under my supervision in the IV semester for the academic year 2020- 2021.

Signature

Internal Examiner

B.SANKEERTHANA  
1602-19-737-099

Signature

External Examiner

## ABSTRACT

Its Vacay Time is a Tour Planning Management System is an integrated software developed for the Tour Operating Companies . It is also called Travel Technology Solution. This plays a vital role in designing the best and perfect plan trip. This platform helps the customer to access all the details about the vacation location, hotels..etc. The main purpose of this system is that the information about tour will be available for both customer and administrator on their finger tips.

The tourism management system allows the user of the system access all the details such as means of transport, hotels, places, etc. The main purpose is to help tourism companies to manage customer and hotels etc. The system can also be used for both professional and business trips. The proposed system maintains centralized repository to make necessary travel arrangements and to retrieve information easily.

# INTRODUCTION

## REQUIREMENTS

<u>Table name</u>	<u>Attributes</u>
States	state_Id number state_Name varchar(30) package varchar(30) days integer
Places	place_Id number place_Name varchar(20) state_Id number
Hotels	hotel_Id number hotel_Name varchar(40) no_Of_Rooms integer state_Id number
Customers	customer_Id number customer_Name varchar(50) no_Of_People number email_id varchar(30) phone_No number address varchar2(60) payment_Id number
Payments	payment_Id number cost number rating
Selects	selectedYesOrNo char(5) vehicle_Type varchar(10) customer_Id number state_Id number

## **AIM AND PRIORITY OF THE PROJECT**

To create a Java GUI based Tour Management System which takes the values like: customer\_id, customer\_name etc. from the person who selects the state (by state\_id) to which he wants to travel. The information of to which state, place he travels and hotel he resides will be stored in the database. If any new updates occur, also will be updated in the database using JDBC connectivity.

## ARCHITECTURE AND TECHNOLOGY

### Software used:

Java Eclipse, Oracle 11g Database, Java SE version 08, SQL\*Plus.

### Java AWT:

**Javax SWING** is an API to develop GUI or window-based applications in java. Javax Swing components are platform-independent i.e. components are displayed irrespective to the view of operating system. SWING is lightweight i.e. its components are independent of the resources of OS.

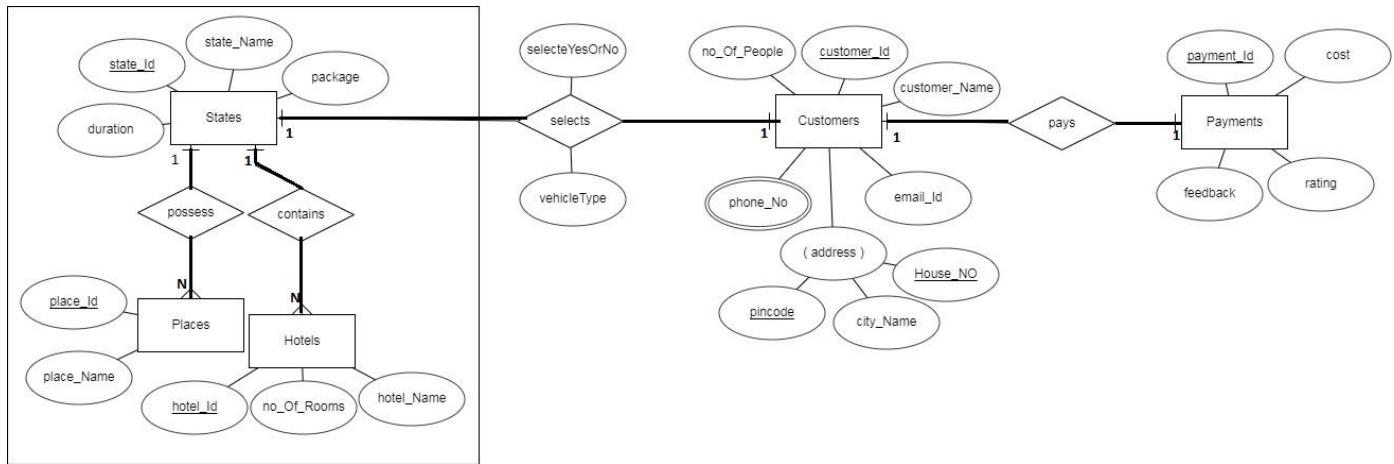
The javax.swing package provides classes for SWING API such as JTextField, JLabel, JTextArea, JMenu, Choice, List ,JMenubar etc.

### SQL:

Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's **Relational** model of database. Today almost all RDBMS (MySql, Oracle, Infomix, Sybase, MS Access) use **SQL** as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

# DESIGN

## Entity relationship diagram



## Database Design:

### DDL Operations

#### States table:

```
SQL> create table States(
 2 state_Id number,
 3 state_Name varchar(30) not null,
 4 package varchar(30),
 5 days integer,
 6 constraint pk_stateID primary key(state_Id));
```

Table created.

```
SQL> desc States;
Name          Null?    Type
STATE_ID      NOT NULL NUMBER
STATE_NAME    NOT NULL VARCHAR2(30)
PACKAGE       VARCHAR2(30)
DAYS          NUMBER(38)
```

#### Places table:

```
SQL> create table Places(
 2 place_Id number,
 3 place_Name varchar(30),
 4 state_Id number not null,
 5 constraint pk_placeID primary key(place_Id),
 6 constraint fk_stateId_Places foreign key(state_Id) references States on delete cascade);
```

Table created.

```
SQL> desc Places;
Name          Null?    Type
PLACE_ID      NOT NULL NUMBER
PLACE_NAME    VARCHAR2(30)
STATE_ID      NOT NULL NUMBER
```

#### Hotels table:

```
SQL> create table Hotels(
 2 hotel_Id number,
 3 hotel_Name varchar(40),
 4 no_Of_Rooms_Available integer,
 5 state_Id number not null,
 6 constraint pk_hotelID primary key(hotel_Id),
 7 constraint fk_stateId_Hotels foreign key(state_Id) references States on delete cascade);
```

Table created.

```
SQL> desc Hotels;
Name          Null?    Type
HOTEL_ID      NOT NULL NUMBER
HOTEL_NAME    VARCHAR2(40)
NO_OF_ROOMS_AVAILABLE NUMBER(38)
STATE_ID      NOT NULL NUMBER
```

## Payments table:

```
SQL> create table Payments(
  2 payment_Id number,
  3 cost integer,
  4 rating number,
  5 customer_Id number not null,
  6 constraint pk_paymentId primary key(payment_Id),
  7 constraint fk_customerId foreign key(customer_Id) references Customers on delete cascade);

Table created.

SQL> desc Payments;
Name          Null?    Type
-----        -----   -----
PAYMENT_ID    NOT NULL NUMBER
COST          NUMBER(38)
RATING         NUMBER
CUSTOMER_ID   NOT NULL NUMBER
```

## Customers table:

```
SQL> create table Customers(
  2 customer_Id number,
  3 customer_Name varchar(20),
  4 no_Of_People integer,
  5 emailId varchar(30),
  6 phone_No number,
  7 address varchar(50),
  8 constraint pk_customerId primary key(customer_Id));

Table created.
```

```
SQL> desc Customers;
Name          Null?    Type
-----        -----   -----
CUSTOMER_ID   NOT NULL NUMBER
CUSTOMER_NAME VARCHAR2(20)
NO_OF_PEOPLE  NUMBER(38)
EMAILID       VARCHAR2(30)
PHONE_NO      NUMBER
ADDRESS        VARCHAR2(50)
```

## Selects Table

```
SQL> create table selects(
  2 selectedYesOrNo char(5),
  3 vehicle_Type varchar(20),
  4 customer_Id number not null,
  5 state_Id number not null,
  6 constraint fk_customerId_selects foreign key(customer_Id) references Customers on delete cascade,
  7 constraint fk_stateId_selects foreign key(state_Id) references States on delete cascade);

Table created.

SQL> desc selects;
Name          Null?    Type
-----        -----   -----
SELECTEDYESORNO  CHAR(5)
VEHICLE_TYPE    VARCHAR2(20)
CUSTOMER_ID     NOT NULL NUMBER
STATE_ID        NOT NULL NUMBER
```

## DML Operations

States table:

```

SQL> insert into States values(&state_Id,'&state_Name','&package',&days);
Enter value for state_id: 11
Enter value for state_name: Goa
Enter value for package: 12000 per person
Enter value for days: 4
old  1: insert into States values(&state_Id,'&state_Name','&package',&days)
new  1: insert into States values(11,'Goa','12000 per person',4)

1 row created.

SQL> /
Enter value for state_id: 12
Enter value for state_name: Kerala
Enter value for package: 13500 per person
Enter value for days: 4
old  1: insert into States values(&state_Id,'&state_Name','&package',&days)
new  1: insert into States values(12,'Kerala','13500 per person',4)

1 row created.

SQL> /
Enter value for state_id: 13
Enter value for state_name: Telangana
Enter value for package: 17500 per person
Enter value for days: 5
old  1: insert into States values(&state_Id,'&state_Name','&package',&days)
new  1: insert into States values(13,'Telangana','17500 per person',5)

1 row created.

SQL> /
Enter value for state_id: 14
Enter value for state_name: Maharashtra
Enter value for package: 12800 per person
Enter value for days: 3
old  1: insert into States values(&state_Id,'&state_Name','&package',&days)
new  1: insert into States values(14,'Maharashtra','12800 per person',3)

1 row created.

SQL> insert into States values(15,'Uttar Pradesh','14,800 per person',4);
1 row created.

SQL> select * from States;
  STATE_ID STATE_NAME          PACKAGE
  -----  -----
  DAYS
  -----
    11  Goa           12000 per person
      4
    12  Kerala        13500 per person
      4
    13  Telangana     17500 per person
      5

  STATE_ID STATE_NAME          PACKAGE
  -----  -----
  DAYS
  -----
    14  Maharashtra   12800 per person
      3
    15  Uttar Pradesh 14,800 per person
      4

```

## Places Table

```

SQL> insert into Places values(&place_Id,'&place_Name',&state_Id);
Enter value for place_id: 1111
Enter value for place_name: Fort Aguada
Enter value for state_id: 11
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1111,'Fort Aguada',11)

1 row created.

SQL> /
Enter value for place_id: 1112
Enter value for place_name: Bogdeshwara Temple
Enter value for state_id: 11
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1112,'Bogdeshwara Temple',11)

1 row created.

SQL> /
Enter value for place_id: 1113
Enter value for place_name: Vagator Beach
Enter value for state_id: 11
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1113,'Vagator Beach',11)

1 row created.

SQL> /
Enter value for place_id: 1114
Enter value for place_name: Anjuna Flea Market
Enter value for state_id: 11
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1114,'Anjuna Flea Market',11)

1 row created.

SQL> select * from places where state_Id=11;

```

PLACE_ID	PLACE_NAME	STATE_ID
1111	Fort Aguada	11
1112	Bogdeshwara Temple	11
1113	Vagator Beach	11
1114	Anjuna Flea Market	11

```

SQL> insert into Places values(&place_Id,'&place_Name',&state_Id);
Enter value for place_id: 1211
Enter value for place_name: Alleppey
Enter value for state_id: 12
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1211,'Alleppey',12)

1 row created.

SQL> /
Enter value for place_id: 1212
Enter value for place_name: Kochi
Enter value for state_id: 12
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1212,'Kochi',12)

1 row created.

SQL> /
Enter value for place_id: 1213
Enter value for place_name: Munnar
Enter value for state_id: 12
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1213,'Munnar',12)

1 row created.

SQL> /
Enter value for place_id: 1214
Enter value for place_name: Thiruvananthapuram
Enter value for state_id: 12
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1214,'Thiruvananthapuram',12)

1 row created.

SQL> select * from Places where state_Id=12;


| PLACE_ID | PLACE_NAME         | STATE_ID |
|----------|--------------------|----------|
| 1211     | Alleppey           | 12       |
| 1212     | Kochi              | 12       |
| 1213     | Munnar             | 12       |
| 1214     | Thiruvananthapuram | 12       |


Enter value for place_name: Charminar
Enter value for state_id: 13
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1312,'Charminar',13)

1 row created.

SQL> /
Enter value for place_id: 1313
Enter value for place_name: Ramoji Film City
Enter value for state_id: 13
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1313,'Ramoji Film City',13)

1 row created.

SQL> /
Enter value for place_id: 1314
Enter value for place_name: Wonderla
Enter value for state_id: 13
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1314,'Wonderla',13)

1 row created.

```

```
SQL> select * from Places where state_Id=13;
PLACE_ID PLACE_NAME STATE_ID
----- -----
1311 Ananthagiri 13
1312 Charminar 13
1313 Ramoji Film City 13
1314 Wonderla 13

SQL> select * from Places where state_Id=14;
PLACE_ID PLACE_NAME STATE_ID
----- -----
1411 Mumbai 14
1412 Pune 14
1413 Nagpur 14
1414 Mahabaleshwar 14

SQL> select * from Places where state_Id=14;
PLACE_ID PLACE_NAME STATE_ID
----- -----
1411 Mumbai 14
1412 Pune 14
1413 Nagpur 14
1414 Mahabaleshwar 14
```

```
SQL> insert into Places values(1115,'Reis Magos Fort',11);
1 row created.

SQL> insert into Places values(1215,'Periyar National Park',12);
1 row created.

SQL> insert into Places values(1315,'Birla Mandir, Tank Bund',13);
1 row created.

SQL> insert into Places values(1415,'Ellora Caves',14);
1 row created.

SQL> insert into Places values(&place_Id,'&place_Name',&state_Id);
Enter value for place_id: 1511
Enter value for place_name: Agra
Enter value for state_id: 15
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1511,'Agra',15)

1 row created.

SQL> /
Enter value for place_id: 1512
Enter value for place_name: Fatehpur Sikri
Enter value for state_id: 15
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1512,'Fatehpur Sikri',15)

1 row created.

SQL> /
Enter value for place_id: 1513
Enter value for place_name: Soor Sarovar
Enter value for state_id: 15
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1513,'Soor Sarovar',15)

1 row created.

SQL> /
Enter value for place_id: 1514
Enter value for place_name: Dayal Bagh
Enter value for state_id: 15
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1514,'Dayal Bagh',15)

1 row created.

SQL> /
Enter value for place_id: 1515
Enter value for place_name: Sikandra
Enter value for state_id: 15
old  1: insert into Places values(&place_Id,'&place_Name',&state_Id)
new  1: insert into Places values(1515,'Sikandra',15)

1 row created.
```

## Hotels Table

```
SQL> insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id);
Enter value for hotel_id: 111
Enter value for hotel_name: Double Tree By Hilton
Enter value for no_of_rooms_available: 100
Enter value for state_id: 11
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(111,'Double Tree By Hilton',100,11)

1 row created.

SQL> /
Enter value for hotel_id: 112
Enter value for hotel_name: Grand Hyatt Goa
Enter value for no_of_rooms_available: 100
Enter value for state_id: 11
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(112,'Grand Hyatt Goa',100,11)

1 row created.

SQL> /
Enter value for hotel_id: 113
Enter value for hotel_name: Serene Waters Resort
Enter value for no_of_rooms_available: 100
Enter value for state_id: 11
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(113,'Serene Waters Resort',100,11)

1 row created.

SQL> /
Enter value for hotel_id: 114
Enter value for hotel_name: Shoolin Homestay
Enter value for no_of_rooms_available: 100
Enter value for state_id: 11
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(114,'Shoolin Homestay',100,11)

1 row created.

SQL> /
Enter value for hotel_id: 115
Enter value for hotel_name: The Green Boutique
Enter value for no_of_rooms_available: 100
Enter value for state_id: 11
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(115,'The Green Boutique',100,11)

1 row created.
```

## Its Vacay Time-Tour Management System

```
SQL> select * from Hotels;

```

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
111	Double Tree By Hilton 11	100
112	Grand Hyatt Goa 11	100
113	Serene Waters Resort 11	100

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
114	Shoolin Homestay 11	100
115	The Green Boutique 11	100

```
SQL> insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id);
Enter value for hotel_id: 121
Enter value for hotel_name: Emarald Resorts
Enter value for no_of_rooms_available: 100
Enter value for state_id: 12
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(121,'Emarald Resorts',100,12)

1 row created.

SQL> /
Enter value for hotel_id: 122
Enter value for hotel_name: Hyatt Regency Thrissur
Enter value for no_of_rooms_available: 100
Enter value for state_id: 12
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(122,'Hyatt Regency Thrissur',100,12)

1 row created.

SQL> /
Enter value for hotel_id: 123
Enter value for hotel_name: WelcomeHotel Coimbatore
Enter value for no_of_rooms_available: 100
Enter value for state_id: 12
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(123,'WelcomeHotel Coimbatore',100,12)

1 row created.

SQL> /
Enter value for hotel_id: 124
Enter value for hotel_name: Courtyard By Marroioott
Enter value for no_of_rooms_available: 100
Enter value for state_id: 12
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(124,'Courtyard By Marroioott',100,12)

1 row created.

SQL> /
Enter value for hotel_id: 125
Enter value for hotel_name: The Raviz Kadavu
Enter value for no_of_rooms_available: 100
Enter value for state_id: 12
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(125,'The Raviz Kadavu',100,12)

1 row created.
```

## Its Vacay Time-Tour Management System

```

SQL> select * from Hotels where state_Id=12;
      HOTEL_ID HOTEL_NAME                               NO_OF_ROOMS_AVAILABLE
      STATE_ID
      121 Emerald Resorts                           100
      12
      122 Hyatt Regency Thrissur                   100
      12
      123 WelcomeHotel Coimbatore                  100
      12

      HOTEL_ID HOTEL_NAME                               NO_OF_ROOMS_AVAILABLE
      STATE_ID
      124 Courtyard By Marriot                     100
      12
      125 The Raviz Kadaavu                      100
      12

SQL> insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id);
Enter value for hotel_id: 131
Enter value for hotel_name: Marigold Hotel
Enter value for no_of_room_available: 100
Enter value for state_id: 13
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id)
new   1: insert into Hotels values(131,'Marigold Hotel',100,13)

1 row created.

SQL> /
Enter value for hotel_id: 132
Enter value for hotel_name: ITC Kohenur
Enter value for no_of_room_available: 100
Enter value for state_id: 13
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id)
new   1: insert into Hotels values(132,'ITC Kohenur',100,13)

1 row created.

SQL> /
Enter value for hotel_id: 133
Enter value for hotel_name: The Golconda Hotel
Enter value for no_of_room_available: 100
Enter value for state_id: 13
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id)
new   1: insert into Hotels values(133,'The Golconda Hotel',100,13)

1 row created.

SQL> /
Enter value for hotel_id: 134
Enter value for hotel_name: Pragathi Resorts
Enter value for no_of_room_available: 100
Enter value for state_id: 13
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id)
new   1: insert into Hotels values(134,'Pragathi Resorts',100,13)

1 row created.

SQL> /
Enter value for hotel_id: 135
Enter value for hotel_name: Leonia Holistic Destination
Enter value for no_of_room_available: 100
Enter value for state_id: 13
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Room_Available,&state_Id)
new   1: insert into Hotels values(135,'Leonia Holistic Destination',100,13)

1 row created.

```

## Its Vacay Time-Tour Management System

```
SQL> select * from Hotels where state_id=13;

```

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
131	Marigold Hotel	100
13		
132	ITC Kohenur	100
13		
133	The Golconda Hotel	100
13		

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
134	Pragathi Resorts	100
13		
135	Leonia Holistic Destination	100
13		

```
SQL> insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id);
Enter value for hotel_id: 141
Enter value for hotel_name: The Taj Mahal Palace
Enter value for no_of_rooms_available: 100
Enter value for state_id: 14
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(141,'The Taj Mahal Palace',100,14)

1 row created.

SQL> /
Enter value for hotel_id: 142
Enter value for hotel_name: Della Resorts
Enter value for no_of_rooms_available: 100
Enter value for state_id: 14
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(142,'Della Resorts',100,14)

1 row created.

SQL> /
Enter value for hotel_id: 143
Enter value for hotel_name: Riva Beach Resorts
Enter value for no_of_rooms_available: 100
Enter value for state_id: 14
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(143,'Riva Beach Resorts',100,14)

1 row created.

SQL> /
Enter value for hotel_id: 144
Enter value for hotel_name: Balaji Sarovar Premeiere
Enter value for no_of_rooms_available: 100
Enter value for state_id: 14
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(144,'Balaji Sarovar Premeiere',100,14)

1 row created.

SQL> /
Enter value for hotel_id: 145
Enter value for hotel_name: taj Holiday Village Resort
Enter value for no_of_rooms_available: 100
Enter value for state_id: 14
old   1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new   1: insert into Hotels values(145,'taj Holiday Village Resort',100,14)

1 row created.
```

## Its Vacay Time-Tour Management System

```
SQL> select * from Hotels where state_Id=14;

```

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
141	The Taj Mahal Palace	100
14		
142	Della Resorts	100
14		
143	Riva Beach Resorts	100
14		

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
STATE_ID		
144	Balaji Sarovar Premeiere	100
14		
145	taj Holiday Village Resort	100
14		

```
SQL> insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id);
Enter value for hotel_id: 151
Enter value for hotel_name: TajMahal Lucknow
Enter value for no_of_rooms_available: 100
Enter value for state_id: 15
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(151,'TajMahal Lucknow',100,15)

1 row created.

SQL> /
Enter value for hotel_id: 152
Enter value for hotel_name: Hyatt Regency Lucknow
Enter value for no_of_rooms_available: 100
Enter value for state_id: 15
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(152,'Hyatt Regency Lucknow',100,15)

1 row created.

SQL> /
Enter value for hotel_id: 153
Enter value for hotel_name: Crowne Plaza Gurgaon
Enter value for no_of_rooms_available: 100
Enter value for state_id: 15
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(153,'Crowne Plaza Gurgaon',100,15)

1 row created.

SQL> /
Enter value for hotel_id: 154
Enter value for hotel_name: Vijan Mahal
Enter value for no_of_rooms_available: 100
Enter value for state_id: 15
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(154,'Vijan Mahal',100,15)

1 row created.

SQL> /
Enter value for hotel_id: 155
Enter value for hotel_name: The Landmark Towers
Enter value for no_of_rooms_available: 100
Enter value for state_id: 15
old  1: insert into Hotels values(&hotel_Id,'&hotel_Name',&no_of_Rooms_Available,&state_Id)
new  1: insert into Hotels values(155,'The Landmark Towers',100,15)

1 row created.
```

SQL> select * from hotels where state_id=15;			
HOTEL_ID	HOTEL_NAME	STATE_ID	NO_OF_ROOMS_AVAILABLE
151	TajMahal Lucknow	15	100
153	Crowne Plaza Gurgaon	15	100
152	Hyatt Regency Lucknow	15	100
HOTEL_ID	HOTEL_NAME	STATE_ID	NO_OF_ROOMS_AVAILABLE
154	Vijan Mahal	15	100
155	The Landmark Towers	15	100

## Customers Table

```
SQL> insert into Customers values(&customer_Id,'&customer_Name',&no_of_People,'&emailId','&phone_No','&address');
Enter value for customer_id: 0001
Enter value for customer_name: Raman Bhalla
Enter value for no_of_people: 4
Enter value for emailId: raman@gmail.com
Enter value for phone_no: 9246990011
Enter value for address: FlatNo 109,Prakruthi Apartments,Hyderabad
old  1: insert into Customers values(&customer_Id,'&customer_Name',&no_of_People,'&emailId','&phone_No','&address')
new  1: insert into Customers values(0001,'Raman Bhalla',4,'raman@gmail.com',9246990011,'FlatNo 109,Prakruthi Apartments,Hyderabad')

1 row created.

SQL> /
Enter value for customer_id: 0002
Enter value for customer_name: Ishitha Iyer
Enter value for no_of_people: 10
Enter value for emailId: ishitha@yahoo.com
Enter value for phone_no: 9394959611
Enter value for address: FlatNo 111,Tulasi Nivas,Mumbai
old  1: insert into Customers values(&customer_Id,'&customer_Name',&no_of_People,'&emailId','&phone_No','&address')
new  1: insert into Customers values(0002,'Ishitha Iyer',10,'ishitha@yahoo.com',9394959611,'FlatNo 111,Tulasi Nivas,Mumbai')

1 row created.

SQL> /
Enter value for customer_id: 0003
Enter value for customer_name: Akshara Shetty
Enter value for no_of_people: 5
Enter value for emailId: akshara@yahoo.com
Enter value for phone_no: 9146738910
Enter value for address: Flat No 382,Chaitanya Gateway,Banglore
old  1: insert into Customers values(&customer_Id,'&customer_Name',&no_of_People,'&emailId','&phone_No','&address')
new  1: insert into Customers values(0003,'Akshara Shetty',5,'akshara@yahoo.com',9146738910,'Flat No 382,Chaitanya Gateway,Banglore')

1 row created.

SQL> /
Enter value for customer_id: 0004
Enter value for customer_name: Naithik Naidu
Enter value for no_of_people: 2
Enter value for emailId: naidu@gmail.com
Enter value for phone_no: 9849556688
Enter value for address: FlatNo 285,Krishna Apartments,Chennai
old  1: insert into Customers values(&customer_Id,'&customer_Name',&no_of_People,'&emailId','&phone_No','&address')
new  1: insert into Customers values(0004,'Naithik Naidu',2,'naidu@gmail.com',9849556688,'FlatNo 285,Krishna Apartments,Chennai')

1 row created.
```

```
SQL> select * from Customers;
CUSTOMER_ID CUSTOMER_NAME      NO_OF_PEOPLE EMAILID
PHONE_NO ADDRESS
----- -----
1 Raman Bhalla           4 raman@gmail.com
9246990011 FlatNo 109,Prakruthi Apartments,Hyderabad
2 Ishitha Iyer            10 ishitha@yahoo.com
9394959611 FlatNo 111,Tulasi Nivas,Mumbai
3 Akshara Shetty          5 akshara@yahoo.com
9146738910 Flat No 382,Chaitanya Gateway,Banglore

CUSTOMER_ID CUSTOMER_NAME      NO_OF_PEOPLE EMAILID
PHONE_NO ADDRESS
----- -----
4 Naithik Naidu           2 naidu@gmail.com
9849556688 FlatNo 285,Krishna Apartments,Chennai

SQL> insert into Customers values(0005,'Ameya Shetty',3,'ameya@yahoo.com',9294930114,'FlatNo 1011, Sri Lakshmi Ventures, Raipur');
1 row created.

SQL> select * from Customers;
CUSTOMER_ID CUSTOMER_NAME      NO_OF_PEOPLE EMAILID
PHONE_NO ADDRESS
----- -----
1 Raman Bhalla           4 raman@gmail.com
9246990011 FlatNo 109,Prakruthi Apartments,Hyderabad
2 Ishitha Iyer            10 ishitha@yahoo.com
9394959611 FlatNo 111,Tulasi Nivas,Mumbai
3 Akshara Shetty          5 akshara@yahoo.com
9146738910 Flat No 382,Chaitanya Gateway,Banglore

CUSTOMER_ID CUSTOMER_NAME      NO_OF_PEOPLE EMAILID
PHONE_NO ADDRESS
----- -----
4 Naithik Naidu           2 naidu@gmail.com
9849556688 FlatNo 285,Krishna Apartments,Chennai
5 Ameya Shetty             3 ameya@yahoo.com
9294930114 FlatNo 1011, Sri Lakshmi Ventures, Raipur
```

## Payments Table

```
SQL> insert into Payments values(&payment_Id,&cost,&rating,&customer_Id);
Enter value for payment_id: 2001
Enter value for cost: 48000
Enter value for rating: 4
Enter value for customer_id: 0001
old  1: insert into Payments values(&payment_Id,&cost,&rating,&customer_Id)
new  1: insert into Payments values(2001,48000,4,0001)

1 row created.
```

## Its Vacay Time-Tour Management System

```
SQL> /
Enter value for payment_id: 2003
Enter value for cost: 67500
Enter value for rating: 4
Enter value for customer_id: 0003
old  1: insert into Payments values(&payment_Id,&cost,&rating,&customer_Id)
new  1: insert into Payments values(2003,67500,4,0003)

1 row created.

SQL> /
Enter value for payment_id: 2004
Enter value for cost: 35000
Enter value for rating: 4
Enter value for customer_id: 0004
old  1: insert into Payments values(&payment_Id,&cost,&rating,&customer_Id)
new  1: insert into Payments values(2004,35000,4,0004)

1 row created.

SQL> insert into Payments values(2002,135000,5,0002);

1 row created.

SQL> select * from Payments;

PAYMENT_ID      COST      RATING CUSTOMER_ID
-----  -----
 2001        48000          4          1
 2003        67500          4          3
 2004        35000          4          4
 2002       135000          5          2

SQL> insert into Payments values(2005,0,0,0005);

1 row created.

SQL> select * from Payments;

PAYMENT_ID      COST      RATING CUSTOMER_ID
-----  -----
 2001        48000          4          1
 2003        67500          4          3
 2004        35000          4          4
 2002       135000          5          2
 2005          0            0          5
```

Selects Table

## Its Vacay Time-Tour Management System

```

SQL> insert into selects values('&selectYesOrNo','&vehicle_Type',&customer_Id,&state_Id);
Enter value for selectyesorno: yes
Enter value for vehicle_type: car
Enter value for customer_id: 0001
Enter value for state_id: 11
old  1: insert into selects values('&selectYesOrNo','&vehicle_Type',&customer_Id,&state_Id)
new  1: insert into selects values('yes','car',0001,11)

1 row created.

SQL> /
Enter value for selectyesorno: yes
Enter value for vehicle_type: train
Enter value for customer_id: 0002
Enter value for state_id: 12
old  1: insert into selects values('&selectYesOrNo','&vehicle_Type',&customer_Id,&state_Id)
new  1: insert into selects values('yes','train',0002,12)

1 row created.

SQL> /
Enter value for selectyesorno: yes
Enter value for vehicle_type: Train
Enter value for customer_id: 0003
Enter value for state_id: 13
old  1: insert into selects values('&selectYesOrNo','&vehicle_Type',&customer_Id,&state_Id)
new  1: insert into selects values('yes','Train',0003,13)

1 row created.

SQL> /
Enter value for selectyesorno: yes
Enter value for vehicle_type: Airline
Enter value for customer_id: 0004
Enter value for state_id: 14
old  1: insert into selects values('&selectYesOrNo','&vehicle_Type',&customer_Id,&state_Id)
new  1: insert into selects values('yes','Airline',0004,14)

1 row created.

SQL> select * from selects;

```

SELEC	VEHICLE_TYPE	CUSTOMER_ID	STATE_ID
yes	car	1	11
yes	train	2	12
yes	Train	3	13
yes	Airline	4	14

```

SQL> insert into selects values('no','','0005,15');

1 row created.

SQL> select * from selects;

```

SELEC	VEHICLE_TYPE	CUSTOMER_ID	STATE_ID
yes	car	1	11
yes	train	2	12
yes	Train	3	13
yes	Airline	4	14
no		5	15

## IMPLEMENTATION

### Front end programs and its connectivity

**Java Database Connectivity (JDBC)** is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

The connection to the database can be performed using Java programming (JDBC API) as:

```

public void connectDatabase()
{
    try
    {
        connection
        DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");
        statement = connection.createStatement();

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        ;
        System.out.println(connectException.getSQLState());
        ;
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}

```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

## States Table:

```

package TMS;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;

public class states
{
    private JPanel pn,pn1,pn2,pn3;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_st_name,JL_st_id,JL_st_package,JL_st_duration;
    private JTextField JTF_st_id,JTF_st_name,JTF_st_package,JTF_st_duration;
    Connection con;
    ResultSet rs;
    Statement stmt;
    private JMenuItem insert1,update1,view1,delete1;
    private List statesList;

    public states(JPanel pn,JFrame jframe,JMenuItem insert1,JMenuItem update1,JMenuItem
view1,JMenuItem delete1)
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectDatabase();

        this.jframe=jframe;
        this.insert1=insert1;
        this.update1=update1;
        this.view1=view1;
        this.delete1=delete1;

        JL_st_id=new JLabel("State_Id:");
        JTF_st_id=new JTextField(10);
        JL_st_name=new JLabel("State Name:");
        JTF_st_name=new JTextField(10);
        JL_st_package=new JLabel("Package:");
        JTF_st_package=new JTextField(10);
        JL_st_duration=new JLabel("Duration:");
        JTF_st_duration=new JTextField(10);

        this.pn=pn;
    }
}

```

## Its Vacay Time-Tour Management System

```
}

public void connectDatabase()
{
    try
    {
        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");

        stmt=con.createStatement();
        stmt.executeUpdate("commit");

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}

private void displaySQLErrors(SQLException e)
{
    JOptionPane.showMessageDialog(pn1,"\\nSQLException: " + e.getMessage() +
        "\\n"+ "SQLState:" + e.getSQLState()+"\\n"+ "VendorError:" + e.getErrorCode() + "\\n");
}

public void loadStates()
{
    try
    {
        statesList=new List();
        statesList.removeAll();
        rs=stmt.executeQuery("select * from States");
        while(rs.next())
        {
            statesList.add(rs.getString("state_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void buildGUI()
{
    insert1.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_st_id.setText(null);
            JTF_st_name.setText(null);
            JTF_st_package.setText(null);
            JTF_st_duration.setText(null);
        }
    });
}
```

```

loadStates();

pn.removeAll();
jframe.invalidate();
jframe.validate();
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_st_id);
pn1.add(JTF_st_id);
pn1.add(JL_st_name);
pn1.add(JTF_st_name);
pn1.add(JL_st_package);
pn1.add(JTF_st_package);
pn1.add(JL_st_duration);
pn1.add(JTF_st_duration);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_insert);
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
statesList=new List(10);
loadStates();
pn2.add(statesList);
pn2.setBounds(200,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO States VALUES(" +
JTF_st_id.getText() + "," +
"""+JTF_st_name.getText() + "," +
"""+JTF_st_package.getText() + "," +
+JTF_st_duration.getText() + ")";
            int i = stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn, "\nInserted"+i+
" rows successfully");
            loadStates();
            System.out.println("Done");
        }
        catch(SQLException e)
    }
})

```

## Its Vacay Time-Tour Management System

```
        {
            displaySQLErrors(e);
        }
    });
}
};

update1.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_modify=new JButton("Modify");

        JTF_st_id.setText(null);
        JTF_st_name.setText(null);
        JTF_st_package.setText(null);
        JTF_st_duration.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_st_id);
        pn1.add(JTF_st_id);
        pn1.add(JL_st_name);
        pn1.add(JTF_st_name);
        pn1.add(JL_st_package);
        pn1.add(JTF_st_package);
        pn1.add(JL_st_duration);
        pn1.add(JTF_st_duration);

        pn3=new JPanel(new FlowLayout());
        pn3.add(JB_modify);
        pn1.setBounds(115,80,300,250);
        pn3.setBounds(200,350,75,35);

        pn2=new JPanel(new FlowLayout());
        statesList=new List(10);
        loadStates();
        pn2.add(statesList);
        pn2.setBounds(200,350,300,180);

        pn.add(pn1);
        pn.add(pn3);
        pn.add(pn2);

        pn.setLayout(new BorderLayout());
        jframe.add(pn);
        jframe.setSize(800,800);
        jframe.validate();

        statesList.addItemListener(new ItemListener() {
            public void itemStateChanged(ItemEvent ievt)
```

```

    {
        try
        {
            rs=stmt.executeQuery("select * from states");
            while (rs.next())
            {
                if
                    (rs.getString("state_id").equals(statesList.getSelectedItem()))
                        break;
                }
                if (!rs.isAfterLast())
                {

                    JTF_st_id.setText(rs.getString("state_Id"));

                    JTF_st_name.setText(rs.getString("state_Name"));

                    JTF_st_package.setText(rs.getString("package"));

                    JTF_st_duration.setText(rs.getString("days"));

                }
            }
            catch (SQLException selectException)
            {
                displaySQLErrors(selectException);
            }
        }
    });
    JB_modify.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            try
            {
                int a=JOptionPane.showConfirmDialog(pn,"Are you
                sure want to update:");
                if(a==JOptionPane.YES_OPTION)
                {
                    String
                    pack=JOptionPane.showInputDialog(pn,"Enter
                    New Package:");
                    JTF_st_package.setText(pack);
                    String query="update states set
                    package='"+pack+"' where
                    state_Id="+JTF_st_id.getText();
                    int i=stmt.executeUpdate(query);
                    JOptionPane.showMessageDialog(pn, "\nUpdated
                    "+i+" rows succesfully");
                    loadStates();
                }
            }
            catch(SQLException e)
            {
                displaySQLErrors(e);
            }
        }
    });
}

```

## Its Vacay Time-Tour Management System

```
        }
    });
});
delete1.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_delete=new JButton("Delete");

        JTF_st_id.setText(null);
        JTF_st_name.setText(null);
        JTF_st_package.setText(null);
        JTF_st_duration.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_st_id);
        pn1.add(JTF_st_id);
        pn1.add(JL_st_name);
        pn1.add(JTF_st_name);
        pn1.add(JL_st_package);
        pn1.add(JTF_st_package);
        pn1.add(JL_st_duration);
        pn1.add(JTF_st_duration);

        pn3=new JPanel(new FlowLayout());
        pn3.add(JB_delete);
        pn1.setBounds(115,80,300,250);
        pn3.setBounds(200,350,75,35);

        pn2=new JPanel(new FlowLayout());
        statesList=new List(10);
        loadStates();
        pn2.add(statesList);
        pn2.setBounds(200,350,300,200);

        pn.add(pn1);
        pn.add(pn3);
        pn.add(pn2);

        pn.setLayout(new BorderLayout());
        jframe.add(pn);
        jframe.setSize(800,800);
        jframe.validate();

        statesList.addItemListener(new ItemListener() {
            public void itemStateChanged(ItemEvent ievt)
            {
                try

```

## Its Vacay Time-Tour Management System

```
{  
    rs=stmt.executeQuery("select * from states");  
    while (rs.next())  
    {  
        if  
(rs.getString("state_id").equals(statesList.getSelectedItem()))  
            break;  
    }  
    if (!rs.isAfterLast())  
    {  
  
        JTF_st_id.setText(rs.getString("state_Id"));  
  
        JTF_st_name.setText(rs.getString("state_Name"));  
  
        JTF_st_package.setText(rs.getString("package"));  
  
        JTF_st_duration.setText(rs.getString("days"));  
  
    }  
}  
catch (SQLException selectException)  
{  
    displaySQLErrors(selectException);  
}  
}  
});  
JB_delete.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent aevt)  
    {  
        try  
        {  
            int a=JOptionPane.showConfirmDialog(pn,"Are you  
sure want to Delete:");  
            if(a==JOptionPane.YES_OPTION)  
            {  
                String query="DELETE FROM STATES WHERE  
state_Id='"+JTF_st_id.getText();  
                int i=stmt.executeUpdate(query);  
                JOptionPane.showMessageDialog(pn, "\nDeleted  
"+i+" rows successfully");  
                loadStates();  
            }  
        }  
        catch(SQLException e)  
        {  
            displaySQLErrors(e);  
        }  
    }  
});  
});  
view1.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent aevt)
```

## Its Vacay Time-Tour Management System

```
{  
    pn.removeAll();  
    jframe.invalidate();  
    jframe.validate();  
    jframe.repaint();  
  
    JLabel view=new JLabel("States View");  
    JB_view=new JButton("View");  
    Font myFont = new Font("Serif",Font.BOLD,50);  
    view.setFont((myFont));  
  
    pn1=new JPanel();  
    pn2=new JPanel();  
    pn1.add(view);  
    pn2.add(JB_view);  
    pn.add(pn1);  
    pn.add(pn2);  
    pn.setBounds(500,800,300,300);  
    pn.setLayout(new FlowLayout());  
  
    jframe.add(pn);  
    jframe.setSize(800,800);  
    jframe.validate();  
  
    JB_view.addActionListener(new ActionListener() {  
        @Override  
        public void actionPerformed(ActionEvent aevt)  
        {  
            JFrame jf=new JFrame("States Details");  
            JTable jt;  
            DefaultTableModel model = new DefaultTableModel();  
            jt = new JTable(model);  
            model.addColumn("State_ID");  
            model.addColumn("State Name");  
            model.addColumn("Package");  
            model.addColumn("Days");  
            try  
            {  
                rs=stmt.executeQuery("select * from states");  
                while(rs.next())  
                {  
                    model.addRow(new Object[]{rs.getInt(1),  
                        rs.getString(2),rs.getString(3),rs.getInt(4)});  
                }  
            }  
            catch(SQLException e)  
            {  
                displaySQLErrors(e);  
            }  
            jt.setEnabled(false);  
            jt.setBounds(30, 40, 300, 300);  
            JScrollPane jsp = new JScrollPane(jt);  
            jf.add(jsp);  
            jf.setSize(800, 400);  
            jf.setVisible(true);  
        }  
    });  
}
```

```
        });
    });
}
}

Places Table
//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;
import java.util.StringTokenizer;
public class places
{
    private JPanel pn1,pn2,pn3,pn;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_pl_name,JL_pl_id,JL_st_id;
    private JTextField JTF_pl_id,JTF_pl_name,JTF_st_id;
    Connection con;
    ResultSet rs;
    Statement stmt;
    private JMenuItem insert2,update2,view2,delete2;
    private List placesList;
    private Choice stateId,placeId;
    public places(JPanel pn,JFrame jframe,JMenuItem insert2,JMenuItem update2,JMenuItem
view2,JMenuItem delete2)
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectDatabase();

        this.jframe=jframe;
        this.insert2=insert2;
        this.update2=update2;
        this.view2=view2;
        this.delete2=delete2;

        JL_st_id=new JLabel("State_Id:");
        JTF_st_id=new JTextField(10);
        JL_pl_id=new JLabel("Place_Id:");
        JTF_pl_id=new JTextField(10);
        JL_pl_name=new JLabel("Place Name:");
        JTF_pl_name=new JTextField(10);
    }
}
```

```

        this.pn=pn;
    }
    public void connectDatabase()
    {
        try
        {
            Connection con=DriverManager.getConnection(
                "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");

            stmt=con.createStatement();
            stmt.executeUpdate("commit");

        }
        catch (SQLException connectException)
        {
            System.out.println(connectException.getMessage());
            System.out.println(connectException.getSQLState());
            System.out.println(connectException.getErrorCode());
            System.exit(1);
        }
    }
    private void displaySQLErrors(SQLException e)
    {
        JOptionPane.showMessageDialog(pn1,"\\nSQLException: " + e.getMessage() +
        "\\n"+ "SQLState:      " + e.getSQLState() + "\\n"+ "VendorError:  " + e.getErrorCode() + "\\n");
    }
    public void loadStatesPlaces()
    {
        try
        {
            placesList=new List();
            placesList.removeAll();
            rs=stmt.executeQuery("select * from Places");
            while(rs.next())
            {
                placesList.add(rs.getString("place_Id"));
            }
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }

    public void loadStates()
    {
        try
        {
            stateId=new Choice();
            stateId.removeAll();
            rs=stmt.executeQuery("select * from States");
            while(rs.next())
            {
                stateId.add(rs.getString("state_Id"));
            }
        }
    }
}

```

## Its Vacay Time-Tour Management System

```
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }

public void loadPlaces()
{
    try
    {
        placeId=new Choice();
        placeId.removeAll();
        rs=stmt.executeQuery("select * from Places");
        while(rs.next())
        {
            placeId.add(rs.getString("place_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void buildGUI()
{
    insert2.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_st_id.setText(null);
            JTF_pl_id.setText(null);
            JTF_pl_name.setText(null);

            loadStates();
            loadPlaces();

            pn.removeAll();
            jframe.invalidate();
            jframe.validate();
            jframe.repaint();

            pn1=new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_st_id);
            pn1.add(JTF_st_id);
            pn1.add(JL_pl_id);
            pn1.add(JTF_pl_id);
            pn1.add(JL_pl_name);
            pn1.add(JTF_pl_name);

            pn3=new JPanel(new FlowLayout());
            pn3.add(JB_insert);
        }
    });
}
```

## Its Vacay Time-Tour Management System

```
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
placesList=new List(10);
loadStatesPlaces();
pn2.add(placesList);
pn2.setBounds(200,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

//placesList.setEnabled(false);
pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO Places VALUES(" +
                JTF_pl_id.getText() + ","
                + "'" +JTF_pl_name.getText() + "' ,"
                +JTF_st_id.getText() + ")";
            int i = stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn,"\\nInserted
"+i+" rows successfully");
            loadStatesPlaces();
            System.out.println("Done");
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }
});
```

```
});
```

```
update2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_modify=new JButton("Modify");

        JTF_st_id.setText(null);
        JTF_pl_id.setText(null);
        JTF_pl_name.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
    }
});
```

## Its Vacay Time-Tour Management System

```
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_st_id);
pn1.add(JTF_st_id);
pn1.add(JL_pl_id);
pn1.add(JTF_pl_id);
pn1.add(JL_pl_name);
pn1.add(JTF_pl_name);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_modify);
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
placesList=new List(10);
loadStatesPlaces();
pn2.add(placesList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

placesList.addItemListener(new ItemListener()
{
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from Places");

            while (rs.next())
            {
                if
(rs.getString("place_Id").equals(placesList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_pl_id.setText(rs.getString("place_Id"));

                JTF_pl_name.setText(rs.getString("place_Name"));

                JTF_st_id.setText(rs.getString("state_Id"));

            }
        }
    }
});
```

```

        catch (SQLException exp)
        {
            displaySQLErrors(exp);
        }
    });

JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try
        {
            int a=JOptionPane.showConfirmDialog(pn,"Are you sure do
you want to update:");
            if(a==JOptionPane.YES_OPTION)
            {
                String
                placeName=JOptionPane.showInputDialog(pn,"Enter
New Place Name:");
                JTF_pl_name.setText(placeName);
                String query="update Places set
place_Name='"+placeName+
"' where place_Id="+JTF_pl_id.getText();
                int i=stmt.executeUpdate(query);
                JOptionPane.showMessageDialog(pn,"\nUpdated "+i+
"rows successfully");
                loadStatesPlaces();
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
    }
});

JB_delete=new JButton("Delete");

JTF_st_id.setText(null);
JTF_pl_id.setText(null);
JTF_pl_name.setText(null);

pn.removeAll();
jframe.invalidate();
jframe.validate();
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_st_id);

```

## Its Vacay Time-Tour Management System

```
pn1.add(JTF_st_id);
pn1.add(JL_pl_id);
pn1.add(JTF_pl_id);
pn1.add(JL_pl_name);
pn1.add(JTF_pl_name);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn3.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
placesList=new List(10);
loadStatesPlaces();
pn2.add(placesList);
pn2.setBounds(250,350,310,190);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

placesList.addItemListener(new ItemListener()
{
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from
Places");
            StringTokenizer st=new
StringTokenizer(placesList.getSelectedItem());
            String q=st.nextToken();

            while (rs.next())
            {

                if
                (rs.getString("place_Id").equals(q))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_pl_id.setText(rs.getString("place_Id"));

                JTF_pl_name.setText(rs.getString("place_Name"));

                JTF_st_id.setText(rs.getString("state_Id"));
            }
        }
    }
});
```

```

        catch (SQLException selectException)
        {
            displaySQLErrors(selectException);
        }
    });
JB_delete.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e)
{
    try
    {
        int a=JOptionPane.showConfirmDialog(pn,"Are you
sure do you want to delete:");
if(a==JOptionPane.YES_OPTION)
{
    StringTokenizer st=new
StringTokenizer(placesList.getSelectedItem(),"->");
    String query="DELETE FROM Places WHERE
place_Id="+st.nextToken();
    int i=stmt.executeUpdate(query);

    JOptionPane.showMessageDialog(pn,"\nDeleted "+i+" rows
successfully");
    loadStatesPlaces();
}
catch(SQLException exp)
{
    displaySQLErrors(exp);
}
    }
});
view2.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent aevt)
{
    pn.removeAll();
    jframe.invalidate();
    jframe.validate();
    jframe.repaint();

    JLabel view=new JLabel("Places View");
    JB_view=new JButton("View");
    Font myFont = new Font("Serif",Font.BOLD,50);
    view.setFont((myFont));

    pn1=new JPanel();
    pn2=new JPanel();
    pn1.add(view);
    pn2.add(JB_view);
    pn.add(pn1);
    pn.add(pn2);
}
});

```

```
pn.setBounds(500,800,300,300);
pn.setLayout(new FlowLayout());

jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_view.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e)
    {
        JFrame jf=new JFrame("Places Details");
        JTable jt;
        DefaultTableModel model = new
        DefaultTableModel();
        jt= new JTable(model);
        model.addColumn("Place Id");
        model.addColumn("Place Name");
        model.addColumn("State Id");

        try
        {
            rs=stmt.executeQuery("select * from
Places");
            while(rs.next())
            {
                model.addRow(new
Object[]{rs.getInt(1),
rs.getString(2),rs.getInt(3)});
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
        jt.setEnabled(false);
        jt.setBounds(30, 40, 180, 150);
        JScrollPane sp = new JScrollPane(jt);
        jf.add(sp);
        jf.setSize(800, 400);
        jf.setVisible(true);
    }
});
```

## Hotels Table

//B.Sankeerthanam 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM

```
package TMS;  
import java.awt.*;  
import java.awt.event.*;  
import javax.swing.*;
```

## Its Vacay Time-Tour Management System

```
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;
import java.util.StringTokenizer;

public class hotels
{
    private JPanel pn,pn1,pn2,pn3;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_htl_name,JL_htl_id,JL_noOfRooms,JL_st_id;
    private JTextField JTF_st_id,JTF_htl_name,JTF_htl_id,JTF_noOfRooms;
    Connection con;
    ResultSet rs;
    Statement stmt;
    private JMenuItem insert3,update3,view3,delete3;
    private List hotelsList;
    private Choice stateId,hotelId;

    public hotels(JPanel pn,JFrame jframe,MenuItem insert3,MenuItem update3,MenuItem
view3,MenuItem delete3)
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectDatabase();

        this.jframe=jframe;
        this.insert3=insert3;
        this.update3=update3;
        this.view3=view3;
        this.delete3=delete3;

        JL_st_id=new JLabel("State_Id:");
        JTF_st_id=new JTextField(10);
        JL_htl_id=new JLabel("Hotel Id:");
        JTF_htl_id=new JTextField(10);
        JL_htl_name=new JLabel("Hotel Name:");
        JTF_htl_name=new JTextField(10);
        JL_noOfRooms=new JLabel("No Of Rooms:");
        JTF_noOfRooms=new JTextField(10);

        this.pn=pn;
    }
    public void connectDatabase()
    {
        try
        {
            Connection con=DriverManager.getConnection(
                "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");
        }
    }
}
```

## Its Vacay Time-Tour Management System

```
        stmt=con.createStatement();
        stmt.executeUpdate("commit");

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}
private void displaySQLErrors(SQLException e)
{
    JOptionPane.showMessageDialog(pn1,"\\nSQLException: " + e.getMessage() +
"\n"+SQLState:      " + e.getSQLState() + "\n"+VendorError:  " + e.getErrorCode() + "\n");
}
public void loadStatesHotels()
{
    try
    {
        hotelsList=new List();
        hotelsList.removeAll();
        rs=stmt.executeQuery("select * from Hotels");
        while(rs.next())
        {
            //hotelsList.add(rs.getString("hotel_Id")+"-
>"+rs.getString("state_Id"));
            hotelsList.add(rs.getString("hotel_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}
public void loadStates()
{
    try
    {
        stateId=new Choice();
        stateId.removeAll();
        rs=stmt.executeQuery("select * from States");
        while(rs.next())
        {
            stateId.add(rs.getString("state_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}
public void loadHotels()
{
```

## Its Vacay Time-Tour Management System

```
try
{
    hotelId=new Choice();
    hotelId.removeAll();
    rs=stmt.executeQuery("select * from Hotels");
    while(rs.next())
    {
        hotelId.add(rs.getString("hotel_Id"));
    }
}
catch(SQLException e)
{
    displaySQLErrors(e);
}

public void buildGUI()
{
    insert3.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_st_id.setText(null);
            JTF_htl_id.setText(null);
            JTF_htl_name.setText(null);
            JTF_noOfRooms.setText(null);

            loadStates();
            loadHotels();

            pn.removeAll();
            jframe.invalidate();
            jframe.validate();
            jframe.repaint();

            pn1=new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_st_id);
            pn1.add(JTF_st_id);
            pn1.add(JL_htl_id);
            pn1.add(JTF_htl_id);
            pn1.add(JL_htl_name);
            pn1.add(JTF_htl_name);
            pn1.add(JL_noOfRooms);
            pn1.add(JTF_noOfRooms);

            pn3=new JPanel(new FlowLayout());
            pn3.add(JB_insert);
            pn1.setBounds(115,80,300,250);
            pn3.setBounds(200,350,75,35);

            pn2=new JPanel(new FlowLayout());
            hotelsList=new List(10);
            loadStatesHotels();
            pn2.add(hotelsList);
        }
    });
}
```

## Its Vacay Time-Tour Management System

```
pn2.setBounds(200,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

//hotelsList.setEnabled(false);
pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO Hotels VALUES("+
                JTF_htl_id.getText() + ", "
                + "" +JTF_htl_name.getText() +",'"
                +JTF_noOfRooms.getText()+"','"
                +""+JTF_st_id.getText() +")";
            int i = stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn, "\nInserted
                "+i+" rows successfully");
            loadStatesHotels();
            System.out.println("Done");
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }
});
});

update3.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_modify=new JButton("Modify");

        JTF_st_id.setText(null);
        JTF_htl_id.setText(null);
        JTF_htl_name.setText(null);
        JTF_noOfRooms.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_st_id);
    }
});
```

## Its Vacay Time-Tour Management System

```
pn1.add(JTF_st_id);
pn1.add(JL_htl_id);
pn1.add(JTF_htl_id);
pn1.add(JL_htl_name);
pn1.add(JTF_htl_name);
pn1.add(JL_noOfRooms);
pn1.add(JTF_noOfRooms);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_modify);
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
hotelsList=new List(10);
loadStatesHotels();
pn2.add(hotelsList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

hotelsList.addItemListener(new ItemListener() {
    @Override
    public void itemStateChanged(ItemEvent ievt)
    {
        try
        {
            rs=stmt.executeQuery("select * from Hotels");
            while (rs.next())
            {
                if
(rs.getString("hotel_id").equals(hotelsList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_htl_id.setText(rs.getString("hotel_Id"));

                JTF_htl_name.setText(rs.getString("hotel_Name"));

                JTF_noOfRooms.setText(rs.getString("no_Of_Rooms_Available"));

                JTF_st_id.setText(rs.getString("state_Id"));

            }
        }
    }
});
```

```

        catch (SQLException selectException)
        {
            displaySQLErrors(selectException);
        }
    });
JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            int a=JOptionPane.showConfirmDialog(pn1,"Are you
sure do you want to update:");
            if(a==JOptionPane.YES_OPTION)
            {
                String
                rooms=JOptionPane.showInputDialog(pn,"Enter
New No Of Rooms:");
                JTF_noOfRooms.setText(rooms);
                String query="update Hotels set
no_of_Rooms_Available='"+rooms+
"' where hotel_Id="+JTF_htl_id.getText();

                int i=stmt.executeUpdate(query);
                JOptionPane.showMessageDialog(pn,"\\nUpdated
"+i+" rows successfully");
                loadStatesHotels();
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
    }
});
delete3.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_delete=new JButton("Delete");

        JTF_st_id.setText(null);
        JTF_htl_id.setText(null);
        JTF_htl_name.setText(null);
        JTF_noOfRooms.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
    }
});

```

## Its Vacay Time-Tour Management System

```
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_st_id);
pn1.add(JTF_st_id);
pn1.add(JL_htl_id);
pn1.add(JTF_htl_id);
pn1.add(JL_htl_name);
pn1.add(JTF_htl_name);
pn1.add(JL_noOfRooms);
pn1.add(JTF_noOfRooms);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115,80,300,250);
pn3.setBounds(180,350,75,35);

pn2=new JPanel(new FlowLayout());
hotelsList=new List(10);
loadStatesHotels();
pn2.add(hotelsList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

hotelsList.addItemListener(new ItemListener() {
    @Override
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from Hotels");
            StringTokenizer st=new StringTokenizer(hotelsList.getSelectedItem());
            //String p=st.nextToken();
            String q=st.nextToken();

            while (rs.next())
            {
                //if (rs.getString("state_Id").equals(p) &&
                if (rs.getString("hotel_Id").equals(q))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_htl_id.setText(rs.getString("hotel_Id"));

                JTF_htl_name.setText(rs.getString("hotel_Name"));
            }
        }
    }
});
```

```

        JTF_noOfRooms.setText(rs.getString("no_Of_Rooms_Available"));

        JTF_st_id.setText(rs.getString("state_Id"));

    }

}

catch (SQLException selectException)
{
    displaySQLErrors(selectException);
}

}

});

JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            int a=JOptionPane.showConfirmDialog(pn,"Are you
sure do you want to delete:");
            if(a==JOptionPane.YES_OPTION)
            {
                StringTokenizer st=new
StringTokenizer(hotelsList.getSelectedItem(),"->");
                String query="DELETE FROM Hotels WHERE
hotel_Id="+st.nextToken();
                int i=stmt.executeUpdate(query);

                JOptionPane.showMessageDialog(pn,"Deleted "+i+" rows
successfully");
                loadStatesHotels();
            }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
    }
});
});

view3.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        JB_view=new JButton("View");
        JLabel view=new JLabel("Hotels View");
        Font myFont = new Font("Serif",Font.BOLD,50);
        view.setFont((myFont));
    }
});

```

```

pn1=new JPanel();
pn2=new JPanel();
pn1.add(view);
pn2.add(JB_view);
pn.add(pn1);
pn.add(pn2);
pn.setLayout(new FlowLayout());

jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_view.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JFrame jf=new JFrame("Hotels Details");
        JTable jt;
        DefaultTableModel model = new DefaultTableModel();

        jt= new JTable(model);
        model.addColumn("Hotel Id");
        model.addColumn("Hotel Name");
        model.addColumn("No Of Rooms");
        model.addColumn("State Id");

        try
        {
            rs=stmt.executeQuery("select * from Hotels");
            while(rs.next())
            {
                model.addRow(new Object[]{rs.getString("hotel_Id"),
                    rs.getString("hotel_Name"),rs.getString("no_Of_Rooms_Available")
                    ,rs.getString("state_Id")});
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
        jt.setEnabled(false);
        jt.setBounds(30, 40, 180, 150);
        JScrollPane sp = new JScrollPane(jt);
        jf.add(sp);
        jf.setSize(800, 400);
        jf.setVisible(true);
    }
});
```

## **Customers Table**

```
//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;

public class customers
{
    private JPanel pn,pn1,pn2,pn3;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_cus_id,JL_cus_name,JL_noOfPeople,JL_emailId,JL_phoneNo,JL_address;
    private JTextField JTF_cus_id,JTF_cus_name,JTF_noOfPeople,JTF_emailId,JTF_phoneNo,JTF_address;
    Connection con;
    ResultSet rs;
    Statement stmt;
    private JMenuItem insert4,update4,view4,delete4;
    private List customersList;

    public customers(JPanel pn,JFrame jframe,JMenuItem insert4,JMenuItem update4,JMenuItem
view4,JMenuItem delete4)
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectDatabase();

        this.jframe=jframe;
        this.insert4=insert4;
        this.update4=update4;
        this.view4=view4;
        this.delete4=delete4;

        JL_cus_id=new JLabel("Customer_Id:");
        JTF_cus_id=new JTextField(10);
        JL_cus_name=new JLabel("Customer Name:");
        JTF_cus_name=new JTextField(10);
        JL_noOfPeople=new JLabel("No Of People:");
        JTF_noOfPeople=new JTextField(10);
        JL_emailId=new JLabel("Email_Id:");
        JTF_emailId=new JTextField(10);
        JL_phoneNo=new JLabel("Phone No:");
        JTF_phoneNo=new JTextField(10);
        JL_address=new JLabel("Address:");
    }
}
```

## Its Vacay Time-Tour Management System

```
JTF_address=new JTextField(10);

        this.pn=pn;
    }
public void connectDatabase()
{
    try
    {
        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");

        stmt=con.createStatement();
        stmt.executeUpdate("commit");

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}
private void displaySQLErrors(SQLException e)
{
    JOptionPane.showMessageDialog(pn1,"\\nSQLException: " + e.getMessage() +
"\n"+ "SQLState: " + e.getSQLState() + "\n" + "VendorError: " + e.getErrorCode() + "\n");
}
public void loadCustomers()
{
    try
    {
        customersList.removeAll();
        rs=stmt.executeQuery("select * from Customers");
        while(rs.next())
        {
            customersList.add(rs.getString("customer_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}
public void buildGUI()
{
    insert4.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_cus_id.setText(null);
            JTF_cus_name.setText(null);
            JTF_noOfPeople.setText(null);
            JTF_emailId.setText(null);
        }
    });
}
```

## Its Vacay Time-Tour Management System

```
JTF_phoneNo.setText(null);
JTF_address.setText(null);

pn.removeAll();
jframe.invalidate();
jframe.validate();
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_cus_id);
pn1.add(JTF_cus_id);
pn1.add(JL_cus_name);
pn1.add(JTF_cus_name);
pn1.add(JL_noOfPeople);
pn1.add(JTF_noOfPeople);
pn1.add(JL_emailId);
pn1.add(JTF_emailId);
pn1.add(JL_phoneNo);
pn1.add(JTF_phoneNo);
pn1.add(JL_address);
pn1.add(JTF_address);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_insert);
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,175,135);

pn2=new JPanel(new FlowLayout());
customersList=new List(10);
loadCustomers();
pn2.add(customersList);
pn2.setBounds(300,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO Customers VALUES(" +
                JTF_cus_id.getText() + ", "
                + "'" +JTF_cus_name.getText() + "', "
                +JTF_noOfPeople.getText() + ", "
                +"""+JTF_emailId.getText()
                +"", "+JTF_phoneNo.getText()+"','"+"+JTF_address.getText()+"')";
            int i = stmt.executeUpdate(query);
        }
    }
});
```

## Its Vacay Time-Tour Management System

```
        JOptionPane.showMessageDialog(pn, "\nInserted  
" + i + " rows successfully");  
        loadCustomers();  
        System.out.println("Done");  
    }  
    catch(SQLException e)  
{  
        displaySQLErrors(e);  
    }  
}  
});  
});  
update4.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent aevt)  
    {  
        JB_modify=new JButton("Modify");  
  
        JTF_cus_id.setText(null);  
        JTF_cus_name.setText(null);  
        JTF_noOfPeople.setText(null);  
        JTF_emailId.setText(null);  
        JTF_phoneNo.setText(null);  
        JTF_address.setText(null);  
  
        pn.removeAll();  
        jframe.invalidate();  
        jframe.validate();  
        jframe.repaint();  
  
        pn1=new JPanel();  
        pn1.setLayout(new GridLayout(10,10));  
        pn1.add(JL_cus_id);  
        pn1.add(JTF_cus_id);  
        pn1.add(JL_cus_name);  
        pn1.add(JTF_cus_name);  
        pn1.add(JL_noOfPeople);  
        pn1.add(JTF_noOfPeople);  
        pn1.add(JL_emailId);  
        pn1.add(JTF_emailId);  
        pn1.add(JL_phoneNo);  
        pn1.add(JTF_phoneNo);  
        pn1.add(JL_address);  
        pn1.add(JTF_address);  
  
        pn3=new JPanel(new FlowLayout());  
        pn3.add(JB_modify);  
        pn1.setBounds(115,80,300,250);  
        pn3.setBounds(200,350,75,35);  
  
        pn2=new JPanel(new FlowLayout());  
        customersList=new List(10);  
        loadCustomers();  
        pn2.add(customersList);  
        pn2.setBounds(200,350,300,180);
```

```

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

customersList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt)
    {
        try
        {
            rs=stmt.executeQuery("select * from customers");
            while (rs.next())
            {
                if
(rs.getString("customer_id").equals(customersList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_cus_id.setText(rs.getString("customer_id"));

                JTF_cus_name.setText(rs.getString("customer_Name"));

                JTF_noOfPeople.setText(rs.getString("no_Of_People"));

                JTF_emailId.setText(rs.getString("emailId"));

                JTF_phoneNo.setText(rs.getString("phone_No"));

                JTF_address.setText(rs.getString("address"));

            }
        }
        catch (SQLException selectException)
        {
            displaySQLErrors(selectException);
        }
    }
});

JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            int a=JOptionPane.showConfirmDialog(pn,"Are you
sure do you want to update:");
            if(a==JOptionPane.YES_OPTION)
            {

```

```

        String
    pack=JOptionPane.showInputDialog(pn,"Enter no of people:");
    JTF_noOfPeople.setText(pack);
    String query="UPDATE Customers set
no_of_People='"+pack+"' where customer_Id="+JTF_cus_id.getText();
    int i=stmt.executeUpdate(query);
    JOptionPane.showMessageDialog(pn,"\\nUpdated
"+i+" rows successfully");
    loadCustomers();
}
}
catch(SQLException e)
{
    displaySQLErrors(e);
}
}
});
});
delete4.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent aevt)
{
JB_delete=new JButton("Delete");

JTF_cus_id.setText(null);
JTF_cus_name.setText(null);
JTF_noOfPeople.setText(null);
JTF_emailId.setText(null);
JTF_phoneNo.setText(null);
JTF_address.setText(null);

pn.removeAll();
jframe.invalidate();
jframe.validate();
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_cus_id);
pn1.add(JTF_cus_id);
pn1.add(JL_cus_name);
pn1.add(JTF_cus_name);
pn1.add(JL_noOfPeople);
pn1.add(JTF_noOfPeople);
pn1.add(JL_emailId);
pn1.add(JTF_emailId);
pn1.add(JL_phoneNo);
pn1.add(JTF_phoneNo);
pn1.add(JL_address);
pn1.add(JTF_address);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115,80,300,250);

```

## Its Vacay Time-Tour Management System

```
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
customersList=new List(10);
loadCustomers();
pn2.add(customersList);
pn2.setBounds(200,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

customersList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt)
    {
        try
        {
            rs=stmt.executeQuery("select * from customers");
            while (rs.next())
            {
                if
(rs.getString("customer_id").equals(customersList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_cus_id.setText(rs.getString("customer_id"));

                JTF_cus_name.setText(rs.getString("customer_name"));

                JTF_noOfPeople.setText(rs.getString("no_Of_People"));

                JTF_emailId.setText(rs.getString("emailId"));

                JTF_phoneNo.setText(rs.getString("phone_No"));

                JTF_address.setText(rs.getString("address"));

            }
        }
        catch (SQLException selectException)
        {
            displaySQLErrors(selectException);
        }
    }
});
JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
```

## Its Vacay Time-Tour Management System

```
try
{
    int a=JOptionPane.showConfirmDialog(pn,"Are you
sure do you want to Delete:");
    if(a==JOptionPane.YES_OPTION)
    {
        //String query="DELETE FROM STATES WHERE
state_Id="+statesList.getSelectedItem();

        String query="DELETE FROM Customers WHERE
customer_Id="+JTF_cus_id.getText();
        int i=stmt.executeUpdate(query);
        JOptionPane.showMessageDialog(pn, "\nDeleted
"+i+" rows successfully");
        loadCustomers();
    }
}
catch(SQLException e)
{
    displaySQLErrors(e);
}
});

view4.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent aevt)
{
    pn.removeAll();
    jframe.invalidate();
    jframe.validate();
    jframe.repaint();

    JLabel view=new JLabel("Customers View");
    JB_view=new JButton("View");
    Font myFont = new Font("Serif",Font.BOLD,50);
    view.setFont((myFont));

    pn1=new JPanel();
    pn2=new JPanel();
    pn1.add(view);
    pn2.add(JB_view);
    pn.add(pn1);
    pn.add(pn2);
    pn.setBounds(500,800,300,300);
    pn.setLayout(new FlowLayout());

    jframe.add(pn);
    jframe.setSize(800,800);
    jframe.validate();

    JB_view.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent aevt)
{
    JFrame jf=new JFrame("Customers Details");
    jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    jf.setVisible(true);
}
});
}
});
```

```
JTable jt;
DefaultTableModel model = new DefaultTableModel();
jt = new JTable(model);
model.addColumn("Customer_ID");
model.addColumn("Customer Name");
model.addColumn("No Of People");
model.addColumn("Email Id");
model.addColumn("Phone No");
model.addColumn("Address");
try
{
    rs=stmt.executeQuery("select * from Customers");
    while(rs.next())
    {
        model.addRow(new Object[]{rs.getString(1),
                               rs.getString(2),rs.getString(3),rs.getString(4),
                               rs.getString(5),rs.getString(6)});
    }
}
catch(SQLException e)
{
    displaySQLErrors(e);
}
jt.setEnabled(false);
jt.setBounds(30, 40, 300, 300);
JScrollPane jsp = new JScrollPane(jt);
jf.add(jsp);
jf.setSize(800, 400);
jf.setVisible(true);
}
});
```

### **Payments Table**

```
//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;
import java.util.StringTokenizer;

public class payments
{
    private JPanel pn1,pn2,pn3,pn;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_pay_id,JL_cost,JL_cus_id,JL_rating;
    private JTextField JTF_pay_id,JTF_cost,JTF_cus_id,JTF_rating;
```

## Its Vacay Time-Tour Management System

```
Connection con;
ResultSet rs;
Statement stmt;
private JMenuItem insert5,update5,view5,delete5;
private List paymentList;
private Choice cusId,payId;
public payments(JPanel pn,JFrame jframe,JMenuItem insert5,JMenuItem update5,JMenuItem view5,JMenuItem delete5)
{
    try
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");
    }
    catch (Exception e)
    {
        System.err.println("Unable to find and load driver");
        System.exit(1);
    }
    connectDatabase();

    this.jframe=jframe;
    this.insert5=insert5;
    this.update5=update5;
    this.view5=view5;
    this.delete5=delete5;

    JL_pay_id=new JLabel("Payment_Id:");
    JTF_pay_id=new JTextField(10);
    JL_cost=new JLabel("Cost:");
    JTF_cost=new JTextField(10);
    JL_rating=new JLabel("Rating:");
    JTF_rating=new JTextField(10);
    JL_cus_id=new JLabel("Customer Id:");
    JTF_cus_id=new JTextField(10);

    this.pn=pn;
}
public void connectDatabase()
{
    try
    {
        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");

        stmt=con.createStatement();
        stmt.executeUpdate("commit");

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}
```

## Its Vacay Time-Tour Management System

```
private void displaySQLErrors(SQLException e)
{
    JOptionPane.showMessageDialog(pn1, "\nSQLException: " + e.getMessage() +
"\n"+ "SQLState:      " + e.getSQLState() + "\n"+ "VendorError:  " + e.getErrorCode() + "\n");
}
public void loadCustomersPayments()
{
    try
    {
        paymentList=new List();
        paymentList.removeAll();
        rs=stmt.executeQuery("select * from Payments");
        while(rs.next())
        {
            paymentList.add(rs.getString("payment_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void loadCustomers()
{
    try
    {
        cusId=new Choice();
        cusId.removeAll();
        rs=stmt.executeQuery("select * from Customers");
        while(rs.next())
        {
            cusId.add(rs.getString("customer_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void loadPayments()
{
    try
    {
        payId=new Choice();
        payId.removeAll();
        rs=stmt.executeQuery("select * from Payments");
        while(rs.next())
        {
            payId.add(rs.getString("payment_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}
```

```

        }

    }

public void buildGUI()
{
    insert5.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_pay_id.setText(null);
            JTF_cost.setText(null);
            JTF_rating.setText(null);
            JTF_cus_id.setText(null);

            loadCustomers();
            loadPayments();

            pn.removeAll();
            jframe.invalidate();
            jframe.validate();
            jframe.repaint();

            pn1=new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_pay_id);
            pn1.add(JTF_pay_id);
            pn1.add(JL_cost);
            pn1.add(JTF_cost);
            pn1.add(JL_rating);
            pn1.add(JTF_rating);
            pn1.add(JL_cus_id);
            pn1.add(JTF_cus_id);

            pn3=new JPanel(new FlowLayout());
            pn3.add(JB_insert);
            pn1.setBounds(115,80,300,250);
            pn3.setBounds(200,350,75,35);

            pn2=new JPanel(new FlowLayout());
            paymentList=new List(10);
            loadCustomersPayments();
            pn2.add(paymentList);
            pn2.setBounds(200,350,300,180);

            pn.add(pn1);
            pn.add(pn3);
            pn.add(pn2);

            paymentList.setEnabled(false);
            pn.setLayout(new BorderLayout());
            jframe.add(pn);
            jframe.setSize(800,800);
            jframe.validate();
        }
    }
}

```

## Its Vacay Time-Tour Management System

```
JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO Payments VALUES(" +
                JTF_pay_id.getText() + ","
                +JTF_cost.getText() +"," +JTF_rating.getText()
                +"," +JTF_cus_id.getText()+"";
            int i = stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn,"\\nInserted
                "+i+" rows successfully");
            loadCustomersPayments();
            System.out.println("Done");
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }
});
```

```
update5.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_modify=new JButton("Modify");

        JTF_pay_id.setText(null);
        JTF_cost.setText(null);
        JTF_rating.setText(null);
        JTF_pay_id.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_pay_id);
        pn1.add(JTF_pay_id);
        pn1.add(JL_cost);
        pn1.add(JTF_cost);
        pn1.add(JL_rating);
        pn1.add(JTF_rating);
        pn1.add(JL_cus_id);
        pn1.add(JTF_cus_id);

        pn3=new JPanel(new FlowLayout());
        pn3.add(JB_modify);
        pn1.setBounds(115,80,300,250);
        pn3.setBounds(200,350,75,35);
```

## Its Vacay Time-Tour Management System

```
pn2=new JPanel(new FlowLayout());
paymentList=new List(10);
loadCustomersPayments();
pn2.add(paymentList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

paymentList.addItemListener(new ItemListener()
{
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from Payments");

            while (rs.next())
            {
                if
(rs.getString("payment_Id").equals(paymentList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_pay_id.setText(rs.getString("payment_Id"));
                JTF_cost.setText(rs.getString("cost"));
                JTF_rating.setText(rs.getString("rating"));

                JTF_cus_id.setText(rs.getString("customer_Id"));
            }
        }
        catch (SQLException exp)
        {
            displaySQLErrors(exp);
        }
    }
});

JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try
        {
            int a=JOptionPane.showConfirmDialog(pn1,"Are you sure do
you want to update:");
            if(a== JOptionPane.YES_OPTION)
            {
```

```

        String
rating=JOptionPane.showInputDialog(pn,"Enter rating:");
JTF_rating.setText(rating);
String query="update Payments set
rating='"+rating+
"' where payment_Id="+JTF_pay_id.getText();
int i=stmt.executeUpdate(query);
JOptionPane.showMessageDialog(pn,"\nUpdated "+i+
" rows successfully");
loadCustomersPayments();
}
}
catch(SQLException exp)
{
    displaySQLErrors(exp);
}
}
});
});
delete5.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent aevt)
{
JB_delete=new JButton("Delete");

JTF_pay_id.setText(null);
JTF_cost.setText(null);
JTF_rating.setText(null);
JTF_pay_id.setText(null);

pn.removeAll();
jframe.invalidate();
jframe.validate();
jframe.repaint();

pn1=new JPanel();
pn1.setLayout(new GridLayout(10,10));
pn1.add(JL_pay_id);
pn1.add(JTF_pay_id);
pn1.add(JL_cost);
pn1.add(JTF_cost);
pn1.add(JL_rating);
pn1.add(JTF_rating);
pn1.add(JL_cus_id);
pn1.add(JTF_cus_id);

pn3=new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115,80,300,250);
pn3.setBounds(200,350,75,35);

pn2=new JPanel(new FlowLayout());
paymentList=new List(10);

```

## Its Vacay Time-Tour Management System

```
loadCustomersPayments();
pn2.add(paymentList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

paymentList.addItemListener(new ItemListener()
{
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from
Payments");
            StringTokenizer st=new
StringTokenizer(paymentList.getSelectedItem());
            String q=st.nextToken();

            while (rs.next())
            {

                if
                (rs.getString("payment_Id").equals(q))
                    break;
            }
            if (!rs.isAfterLast())
            {

                JTF_pay_id.setText(rs.getString("payment_Id"));

                JTF_cost.setText(rs.getString("cost"));

                JTF_rating.setText(rs.getString("rating"));

                JTF_cus_id.setText(rs.getString("customer_Id"));
            }
        catch (SQLException selectException)
        {
            displaySQLErrors(selectException);
        }
    }
});
```

JB\_delete.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent e)  
    {  
        try  
        {

```

        int a=JOptionPane.showConfirmDialog(pn,"Are you
        sure do you want to delete:");
        if(a==JOptionPane.YES_OPTION)
        {
            StringTokenizer st=new
StringTokenizer(paymentList.getSelectedItem(),"->");
            String query="DELETE FROM Payments WHERE
            payment_Id="+st.nextToken();
            int i=stmt.executeUpdate(query);

            JOptionPane.showMessageDialog(pn,"\nDeleted "+i+" rows successfully");
            loadCustomersPayments();
        }
    }
    catch(SQLException exp)
    {
        displaySQLErrors(exp);
    }
}
});

view5.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        JLabel view=new JLabel("Payments View");
        JB_view=new JButton("View");
        Font myFont = new Font("Serif",Font.BOLD,50);
        view.setFont(myFont);

        pn1=new JPanel();
        pn2=new JPanel();
        pn1.add(view);
        pn2.add(JB_view);
        pn.add(pn1);
        pn.add(pn2);
        pn.setLayout(new FlowLayout());

        jframe.add(pn);
        jframe.setSize(800,800);
        jframe.validate();

        JB_view.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e)
            {
                JFrame jf=new JFrame("Payments Details");
                JTable jt;
            }
        });
    }
});

```

```

        DefaultTableModel model = new
        DefaultTableModel();
        jt= new JTable(model);
        model.addColumn("Payment Id");
        model.addColumn("Cost");
        model.addColumn("Rating");
        model.addColumn("Customer Id");
        try
        {
            rs=stmt.executeQuery("select * from
            Payments");
            while(rs.next())
            {
                model.addRow(new
                Object[]{rs.getString("payment_Id"),
                rs.getString("cost"),rs.getString("rating"),rs.getString("customer_Id")});
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }

        jt.setEnabled(false);
        jt.setBounds(30, 40, 180, 150);
        JScrollPane sp = new JScrollPane(jt);
        jf.add(sp);
        jf.setSize(800, 400);
        jf.setVisible(true);
    }
}
});
```

## Selects Table

```

//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;
import java.util.StringTokenizer;

public class selects
{
    private JPanel pn,pn1,pn2,pn3;
    private JFrame jframe;
    private JButton JB_insert,JB_modify,JB_view,JB_delete;
    private JLabel JL_selectYN,JL_cus_id,JL_vehicleType,JL_st_id;
    private JTextField JTF_st_id,JTF_selectYN,JTF_cus_id,JTF_vehicleType;
```

## Its Vacay Time-Tour Management System

```
Connection con;
ResultSet rs;
Statement stmt;
private JMenuItem insert6,update6,view6,delete6;
private List selectsList;
private Choice stateId,customerId;

public selects(JPanel pn,JFrame jframe,JMenuItem insert6,JMenuItem update6,JMenuItem
view6,JMenuItem delete6)
{
    try
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");
    }
    catch (Exception e)
    {
        System.err.println("Unable to find and load driver");
        System.exit(1);
    }
    connectDatabase();

    this.jframe=jframe;
    this.insert6=insert6;
    this.update6=update6;
    this.view6=view6;
    this.delete6=delete6;

    JL_st_id=new JLabel("State_Id:");
    JTF_st_id=new JTextField(10);
    JL_cus_id=new JLabel("Customer Id:");
    JTF_cus_id=new JTextField(10);
    JL_selectYN=new JLabel("Select Yes Or No:");
    JTF_selectYN=new JTextField(10);
    JL_vehicleType=new JLabel("Vehicle Type:");
    JTF_vehicleType=new JTextField(10);

    this.pn=pn;
}

public void connectDatabase()
{
    try
    {
        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","it19737099","vasavi");

        stmt=con.createStatement();
        stmt.executeUpdate("commit");

    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}
```

## Its Vacay Time-Tour Management System

```
}

private void displaySQLErrors(SQLException e)
{
    JOptionPane.showMessageDialog(pn1, "\nSQLException: " + e.getMessage() +
"\n"+ "SQLState:      " + e.getSQLState() + "\n"+ "VendorError:   " + e.getErrorCode() + "\n");
}

public void loadSelects()
{
    try
    {
        selectsList=new List();
        selectsList.removeAll();
        rs=stmt.executeQuery("select * from Selects");
        while(rs.next())
        {
            selectsList.add(rs.getString("state_Id")+"-
">"+rs.getString("customer_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void loadStates()
{
    try
    {
        stateId=new Choice();
        stateId.removeAll();
        rs=stmt.executeQuery("select * from States");
        while(rs.next())
        {
            stateId.add(rs.getString("state_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void loadCustomers()
{
    try
    {
        customerId=new Choice();
        customerId.removeAll();
        rs=stmt.executeQuery("select * from Customers");
        while(rs.next())
        {
            customerId.add(rs.getString("customer_Id"));
        }
    }
    catch(SQLException e)
    {
        displaySQLErrors(e);
    }
}
```

## Its Vacay Time-Tour Management System

```
        }
    }
public void buildGUI()
{
    insert6.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt)
        {
            JB_insert=new JButton("Submit");

            JTF_st_id.setText(null);
            JTF_cus_id.setText(null);
            JTF_selectYN.setText(null);
            JTF_vehicleType.setText(null);

            loadStates();
            loadCustomers();

            pn.removeAll();
            jframe.invalidate();
            jframe.validate();
            jframe.repaint();

            pn1=new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_st_id);
            pn1.add(JTF_st_id);
            pn1.add(JL_cus_id);
            pn1.add(JTF_cus_id);
            pn1.add(JL_selectYN);
            pn1.add(JTF_selectYN);
            pn1.add(JL_vehicleType);
            pn1.add(JTF_vehicleType);

            pn3=new JPanel(new FlowLayout());
            pn3.add(JB_insert);
            pn1.setBounds(115,80,300,250);
            pn3.setBounds(200,350,75,35);

            pn2=new JPanel(new FlowLayout());
            selectsList=new List(10);
            loadSelects();
            pn2.add(selectsList);
            pn2.setBounds(200,350,300,180);

            pn.add(pn1);
            pn.add(pn3);
            pn.add(pn2);

            pn.setLayout(new BorderLayout());
            jframe.add(pn);
            jframe.setSize(800,800);
            jframe.validate();

            JB_insert.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent aevt)
                {
                    String st_id=JTF_st_id.getText();
                    String cus_id=JTF_cus_id.getText();
                    String selectYN=JTF_selectYN.getText();
                    String vehicleType=JTF_vehicleType.getText();

                    if(st_id.equals("") || cus_id.equals("") || selectYN.equals("") || vehicleType.equals(""))
                    {
                        JOptionPane.showMessageDialog(jframe,"Please fill all fields");
                    }
                    else
                    {
                        // Logic to insert data into database
                    }
                }
            });
        }
    });
}
```

```

        public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            String query= "INSERT INTO Selects VALUES ('"+
                JTF_selectYN.getText() + "','" +
                +JTF_vehicleType.getText() +"','" +
                +JTF_cus_id.getText() +
                ","+JTF_st_id.getText() +")";
            int i = stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn, "\nInserted
                "+i+" rows successfully");
            loadSelects();
            System.out.println("Done");
        }
        catch(SQLException e)
        {
            displaySQLErrors(e);
        }
    }
});

update6.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_modify=new JButton("Modify");

        JTF_st_id.setText(null);
        JTF_cus_id.setText(null);
        JTF_selectYN.setText(null);
        JTF_vehicleType.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_st_id);
        pn1.add(JTF_st_id);
        pn1.add(JL_cus_id);
        pn1.add(JTF_cus_id);
        pn1.add(JL_selectYN);
        pn1.add(JTF_selectYN);
        pn1.add(JL_vehicleType);
        pn1.add(JTF_vehicleType);

        pn3=new JPanel(new FlowLayout());
        pn3.add(JB_modify);
        pn1.setBounds(115,80,300,250);
        pn3.setBounds(200,350,75,35);

        pn2=new JPanel(new FlowLayout());
    }
});

```

## Its Vacay Time-Tour Management System

```
selectsList=new List(10);
loadSelects();
pn2.add(selectsList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

selectsList.addItemListener(new ItemListener() {
    @Override
    public void itemStateChanged(ItemEvent ievt)
    {
        try
        {
            rs=stmt.executeQuery("select * from Selects");
            while (rs.next())
            {
                if
(rs.getString("state_Id").equals(selectsList.getSelectedItem()) &&
rs.getString("customer_Id").equals(selectsList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

JTF_selectYN.setText(rs.getString("selectYesOrNo"));

JTF_vehicleType.setText(rs.getString("vehicle_Type"));

JTF_cus_id.setText(rs.getString("customer_Id"));

JTF_st_id.setText(rs.getString("state_Id"));
}
catch (SQLException selectException)
{
    displaySQLErrors(selectException);
}
}
});

JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        try
        {
            int a= JOptionPane.showConfirmDialog(pn1,"Are you
sure do you want to update:");
            if(a==JOptionPane.YES_OPTION)
```

```

        {
            String
            vehicleType=JOptionPane.showInputDialog(pn,"Enter
                New Vehicle Type:");
            JTF_vehicleType.setText(vehicleType);
            String query="update Selects set
                vehicle_Type='"+vehicleType+
                "' where
                customer_Id="+JTF_cus_id.getText()+" and
                state_Id="+JTF_st_id.getText();

            int i=stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(pn, "\nUpdated
                "+i+" rows successfully");
            loadSelects();
        }
    } catch(SQLException exp)
    {
        displaySQLErrors(exp);
    }
}
});;
});

delete6.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        JB_delete=new JButton("Delete");

        JTF_st_id.setText(null);
        JTF_cus_id.setText(null);
        JTF_selectYN.setText(null);
        JTF_vehicleType.setText(null);

        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10,10));
        pn1.add(JL_st_id);
        pn1.add(JTF_st_id);
        pn1.add(JL_cus_id);
        pn1.add(JTF_cus_id);
        pn1.add(JL_selectYN);
        pn1.add(JTF_selectYN);
        pn1.add(JL_vehicleType);
        pn1.add(JTF_vehicleType);

        pn3=new JPanel(new FlowLayout());
        pn3.add(JB_delete);
        pn1.setBounds(115,80,300,250);
    }
});

```

## Its Vacay Time-Tour Management System

```
pn3.setBounds(180,350,75,35);

pn2=new JPanel(new FlowLayout());
selectsList=new List(10);
loadSelects();
pn2.add(selectsList);
pn2.setBounds(250,350,300,180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jframe.add(pn);
jframe.setSize(800,800);
jframe.validate();

selectsList.addItemListener(new ItemListener() {
    @Override
    public void itemStateChanged(ItemEvent e)
    {
        try
        {
            rs=stmt.executeQuery("select * from Selects");

            while (rs.next())
            {

                if
(rs.getString("state_Id").equals(selectsList.getSelectedItem()))
                &&
rs.getString("customer_Id").equals(selectsList.getSelectedItem()))
                    break;
            }
            if (!rs.isAfterLast())
            {

JTF_selectYN.setText(rs.getString("selectYesOrNo"));

JTF_vehicleType.setText(rs.getString("vehicle_Type"));

JTF_cus_id.setText(rs.getString("customer_Id"));

JTF_st_id.setText(rs.getString("state_Id"));
}
catch (SQLException selectException)
{
        displaySQLErrors(selectException);
}
}
});
JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
{
```

```

        try
        {
            int a=JOptionPane.showConfirmDialog(pn,"Are you
sure do you want to delete:");
            if(a==JOptionPane.YES_OPTION)
            {
                StringTokenizer st=new
                StringTokenizer(selectsList.getSelectedItem(),"->");
                String query="DELETE FROM Selects WHERE
state_Id="+st.nextToken()+" and customer_Id="+st.nextToken();
                int i=stmt.executeUpdate(query);

                JOptionPane.showMessageDialog(pn,"Deleted "+i+" rows
successfully");
                loadSelects();
            }
            catch(SQLException exp)
            {
                displaySQLErrors(exp);
            }
        }
    });
};

view6.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt)
    {
        pn.removeAll();
        jframe.invalidate();
        jframe.validate();
        jframe.repaint();

        JB_view=new JButton("View");
        JLabel view=new JLabel("OverView On Data");
        Font myFont = new Font("Serif",Font.BOLD,50);
        view.setFont((myFont));

        pn1=new JPanel();
        pn2=new JPanel();
        pn1.add(view);
        pn2.add(JB_view);
        pn.add(pn1);
        pn.add(pn2);
        pn.setLayout(new FlowLayout());

        jframe.add(pn);
        jframe.setSize(800,800);
        jframe.validate();

        JB_view.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent aevt)
            {
                JFrame jf=new JFrame("Total Details");

```

```

        JTable jt;
        DefaultTableModel model = new DefaultTableModel();

        jt= new JTable(model);
        model.addColumn("State Id");
        model.addColumn("Customer Id");
        model.addColumn("Selects Yes Or No");
        model.addColumn("Vehicle Type");

        try
        {
            rs=stmt.executeQuery("select * from Selects");
            while(rs.next())
            {
                model.addRow(new Object[]{rs.getString(1),
                    rs.getString(2),rs.getString(3)
                    ,rs.getString(4)});
            }
        }
        catch(SQLException exp)
        {
            displaySQLErrors(exp);
        }
        jt.setEnabled(false);
        jt.setBounds(30, 40, 180, 150);
        JScrollPane sp = new JScrollPane(jt);
        jf.add(sp);
        jf.setSize(800, 400);
        jf.setVisible(true);
    }
}
});
```

**Main**

```

//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
import java.awt.*;
import javax.swing.*;
public class TMSUI extends JFrame
{
    private JPanel Jpn0,Jpn1;
    private JMenuBar mbar;
    private JMenu states,places,hotels,customers,payments,selects,statesAvailable;
    private JMenuItem insert1,update1,view1,delete1;
    private JMenuItem insert2,update2,view2,delete2;
    private JMenuItem insert3,update3,view3,delete3;
    private JMenuItem insert4,update4,view4,delete4;
    private JMenuItem insert5,update5,view5,delete5;
    private JMenuItem insert6,update6,view6,delete6;
    private JMenuItem s1,s2,s3,s4,s5;

    private JLabel nameLabel;
```

## Its Vacay Time-Tour Management System

```
public void defineComponents()
{
    Jpn0=new JPanel();
    Jpn1=new JPanel();

    mbar=new JMenuBar();

    statesAvailable=new JMenu("List Of States Available");
    states=new JMenu("States");
    places=new JMenu("Places");
    hotels=new JMenu("Hotels");
    customers=new JMenu("Customers");
    payments=new JMenu("Payments");
    selects=new JMenu("Selects");

    labelName=new JLabel("It's Vacay Time-Tour Management System");
}

public void addComponents()
{
    add(Jpn0);
    Jpn1.add(labelName);
    Jpn1.setAlignmentY(CENTER_ALIGNMENT);
    Jpn1.setBounds(500,500,800,100);
    Jpn0.add(Jpn1);

    setJMenuBar(mbar);

    mbar.add(statesAvailable);
    mbar.add(states);
    mbar.add(places);
    mbar.add(hotels);
    mbar.add(customers);
    mbar.add(payments);
    mbar.add(selects);

    statesAvailable.add(s1=new JMenuItem("Goa"));
    statesAvailable.add(s2=new JMenuItem("Kerala"));
    statesAvailable.add(s3=new JMenuItem("Telangana"));
    statesAvailable.add(s4=new JMenuItem("Maharashtra"));
    statesAvailable.add(s5=new JMenuItem("Uttar Pradesh"));

    states.add(insert1=new JMenuItem("Insert"));
    states.add(update1=new JMenuItem("Update"));
    states.add(view1=new JMenuItem("View"));
    states.add(delete1=new JMenuItem("Delete"));

    places.add(insert2=new JMenuItem("Insert"));
    places.add(update2=new JMenuItem("Update"));
    places.add(view2=new JMenuItem("View"));
    places.add(delete2=new JMenuItem("Delete"));

    hotels.add(insert3=new JMenuItem("Insert"));
    hotels.add(update3=new JMenuItem("Update"));
    hotels.add(view3=new JMenuItem("View"));
    hotels.add(delete3=new JMenuItem("Delete"));
}
```

## Its Vacay Time-Tour Management System

```
customers.add(insert4=new JMenuItem("Insert"));
customers.add(update4=new JMenuItem("Update"));
customers.add(view4=new JMenuItem("View"));
customers.add(delete4=new JMenuItem("Delete"));

payments.add(insert5=new JMenuItem("Insert"));
payments.add(update5=new JMenuItem("Update"));
payments.add(view5=new JMenuItem("View"));
payments.add(delete5=new JMenuItem("Delete"));

selects.add(insert6=new JMenuItem("Insert"));
selects.add(update6=new JMenuItem("Update"));
selects.add(view6=new JMenuItem("View"));
selects.add(delete6=new JMenuItem("Delete"));

}

public void registerComponents()
{
    states obj_states=new states(Jpn0,TMSUI.this,insert1,update1,view1,delete1);
    obj_states.buildGUI();
    places obj_places=new places(Jpn0,TMSUI.this,insert2,update2,view2,delete2);
    obj_places.buildGUI();
    hotels obj_hotels=new hotels(Jpn0,TMSUI.this,insert3,update3,view3,delete3);
    obj_hotels.buildGUI();
    customers obj_customers=new customers(Jpn0,TMSUI.this,insert4,update4,view4,delete4);
    obj_customers.buildGUI();
    payments obj_payments=new payments(Jpn0,TMSUI.this,insert5,update5,view5,delete5);
    obj_payments.buildGUI();
    selects obj_selects=new selects(Jpn0,TMSUI.this,insert6,update6,view6,delete6);
    obj_selects.buildGUI();
}
public TMSUI()
{
    defineComponents();
    addComponents();
    registerComponents();
    setSize(400,500);
    setBackground(Color.GRAY);
    setVisible(true);
    setTitle("IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

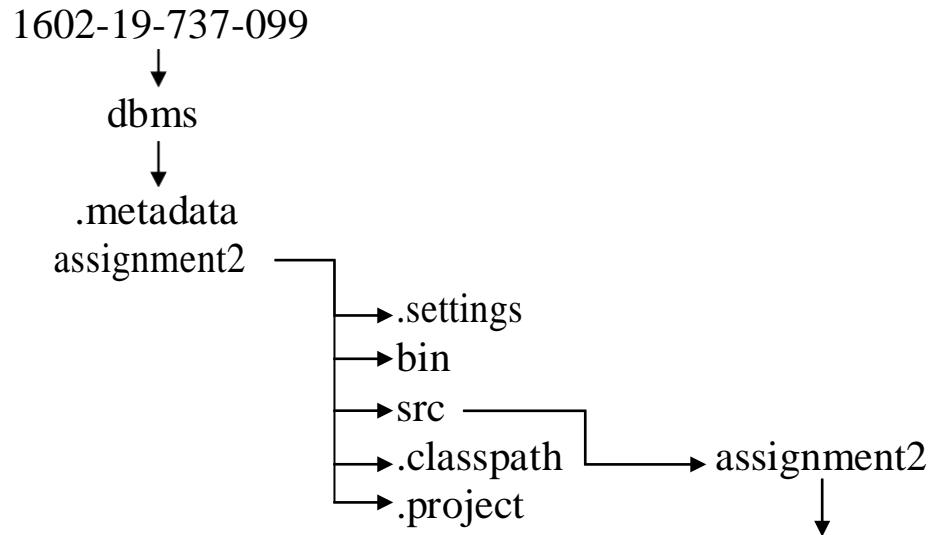
//B.Sankeerthana 1602-19-737-099 ITS VACAY TIME TOUR MANAGEMENT SYSTEM
package TMS;
public class Main
{
    public static void main(String []args)
    {
        new TMSUI();
    }
}
```

## Github links and folder structure:

Link: [https://github.com/Bashetti/DBMS-PROJECT\\_2\\_2](https://github.com/Bashetti/DBMS-PROJECT_2_2)

### Folder structure:

1602-19-737-099.rar  
dbms\_report.pdf  
Its Vacay Time\_Tour Management  
System.pdf ( assignment 1)



## Its Vacay Time-Tour Management System

The image shows three screenshots of a file explorer interface, likely from a terminal-based file manager like WinSCP or similar. The top two screenshots show the contents of the 'src' directory, while the bottom one shows the contents of the 'TMS' folder within it.

**Screenshot 1 (Top):** Shows the root directory structure. The 'src' folder contains several files and folders:

Name	Date modified	Type	Size
.settings	18-06-2021 12:11	File folder	
bin	18-06-2021 12:11	File folder	
src	18-06-2021 12:11	File folder	
.classpath	14-06-2021 10:44	CLASSPATH File	1 KB
.project	14-06-2021 10:43	PROJECT File	1 KB

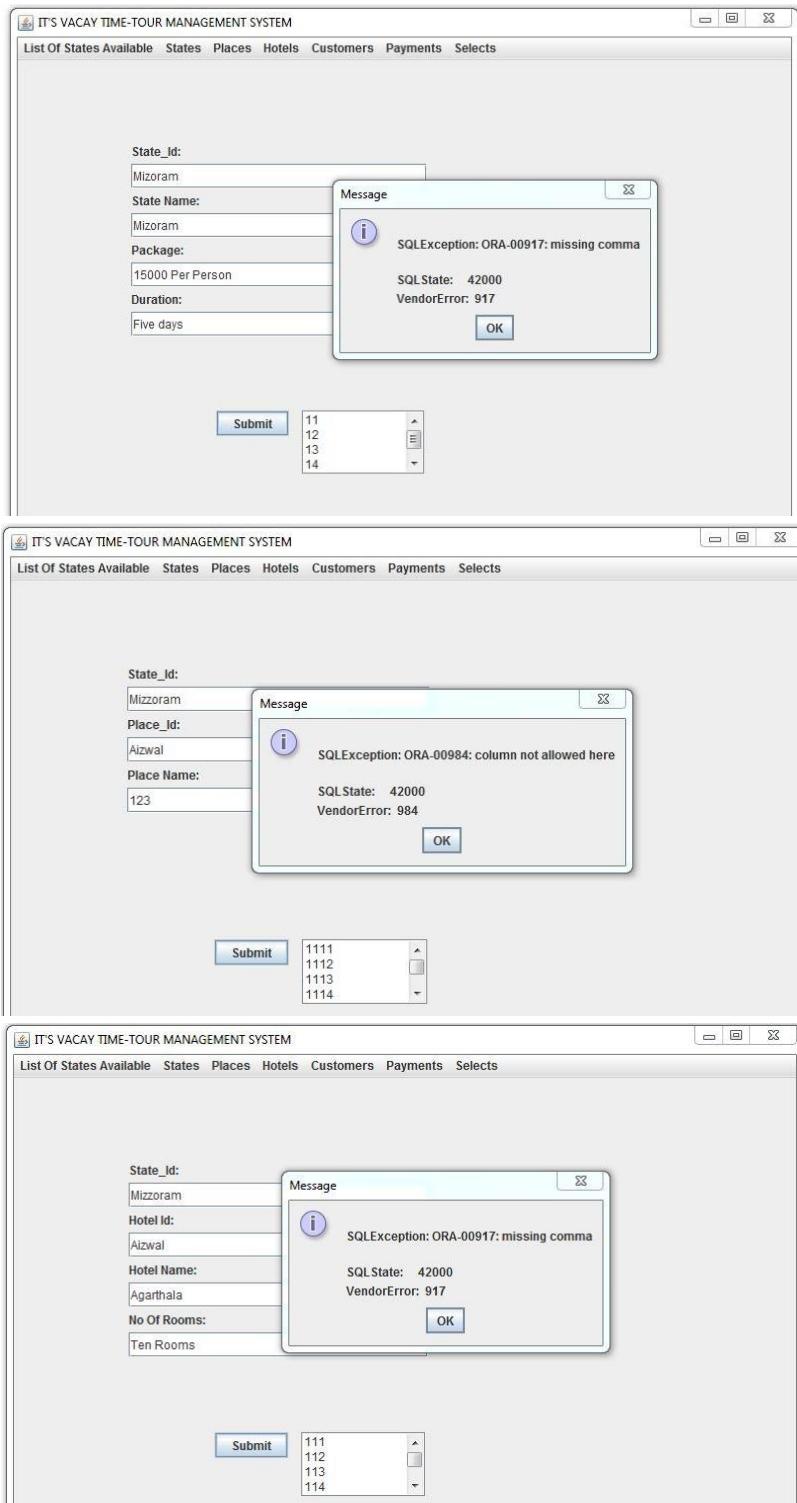
**Screenshot 2 (Middle):** Shows the 'src' directory itself. It contains a single folder named 'TMS':

Name	Date modified	Type	Size
TMS	18-06-2021 12:11	File folder	

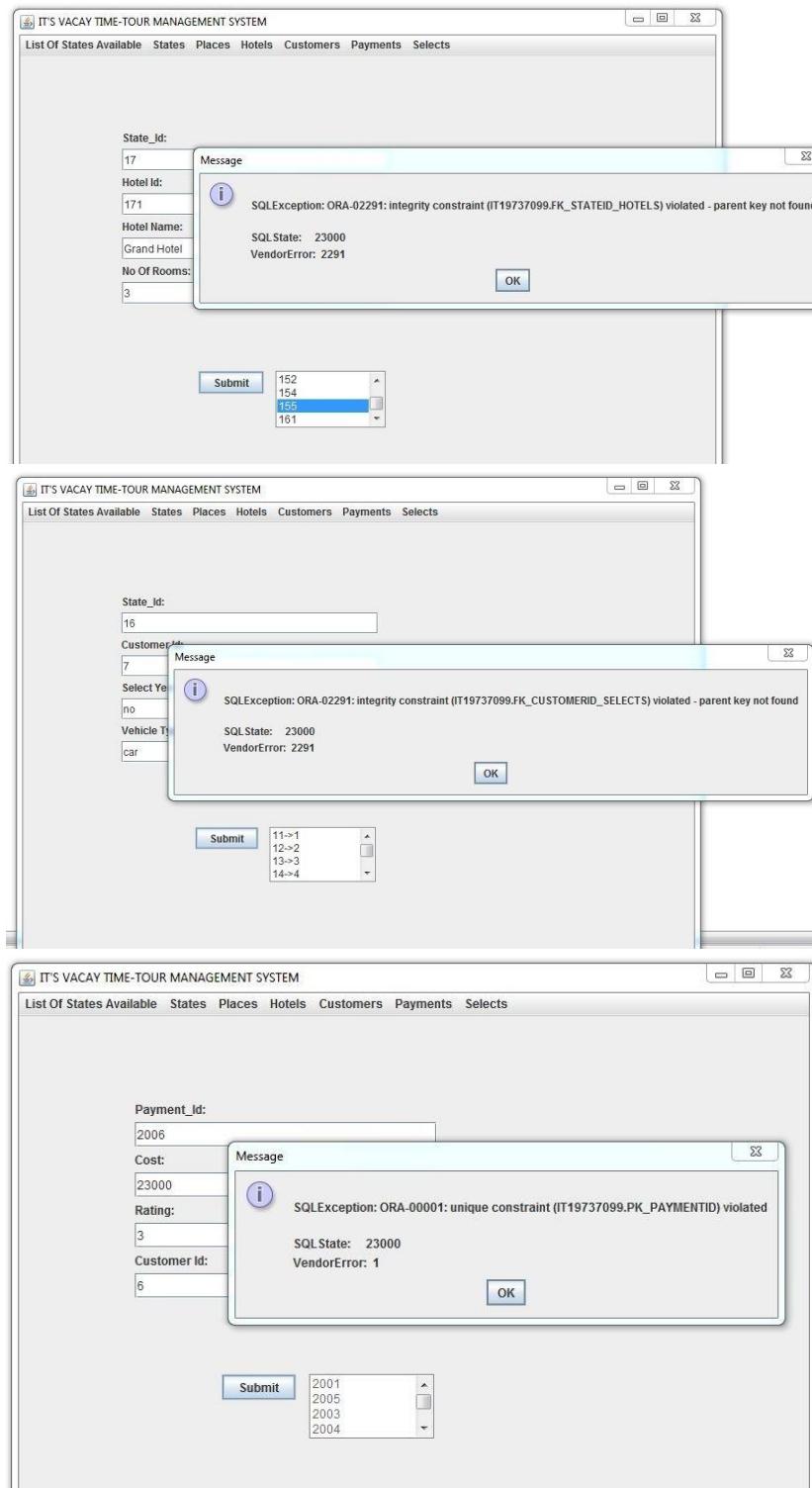
**Screenshot 3 (Bottom):** Shows the contents of the 'TMS' folder. It contains eight Java source files:

Name	Date modified	Type	Size
customers.java	16-06-2021 17:14	Java Source File	13 KB
hotels.java	16-06-2021 22:05	Java Source File	13 KB
Main.java	15-06-2021 21:59	Java Source File	1 KB
payments.java	16-06-2021 22:12	Java Source File	13 KB
places.java	16-06-2021 22:11	Java Source File	12 KB
selects.java	17-06-2021 10:56	Java Source File	13 KB
states.java	16-06-2021 17:13	Java Source File	12 KB
TMSUI.java	17-06-2021 15:43	Java Source File	5 KB

# TESTING

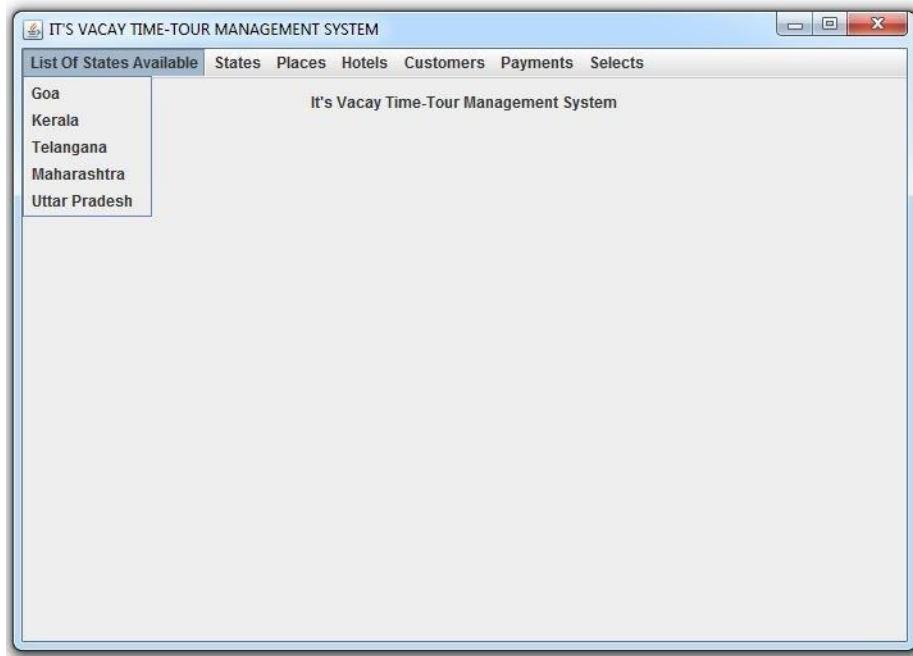
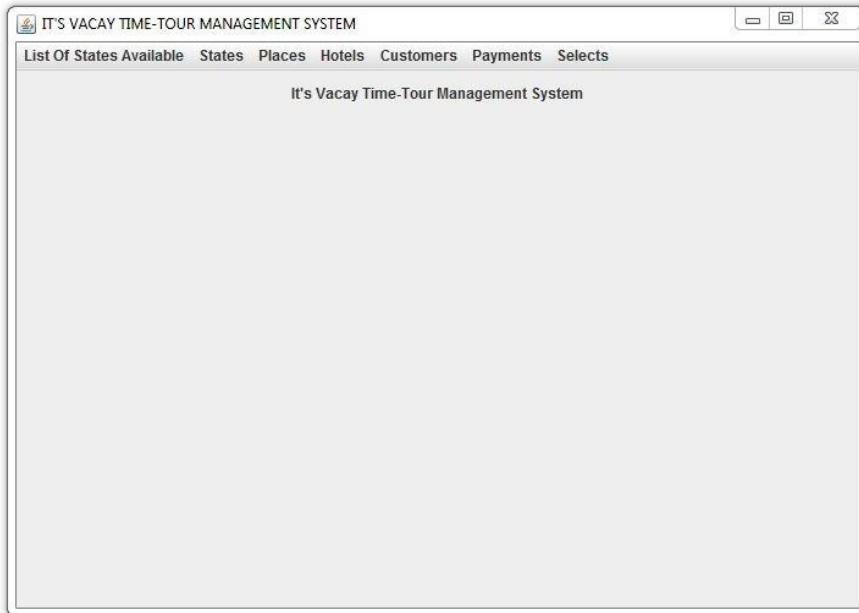


## Its Vacay Time-Tour Management System



# RESULTS

## Main



## States Table

The state\_id is an unique value and it can only have a numeric value and not anything else, similarly the state\_Name can only be a string. In case of any unexpected or incorrect input it throws an exception.

## Its Vacay Time-Tour Management System

The image displays three windows of a software application titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM".

**Top Window:** Shows a menu bar with "List Of States Available", "States", "Places", "Hotels", "Customers", "Payments", and "Selects". A context menu is open over the "States" tab, listing "Insert", "Update", "View", and "Delete". The main area displays the text "It's Vacay Time-Tour Management System".

**Middle Window:** Shows the "States" module interface. It includes fields for "State\_Id" (16), "State Name" (Karnataka), "Package" (11000 Per Person), and "Duration" (3). Below these fields is a "Submit" button and a dropdown menu containing the values 11, 12, 13, and 14.

**Bottom Window:** Shows the same "States" module interface as the middle window. After clicking the "Submit" button, a "Message" dialog box appears stating "Inserted 1 rows successfully" with an "OK" button.

## Its Vacay Time-Tour Management System

**Screenshot 1: State Insertion Failure**

State\_ID:  
16  
State Name:  
Madhya Pradesh  
Package:  
14000 Per Person  
Duration:  
5

Message

SQLException: ORA-00001: unique constraint (IT19737099.PK\_STATEID) violated  
SQLState: 23000  
VendorError: 1

OK

Submit

11  
12  
13  
14

**Screenshot 2: States View**

States View

**Screenshot 3: States Details**

State_ID	State Name	Package	Days
11	Goa	12000 per person	4
12	Kerala	13500 per person	4
13	Telangana	17500 per person	5
14	Maharashtra	12800 Per Person	3
15	Uttar Pradesh	14800 Per Person	4
16	Karnataka	11000 Per Person	3

**Screenshot 4: State Update Confirmation**

State\_ID:  
16  
State Name:  
Karnataka  
Package:  
11000 Per Person  
Duration:  
3

Select an Option

Are you sure want to update?

Yes No Cancel

Modify

13  
14  
15  
16

## Its Vacay Time-Tour Management System

The image displays three windows of a software application titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM".

**Screenshot 1:** A dialog box titled "Input" is open, prompting for a new package value. The input field contains "13000 Per Person". The main window shows fields for State\_ID (16), State Name (Karnataka), Package (11000 Per Person), and Duration (3). A dropdown menu labeled "Modify" is open, showing options 13, 14, 15, and 16, with 16 selected.

**Screenshot 2:** A message dialog box titled "Message" indicates that 1 row was updated successfully. The main window has been updated: State\_ID is now 16, Package is 13000 Per Person, and Duration is 3. The "Modify" dropdown still shows 16 selected.

**Screenshot 3:** A window titled "States View" displays a table titled "States Details". The table lists the following data:

State_ID	State Name	Package	Days
11	Goa	12000 per person	4
12	Kerala	13500 per person	4
13	Telangana	17500 per person	5
14	Maharashtra	12800 Per Person	3
15	Uttar Pradesh	14800 Per Person	4
16	Karnataka	13000 Per Person	3

## Its Vacay Time-Tour Management System

The image consists of three vertically stacked screenshots of a software application window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM".

**Screenshot 1:** The window shows a form with fields: State\_Id (16), State Name (Karnataka), Package (13000 Per Person), Duration (3). A "Delete" button is visible. A modal dialog box titled "Select an Option" asks "Are you sure want to Delete:" with "Yes", "No", and "Cancel" buttons. Below the form is a dropdown menu with items 13, 14, 15, and 16, where 16 is highlighted.

**Screenshot 2:** The same window after deletion. The modal dialog box now says "Deleted 1 rows succesfully" with an "OK" button. The dropdown menu still shows item 16.

**Screenshot 3:** The window title is "States View". It contains a "View" button and a table titled "States Details" with the following data:

State_ID	State Name	Package	Days
11	Goa	12000 per person	4
12	Kerala	13500 per person	4
13	Telangana	17500 per person	5
14	Maharashtra	12800 Per Person	3
15	Uttar Pradesh	14800 Per Person	4

## PLACES TABLE

Place\_Id is the primary and unique key. State\_id is the foreign key. For one state\_id there are 5 respective place\_ids, upon wrong state\_id error is popped.

The image consists of three vertically stacked screenshots of a Java Swing application window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM".

**Screenshot 1:** Shows the "Places" form. Fields: State\_Id (16), Place\_Id (1611), Place Name (The Heritage Hotel). A message box says "Inserted 1 rows successfully".

Submit	1111 1112 1113 1114
--------	------------------------------

**Screenshot 2:** Shows the "Places" form. Fields: State\_Id (17), Place\_Id (1), Place Name (Mysore). An error message box shows an SQLException: ORA-02291: integrity constraint (IT19737099.FK\_STATEID\_PLACES) violated - parent key not found, SQL State: 23000, VendorError: 2291.

Submit	1111 1112 1113 1114
--------	------------------------------

**Screenshot 3:** Shows the "Places View" form with a single "View" button.

## Its Vacay Time-Tour Management System

**Places Details**

Place Id	Place Name	State Id
1111	Fort Aguada	11
1112	Bogdeshwara Temple	11
1113	Vagator Beach	11
1114	Anjuna Flea Market	11
1211	Alleppey	12
1212	Kochi	12
1213	Munnar	12
1214	Thiruvananthapuram	12
1311	Ananthagiri	13
1312	Charminar	13
1313	Ramoji Film City	13
1314	Wonderla	13
1411	Mumbai	14
1412	Pune	14
1413	Nagpur	14
1414	Mahabaleshwar	14
1115	Regis Mata Fort	11
1315	Birla Mandir And Tank Bund	13
1215	Periyar National Park	12
1415	Ellora Caves	14
1511	Agra	15
1512	Fatehpur Sikri	15
1513	Soor Sarovar	15
1514	Dayal Bagh	15
1515	Sikandra	15
1611	The Heritage Hotel	16

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16

Place\_Id: 1611

Place Name: The Heritage Hotel

Select an Option

Are you sure do you want to update?

Yes No Cancel

Modify

1513  
1514  
1515  
1611

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16

Place\_Id: 1611

Place Name: Banglore

Message

Updated 1 rows successfully

OK

Modify

1513  
1514  
1515  
1611

Since state\_id is foreign key and place\_id is primary key. The deletion of state\_id will delete respective place\_ids in places table.

The screenshot displays two instances of the 'Places View' application window from the 'IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM'. Both windows show a table titled 'Places Details' with columns: Place Id, Place Name, and State Id.

**Top Window Data:**

Place Id	Place Name	State Id
1212	Kochi	12
1213	Munnar	12
1214	Thiruvananthapuram	12
1311	Ananthagiri	13
1312	Charminar	13
1313	Ramoji Film City	13
1314	Wonderla	13
1411	Mumbai	14
1412	Pune	14
1413	Nagpur	14
1414	Mahabaleshwar	14
1115	Regis_Mata Fort	11
1315	Birla Mandir And Tank Bund	13
1215	Periyar National Park	12
1415	Ellora Caves	14
1511	Agra	15
1512	Fatehpur Sikri	15
1513	Soor Sarovar	15
1514	Dayal Bagh	15
1515	Sikandra	15
1611	Banglore	16

**Bottom Window Data:**

Place Id	Place Name	State Id
1111	Fort Aguada	11
1112	Bogdeshwara Temple	11
1113	Vagator Beach	11
1114	Anjuna Flea Market	11
1211	Alleppey	12
1212	Kochi	12
1213	Munnar	12
1214	Thiruvananthapuram	12
1311	Ananthagiri	13
1312	Charminar	13
1313	Ramoji Film City	13
1314	Wonderla	13
1411	Mumbai	14
1412	Pune	14
1413	Nagpur	14
1414	Mahabaleshwar	14
1115	Regis_Mata Fort	11
1315	Birla Mandir And Tank Bund	13
1215	Periyar National Park	12
1415	Ellora Caves	14
1511	Agra	15

## Hotels Table

Hotel\_Id is the primary key and for respective foreign key state\_id hotel\_Ids exists.

## Its Vacay Time-Tour Management System

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

Hotels

Insert  
Update  
View  
Delete

State\_Id:

State Name:

Package:

Duration:

Submit

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

Hotels

State\_Id:

Hotel Id:

Hotel Name:

No Of Rooms:

Message

Inserted 1 rows successfully

OK

Submit

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

Hotels

State\_Id:

Hotel Id:

Hotel Name:

No Of Rooms:

Message

SQLException: ORA-02291: integrity constraint (IT19737099.FK\_STATEID\_HOTELS) violated - parent key not found

SQLState: 23000  
VendorError: 2291

OK

Submit

## Its Vacay Time-Tour Management System

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

**State\_Id:**  
16

**Hotel Id:**  
161

**Hotel Name:**  
Grand Hotel

**No Of Rooms:**  
3

**Message**

SQLException: ORA-00001: unique constraint (IT19737099.PK\_HOTELID) violated

SQLState: 23000  
VendorError: 1

**OK**

**Submit**

152  
154  
**155**  
161

**Hotels Details**

Hotel Id	Hotel Name	No Of Rooms	State Id
111	Double Tree By Hilton	100	11
112	Grand Hyatt Goa	100	11
113	Serene Water Resort	100	11
114	Shoolin Homestay	100	11
115	The Green Boutique	100	11
121	Emarald Resorts	100	12
122	Hyatt Regency Thrissur	100	12
123	Welcome Hotel Coimbatore	100	12
124	Courtyard By Marrorott	100	12
125	The Raviz Kadavu	100	12
131	Marigold Hotel	100	13
132	ITC Kohenur	100	13
133	The Golconda Hotel	100	13
134	Pragathi Resorts	100	13
135	Leonia Holistic Destination	100	13
141	The Taj Mahal Palace	100	14
142	Della Resorts	100	14
143	Riva Beach Resorts	100	14
144	Balaji Sarovar Premeiere	100	14
145	Taj Holiday Village Resort	100	14
151	TajMahal Lucknow	100	15
153	Crowne Plaza Gurgaon	100	15
152	Hyatt Regency Lucknow	100	15
154	Vijan Mahal	100	15
155	The Landmark Towers	100	15
161	The New Grand Hotel	20	16

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

**State\_Id:**  
16

**Hotel Id:**  
161

**Hotel Name:**  
The New Grand Hotel

**No Of Rooms:**  
20

**Select an Option**

Are you sure do you want to update?

**Yes** **No** **Cancel**

**Modify**

152  
154  
155  
**161**

## Its Vacay Time-Tour Management System

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

State\_Id:  
16

Hotel Id:  
161

Hotel Name:  
The New Grand Hotel

No Of Rooms:  
4

**Message**

Updated 1 rows successfully

**OK**

**Modify**

152  
154  
155  
**161**

**Hotels View**

**View**

**Hotels Details**

Hotel Id	Hotel Name	No Of Rooms	State Id
121	Emarald Resorts	100	12
122	Hyatt Regency Thrissur	100	12
123	WelcomeHotel Coimbatore	100	12
124	Courtyard By Marriot	100	12
125	The Raviz Kadavu	100	12
131	Marigold Hotel	100	13
132	ITC Kohenur	100	13
133	The Golconda Hotel	100	13
134	Pragathi Resorts	100	13
135	Leonia Holistic Destination	100	13
141	The Taj Mahal Palace	100	14
142	Della Resorts	100	14
143	Riva Beach Resorts	100	14
144	Balaji Sarovar Premeiere	100	14
145	taj Holiday Village Resort	100	14
151	TajMahal Lucknow	100	15
153	Crowne Plaza Gurgaon	100	15
152	Hyatt Regency Lucknow	100	15
154	Vijan Mahal	100	15
155	The Landmark Towers	100	15
161	The New Grand Hotel	4	16

**IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM**

List Of States Available States Places Hotels Customers Payments Selects

State\_Id:  
16

Hotel Id:  
161

Hotel Name:  
The New Grand Hotel

No Of Rooms:  
4

**Select an Option**

Are you sure do you want to delete?

**Yes**   **No**   **Cancel**

**Delete**

152  
154  
155  
**161**

## Its Vacay Time-Tour Management System

The screenshot shows two windows of the application. The top window is titled 'List Of States Available' and contains fields for State\_Id (16), Hotel Id (161), Hotel Name (The New Grand Hotel), and No Of Rooms (4). A message box titled 'Message' displays 'Deleted 1 rows successfully'. The bottom window is titled 'Hotels View' and shows a grid of hotel details:

Hotel Id	Hotel Name	No Of Rooms	State Id
115	The Green Boutique	100	11
121	Emarald Resorts	100	12
122	Hyatt Regency Thrissur	100	12
123	WelcomeHotel Coimbatore	100	12
124	Courtyard By Marriot	100	12
125	The Raviz Kadavu	100	12
131	Marigold Hotel	100	13
132	ITC Kohenur	100	13
133	The Golconda Hotel	100	13
134	Pragathi Resorts	100	13
135	Leonia Holistic Destination	100	13
141	The Taj Mahal Palace	100	14
142	Della Resorts	100	14
143	Riva Beach Resorts	100	14
144	Balaji Sarovar Premeiere	100	14
145	Taj Holiday Village Resort	100	14
151	TajMahal Lucknow	100	15
153	Crowne Plaza Gurgaon	100	15
152	Hyatt Regency Lucknow	100	15
154	Vijan Mahal	100	15
155	The Landmark Towers	100	15

## Customers Table

Customer\_id is the primary key. For different customers,different ids exists.

The screenshot shows the 'Customers' tab of the application. A context menu is open over the customer details section, listing options: Insert, Update, View, and Delete. The customer details are as follows:

Customer_Id:	
Customer Name:	
No Of People:	
Email_Id:	
Phone No:	
Address:	

Below the details are a 'Submit' button and a numeric input field containing values from 1 to 6.

## Its Vacay Time-Tour Management System

**Customer Insertion Screen:**

The screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM". The menu bar includes "List Of States Available", "States", "Places", "Hotels", "Customers", "Payments", and "Selects". The main area contains a form with the following fields:

Customer_Id:	06
Customer Name:	Rishi Shekawath
No Of People:	5
Email_Id:	rishiShekawath@mail.in
Phone No:	7890321276
Address:	-12/A,Leela Nagar,Bhopal..

A "Submit" button is located below the form. To the right, there is a vertical list of numbers from 1 to 5. A message dialog box is displayed in the center-right, stating "Inserted 1 rows successfully" with an "OK" button.

**Customer View Screen:**

The screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM" with a "View" button in the top right. The menu bar is identical to the previous screen. The main area displays a table titled "Customers Details" with the following data:

Customer_ID	Customer Name	No Of People	Email Id	Phone No	Address
1	Raman Bhalla	4	raman@gmail.com	9246990011	FlatNo 109,Prakruthi ...
2	Ishitha Iyer	10	ishitha@yahoo.com	9394959611	FlatNo 111,Tulasi Niv...
3	Akshara Shetty	5	akshara@yahoo.com	9146738910	Flat No 382,Chaitanya...
4	Naithik Naidu	2	naidu@gmail.com	9849556688	FlatNo 285,Krishna A...
5	Ameya Shetty	3	ameya@yahoo.com	9294930114	Flat NO 1011,Sri Lak...
6	Rishi Shekawath	5	rishiShekawath@mail...7890321276	H_NO:2-9-12/A,Leela...	

**Customer Update Screen:**

The screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM". The menu bar is identical. The main area contains a form with the same set of fields as the insertion screen, but with different values (Customer\_Id: 6). Below the form is a "Modify" button. To the right, there is a vertical list of numbers from 1 to 6. A confirmation dialog box is displayed in the center-right, asking "Are you sure do you want to update:" with "Yes", "No", and "Cancel" buttons.

## Its Vacay Time-Tour Management System

The screenshots illustrate the application's functionality:

- Screenshot 1:** A modal dialog titled "Input" asks for "Enter no of people:" with a value of "2". The main window shows a list of people with item "6" selected.
- Screenshot 2:** A modal dialog titled "Message" displays "Updated 1 rows succesfully" after a modification. The main window shows the list of people with item "6" still selected.
- Screenshot 3:** A table titled "Customers Details" lists customer information. The columns are Customer\_ID, Customer Name, No Of People, Email Id, Phone No, and Address. The data includes entries for Raman Bhalla, Ishitha Iyer, Akshara Shetty, Naitthik Naidu, Ameya Shetty, and Rishi Shekawath.

Customer_ID	Customer Name	No Of People	Email Id	Phone No	Address
1	Raman Bhalla	4	raman@gmail.com	9246990011	FlatNo 109,Prakruthi ...
2	Ishitha Iyer	10	ishitha@yahoo.com	9394959611	FlatNo 111,Tulasi Niv...
3	Akshara Shetty	5	akshara@yahoo.com	9146738910	Flat No 382,Chaitanya...
4	Naitthik Naidu	2	naidu@gmail.com	9849556688	FlatNo 285,Krishna A...
5	Ameya Shetty	3	ameya@yahoo.com	9294930114	Flat NO 1011,Sri Lak...
6	Rishi Shekawath	2	rishiShekawath@mai...	7890321276	H_NO:2-9-12/A,Leela...

## Payments Table

Payment\_id is the primary key and for respective customer\_id (foreign key), payment\_id exists.

## Its Vacay Time-Tour Management System

The image displays three screenshots of a Windows application window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM". The window has a menu bar with items: List Of States Available, States, Places, Hotels, Customers, Payments, Selects. The "Payments" item is highlighted.

The first screenshot shows a sub-menu for "Payments" with options: Insert, Update, View, and Delete. The "Insert" option is selected. The main area of the window is empty.

The second screenshot shows the "Insert" form. It contains fields for Payment\_Id (2006), Cost (26000), and Rating (4). A dropdown menu for Customer Id shows the value 6. A message box says "Inserted 1 rows successfully" with an OK button. Below the form is a "Submit" button and a dropdown menu showing values 2001, 2005, 2003, and 2004.

The third screenshot shows the same "Insert" form after attempting to insert again. The message box displays an SQL exception: "SQLException: ORA-00001: unique constraint (IT19737099.PK\_PAYMENTID) violated" along with SQL State: 23000 and VendorError: 1. The dropdown menu for Customer Id still shows the value 6.

## Its Vacay Time-Tour Management System

**Payments View**

Payment Id	Cost	Rating	Customer Id
2001	48000	4	1
2005	0	0	5
2003	67500	4	3
2004	35000	4	4
2002	135000	4.5	2
2006	26000	4	6

**Payment Details**

Payment_Id:	2006
Cost:	26000
Rating:	4
Customer Id:	6

Select an Option

Are you sure do you want to update?

Yes   No   Cancel

**Modify**

2003  
2004  
2002  
**2006**

**Input**

Enter rating:

4.5

OK   Cancel

**Modify**

2003  
2004  
2002  
**2006**

## Its Vacay Time-Tour Management System

The image displays three separate windows of the "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM".

**Top Window:** A form titled "Payments Details" with fields for Payment\_Id (2006), Cost (26000), Rating (4.5), and Customer Id (6). A "Modify" button is present. A message dialog box is overlaid, stating "Updated 1 rows successfully" with an "OK" button.

Payment Id	Cost	Rating	Customer Id
2001	48000	4	1
2005	0	0	5
2003	67500	4	3
2004	35000	4	4
2002	135000	4.5	2
2006	26000	4.5	6

**Middle Window:** A title bar "Payments View" with a "View" button. Below it is a table titled "Payments Details" showing the same data as the top window.

Payment Id	Cost	Rating	Customer Id
2001	48000	4	1
2005	0	0	5
2003	67500	4	3
2004	35000	4	4
2002	135000	4.5	2
2006	26000	4.5	6

**Bottom Window:** A form with fields for Payment\_Id (2006), Cost (26000), Rating (4.5), and Customer Id (6). A "Delete" button is present. A confirmation dialog box is overlaid, asking "Are you sure do you want to delete:" with "Yes", "No", and "Cancel" buttons.

Payment Id	Cost	Rating	Customer Id
2003	2004	2002	2006

## Its Vacay Time-Tour Management System

The top screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM". The menu bar includes "List Of States Available", "States", "Places", "Hotels", "Customers", "Payments", and "Selects". The main area contains fields for "Payment\_Id" (2006), "Cost" (26000), and "Rating" (4.5). A dropdown menu for "Customer Id" shows the value 6. A message dialog box says "Deleted 1 rows successfully" with an "OK" button. Below these are buttons for "Delete" and a dropdown list containing 2003, 2004, 2002, and 2006, with 2006 selected.

The bottom screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM" with the title bar "Payments View" and a "View" button. It displays a table titled "Payments Details" with the following data:

Payment Id	Cost	Rating	Customer Id
2001	48000	4	1
2005	0	0	5
2003	67500	4	3
2004	35000	4	4
2002	135000	4.5	2

### Selects Table

Select\_id connects both customer\_id and state\_id together featuring which customer has chosen which state in order to travle.

This screenshot shows a window titled "IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM" with the menu bar "List Of States Available", "States", "Places", "Hotels", "Customers", "Payments", and "Selects". The "Selects" tab is active, showing a context menu with options: "Insert", "Update", "View", and "Delete". The main area contains four input fields: "State\_Id", "Customer Id", "Select Yes Or No", and "Vehicle Type". Below these is a "Submit" button and a dropdown menu showing the values 11->1, 12->2, 13->3, and 14->4.

## Its Vacay Time-Tour Management System

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16  
Customer Id: 6  
Select Yes Or No: Yes  
Vehicle Type: Bus

Message: Inserted 1 rows successfully

Submit 11->1  
12->2  
13->3  
14->4

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16  
Customer Id: 7  
Select Yes Or No: no  
Vehicle Type: car

Message: SQLException: ORA-02291: integrity constraint (IT19737099.FK\_CUSTOMERID\_SELECTS) violated - parent key not found  
SQL State: 23000  
VendorError: 2291

OK

Submit 11->1  
12->2  
13->3  
14->4

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16  
Customer Id: 6  
Select Yes Or No: Yes  
Vehicle Type: Bus

Select an Option: Are you sure do you want to update?  
Yes No Cancel

Modify 11->1  
12->2  
13->3  
14->4

## Its Vacay Time-Tour Management System

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16  
Customer Id: 6  
Select Yes Or No: Yes  
Vehicle Type: Bus

Input  
Enter New Vehicle Type:  
Car  
OK Cancel

Modify  
11->1  
12->2  
13->3  
14->4

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

State\_Id: 16  
Customer Id: 6  
Select Yes Or No: Yes  
Vehicle Type: Car

Message  
Updated 1 rows successfully  
OK

Modify  
11->1  
12->2  
13->3  
14->4

IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM

List Of States Available States Places Hotels Customers Payments Selects

**OverView On Data** View

Total Details

State Id	Customer Id	Selects Yes Or No	Vehicle_Type
yes	car	1	11
yes	train	2	12
yes	Train	3	13
Yes	Airline	4	14
no		5	15
Yes	Car	6	16

## Its Vacay Time-Tour Management System

**Screenshot 1:** IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM - List Of States Available

Form fields:

- State\_Id: 16
- Customer Id: 6
- Select Yes Or No: Yes
- Vehicle Type: Car

A modal dialog titled "Select an Option" asks: "Are you sure do you want to delete?". Buttons: Yes, No, Cancel.

Below the form is a "Delete" button and a dropdown menu showing options: 13->3, 14->4, 15->5, 16->6 (the last one is selected).

**Screenshot 2:** IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM - List Of States Available

Form fields:

- State\_Id: 16
- Customer Id: 6
- Select Yes Or No: Yes
- Vehicle Type: Car

A modal dialog titled "Message" shows: "Deleted 1 rows successfully". Button: OK.

Below the form is a "Delete" button and a dropdown menu showing options: 13->3, 14->4, 15->5, 16->6 (the last one is selected).

**Screenshot 3:** IT'S VACAY TIME-TOUR MANAGEMENT SYSTEM - OverView On Data

Section header: OverView On Data

Table: Total Details

State Id	Customer Id	Selects Yes Or No	Vehicle_Type
yes	car	1	11
yes	train	2	12
yes	Train	3	13
Yes	Airline	4	14
no		5	15

**DATA FROM DATABASE:****1. States table**

```
SQL> select * from States;
```

STATE_ID	STATE_NAME	PACKAGE
11	Goa	12000 per person
4		
12	Kerala	13500 per person
4		
13	Telangana	17500 per person
5		

STATE_ID	STATE_NAME	PACKAGE
14	Maharashtra	12800 Per Person
3		
15	Uttar Pradesh	14800 Per Person
4		
16	Karnataka	11000 Per Person
3		

6 rows selected.

**2. Places Table**

```
SQL> select * from places where state_id=16;
```

PLACE_ID	PLACE_NAME	STATE_ID
1611	Banglore	16

**3. Hotels Table**

```
SQL> select * from hotels where state_id=16;
```

HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS_AVAILABLE
161	The New Grand Hotel	20
16		

**4. Customers Table**

```
SQL> select * from customers;
```

CUSTOMER_ID	CUSTOMER_NAME	NO_OF_PEOPLE	EMAILID
1	Raman Bhalla	4	raman@gmail.com
9246990011	FlatNo 109,Prakruthi Apartments,Hyderabad		
2	Ishitha Iyer	10	ishitha@yahoo.com
9394959611	FlatNo 111,Tulasi Nivas,Mumbai		
3	Akshara Shetty	5	akshara@yahoo.com
9146738910	Flat No 382,Chaitanya Gateway,Banglore		

CUSTOMER_ID	CUSTOMER_NAME	NO_OF_PEOPLE	EMAILID
4	Naithik Naidu	2	naidu@gmail.com
9849556688	FlatNo 285,Krishna Apartments,Chennai		
5	Ameya Shetty	3	ameya@yahoo.com
9294930114	Flat NO 1011,Sri Lakshmi Ventures,Raipur		
6	Rishi Shekawath	2	rishiShekawath@mail.in
7890321276	H_NO:2-9-12/A,LeelaNagar,Bhopal		

6 rows selected.

## 5. Payments Table

```
SQL> select * from payments;
```

PAYMENT_ID	COST	RATING	CUSTOMER_ID
2001	48000	4	1
2005	0	0	5
2003	67500	4	3
2004	35000	4	4
2002	135000	4.5	2
2006	26000	4	6

```
6 rows selected.
```

## 6. Selects Table

```
SQL> select * from selects;
```

SELEC	VEHICLE_TYPE	CUSTOMER_ID	STATE_ID
yes	car	1	11
yes	train	2	12
yes	Train	3	13
Yes	Airline	4	14
Yes	car	6	16
no		5	15

```
6 rows selected.
```

## **DISCUSSION AND FUTURE WORK**

So far this project has helped us to select travelling destination by the users. Customer choosing a particular state to travel and booking the hotels for residence is the crucial aspect as this is the main parameter for which the project is designed. Since visiting an unknown place brings a lot of excitement to the person, at the same time finding a residence for stay would also be a problem. So to make the customer comfortable, this project helps him out.

In future the most probable aspects could be developing GPS so that one can travel endlessly without any fear ,improvising payment options like adding paytm ,paypal etc and provide the customer with the visual graphical pictures of the destination. Lastly ,providing the additional deals or promotional packages based on the seasons which would attract customers.

[Type text]

Its Vacay Time-Tour Management System

## REFERENCES

<https://www.oracle.com/in/database/technologies/112010-win64soft.html>

<https://docs.oracle.com/javase/8/docs/api/>

<https://www.javatpoint.com/example-to-connect-to-the-mysql-database>

Its Vacay Time-Tour Management System

# THANK YOU

B.Sankeerthana 1602-19 -737-099