

**Module Code & Module Title**

**CS6001NA Advanced Database Systems Development**

**Assessment Weightage & Type**

**40% Individual Coursework**

**Year and Semester**

**2019-20 Autumn**

**Student Name: Reisha Shakya**

**London Met ID: 17030962**

**College ID: NP01CP4A170026**

**Assignment Due Date: 31st January 2020**

**Assignment Submission Date: 31st January 2020**

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.*

**Acknowledgement**

Doing this project was very fascinating and captivating. The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of the coursework. All the credit goes to the supervision and assistance.

I would like to show my gratitude to Ms. Subeksha Shrestha, Course Instructor and Tutor for giving us a good guideline for assignment throughout numerous consultations. I am highly indebted to him for her guidance and constant supervision as well as for providing necessary information regarding the project.

I also owe our deep gratitude to Mr. Rohit Panday for conducting lecture classes and providing all the necessary information and also helping us and clearing out our confusion. I am thankful and fortunate enough to get constant encouragement, support and guidance from everyone.

Thank you for your valuable time and effort throughout the module.

**Table of Contents**

[1. Introduction 2](#_Toc31371570)

[2. Normalization 3](#_Toc31371571)

[2.1. Normalization of figure 1 4](#_Toc31371572)

[2.1.1. UNF(Finding Repeating group) 4](#_Toc31371573)

[2.1.2. 1NF(Remove Repeating Group) 4](#_Toc31371574)

[2.1.3. 2NF(Remove Partial Dependencies) 5](#_Toc31371575)

[2.1.4. 3NF(Remove Transitive Dependencies) 5](#_Toc31371576)

[2.2. Normalization of figure 2 7](#_Toc31371577)

[2.2.1. UNF(Finding Repeating group) 7](#_Toc31371578)

[2.2.2. 1NF(Remove Repeating Group) 7](#_Toc31371579)

[2.2.3. 2NF(Remove Partial Dependencies) 7](#_Toc31371580)

[2.2.4. 3NF (Removing Transitive dependencies) 8](#_Toc31371581)

[2.3. Integration 9](#_Toc31371582)

[3. ER-Diagram 10](#_Toc31371583)

[3.1. Assumption 11](#_Toc31371584)

[4. Data Dictionary 12](#_Toc31371585)

[5. Generation of Database 14](#_Toc31371586)

[5.1. Create Statements 14](#_Toc31371587)

[5.2. Insert Statements 18](#_Toc31371588)

[5.3. Select Statements 41](#_Toc31371589)

[6. Implementation of Web-Based Database Application 52](#_Toc31371590)

[6.1. Basic Webforms 52](#_Toc31371591)

[6.1.1. Staff Details 52](#_Toc31371592)

[6.1.2. Customer Details 53](#_Toc31371593)

[6.1.3. Package Details 54](#_Toc31371594)

[6.1.4. Tour Guide Details 55](#_Toc31371595)

[6.1.5. Designation Details 56](#_Toc31371596)

[6.2. Customer-Package Schedule Form 57](#_Toc31371597)

[6.3. Staff-Role Schedule Form 57](#_Toc31371598)

[6.4. Package-Activity Schedule Form 58](#_Toc31371599)

[7. Testing 59](#_Toc31371600)

[8. User Manual 59](#_Toc31371601)

[9. Further Discussion 60](#_Toc31371602)

[10. References 61](#_Toc31371603)

[11. Appendix 62](#_Toc31371604)

**Table of Figures**

[Figure 1 Steps in Normalization (KERDPRASOP & KERDPRASOP , 2011) 3](#_Toc31371605)

[Figure 2 ERD of the System 10](#_Toc31371606)

[Figure 3 Select Statement for Activity 41](#_Toc31371607)

[Figure 4 Select Statement for Booking 42](#_Toc31371608)

[Figure 5 Select Statement for Customer 43](#_Toc31371609)

[Figure 6 Select Statement for Designation 44](#_Toc31371610)

[Figure 7 Select Statement for Destination 45](#_Toc31371611)

[Figure 8 Select Statement for PackageActivity 46](#_Toc31371612)

[Figure 9 Select Statement for PackageDays 47](#_Toc31371613)

[Figure 10 Select Statement for PackageDestination 48](#_Toc31371614)

[Figure 11 Select Statement for Staff 49](#_Toc31371615)

[Figure 12 Select Statement for TourGuide 50](#_Toc31371616)

[Figure 13 Select Statement for TourPackages 51](#_Toc31371617)

[Figure 14 Staff Detail Page 52](#_Toc31371618)

[Figure 15 Customer Details Page 53](#_Toc31371619)

[Figure 16 Tour Packages Details Page 54](#_Toc31371620)

[Figure 17 Tour Guide Details Page 55](#_Toc31371621)

[Figure 18 Designation Details Page 56](#_Toc31371622)

[Figure 19 Customer-Package Schedule Page 57](#_Toc31371623)

[Figure 20 View Staff-Role Mapping Page 57](#_Toc31371624)

[Figure 21 View Package-Activity Schedule 58](#_Toc31371625)

[Figure 22 Home Page 59](#_Toc31371626)

**Table of Tables**

[Table 1 Data Dictionary 12](#_Toc31371627)

# 1. Introduction

The coursework is an individual assessment weighted 40% of the marks for the module. A Tour and Travel company is established to help customer for making their holiday incredible. The company have a number of staff. Each staff of the company have different roles such as receptionist, tour guide, travel agent etc. assigned by company but the company assign only one role to each staff. The company assigned multiple tour guide for a tour. There are number of tour package available in the company. A customer can take only one package at a time.

# 2. Normalization

Normalization is a process of correcting table structure for reducing redundancy and data anomalies and minimize the amount of space required to store data. It applies a series of rules called normal forms (Coronel & Morris, 2018). The database in the coursework is said to normalize till 3NF.

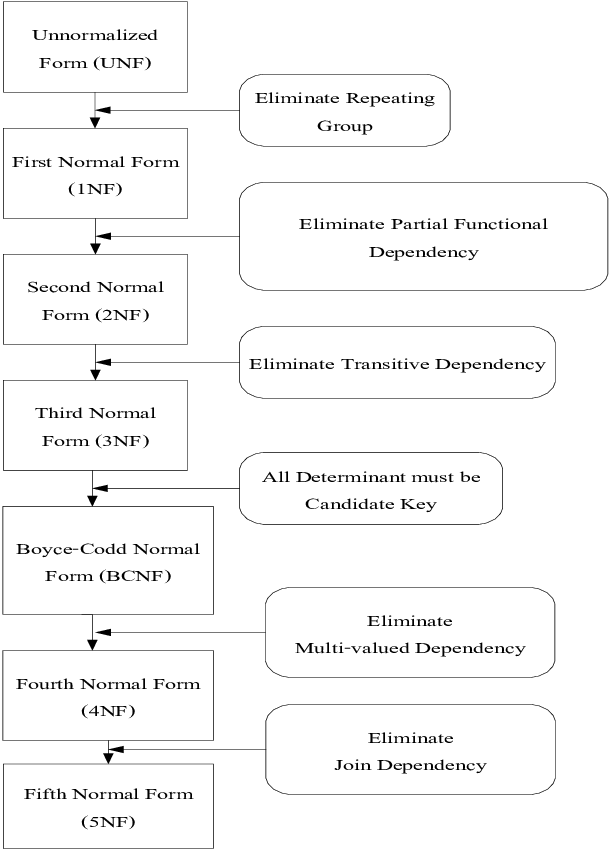


Figure 1 Steps in Normalization (KERDPRASOP & KERDPRASOP , 2011)

## 2.1. Normalization of figure 1

### 2.1.1. UNF(Finding Repeating group)

A database is said to be in UNF when it has not been normalized at all. The rules for creating an un-normalized form are:

* The attributes of an entity should be identified.
* Primary Keys should be identified.
* The repeating group should be identified. (using ‘{}’ curly braces).

In a relation, a characteristic describing a group of multiple entries of the same or multiple types for a single key attribute occurrence can be known as repeating group. Example: a student can have multiple courses, here course is repeating group (Coronel & Morris, 2018).

The UNF for the figure 1, package record is given below:

**Package** = (PackageID, PackageName, TotalNoDays {Destination\_id, Destination}, Description, Price, Difficulty)

### 2.1.2. 1NF(Remove Repeating Group)

In 1NF only single values are allowed at the intersection of each row and column resulting as no repeating groups. For the database to be in 1NF it has to be in UNF. The other rules for 1NF are:

* Repeating groups from UNF must be eliminated to a new table.
* The primary key should be identified of the new table.
* The primary key of new table should be composite having PK of original and new relation.

The 1NF for the figure 1 is given below:

**Package1** = (PackageID, PackageName, Description, Price, Difficulty, TotalNoDays)

**Packageline1** = (PackageID\*, Destination\_id, Destination)

### 2.1.3. 2NF(Remove Partial Dependencies)

For the relation to be in 2NF it has to be in 1NF and also should include no partial dependencies. Partial dependency means when a non-key is functionally dependent on part of a composite key (combination of two or more columns which is used to specify primary key) (Coronel & Morris, 2018). The steps for 2NF are:

* If an entity has a single-column PK no need to check it as it is already in 2NF.
* If an entity has a composite PK, functional dependencies between non key and composite key and parts of composite key should be shown.
* Non-keys that depend on part of composite should be removed to a new table and identify its keys.

The 2NF for figure 1 is given below:

PackageID 🡪

DestinationID 🡪 Destination

PackageID, Destination 🡪

**Package2** = (PackageID, PackageName, TotalNoDays, Difficulty, Description, Price)

**Packageline2** = (PackageID\*, DestinationID\*)

**Destination2** = (DestinationID, Destination)

### 2.1.4. 3NF(Remove Transitive Dependencies)

For the database to be in 3NF it has to be in 2NF and also should include no transitive dependencies. Transitive dependency means when a non-key is functionally dependent on another non-key (keys other than primary key is non-key) (Coronel & Morris, 2018). The steps for 3NF are:

* If an entity has a single non key no need to check it as it is already in 3NF.
* If an entity has 2 or more than 2 non-keys, functional dependencies between non key and non-key should be identified and should be eliminated to a new table.
* Keys of the new table should be identified.

There are no transitive dependencies present in table, table is already in 3 NF

form.

**Package3** = (PackageID, PackageName, TotalNoDays, Difficulty, Description, Price)

**Packageline3** = (PackageID\*, DestinationID\*)

**Destination3** = (DestinationID, Destination)

## 2.2. Normalization of figure 2

### 2.2.1. UNF(Finding Repeating group)

**TrackingPackage** = (PackageId, PackageName, StartDate, EndDate, {StaffId, StaffName}, {Day, TravelDetails, {ActivityNo, PackageActivity, Status}, TravelMode, DifficultyLevel}

### 2.2.2. 1NF(Remove Repeating Group)

**TrackingPackage1** = (PackageId, PackageName, StartDate, EndDate)

**TourGuide** = (PackageId \*, StaffId, StaffName )

**PackageDays1** = (PackageId\*, Day, TravelDetails, TravelMode, DifficultyLevel )

**PackageActivity1** = (PackageId\*, Day\*,ActivityNo, Activity, Status)

### 2.2.3. 2NF(Remove Partial Dependencies)

**TrackingPackage2** = (PackageId, PackageName, StartDate, EndDate)

PackageId 🡪

Day 🡪

PackageId, Day 🡪 TravelDetails, TravelMode, DifficultyLevel

**PackageDays2** = (PackageId\*, Day, TravelDetails, TravelMode, DifficultyLevel)

PackageId 🡪

Day 🡪

PackageId, Day 🡪

ActivityNo 🡪 Activity

PackageId, ActivityNo 🡪

ActivityNo, Day 🡪

PackageId, ActivityNo, Day 🡪 Status

**PackageActivity2**= (PackageId\*, Day\*,ActivityNo\*, Status)

**Activity2** = (ActivityNo, Activity)

PackageId 🡪

StaffId 🡪 StaffName

**TourGuide2** = (PackageId \*, StaffId\*)

**Staff2** = (StaffId, StaffName)

### 2.2.4. 3NF (Removing Transitive dependencies)

There are no transitive dependencies present in table, table is already in 3 NF

form.

**TrackingPackage3** = (PackageId, PackageName, StartDate, EndDate, TourGuide)

**PackageDays3** = (PackageId\*, Day, TravelDetails, TravelMode, DifficultyLevel)

**PackageActivity3**= (PackageId\*, Day\*,ActivityNo\*, Status)

**Activity3** = (ActivityNo, Activity)

**TourGuide3** = (PackageId \*, StaffId\*)

**Staff3** = (StaffId, StaffName)

## 2.3. Integration

**Designation** = (DesignationId, DName)

**TourPackages** = (PackageID, PackageName, TotalNoDays, Difficulty, Description, Price, StartDate, EndDate)

**Customer** = (CustomerId, CustomerName, CPhone, CAddress, CEmail)

**Destination** = (DestinationID, Destination)

**TourGuide** = (PackageId \*, StaffId\*)

**PackageDestination**= (PackageID\*, DestinationID\*)

**Staff** = (staffId, StaffName, Phone, DesignationId\*, Salary)

**PackageDays** = (PackageId\*, Day, TravelDetails, TravelMode, DifficultyLevel)…

**PackageActivity** = (PackageId\*, Day\*,ActivityNo\*, Status)

**Activity** = (ActivityNo, Activity)

**Booking** = (BookingId, CustomerID\*, PackageID\*, BookingDate)

# 3. ER-Diagram

An entity-relationship diagram is a pictorial way of representing the relationship between the entities in the database. ERD is one of the most common data model where objects are divided into entities and its characteristics into attributes. There are 3 basic elements: entity, relationship and attribute (Nishadha, 2017). The below ERD has been created using SQL Developer Data Modeler. It is a free graphical tool that improves productivity and simplifies data modeling tasks where users can create, browse and edit, logical, relational, physical, multi-dimensional, and data type models supporting collaborative development through integrated source code control (Oracle, 2020).

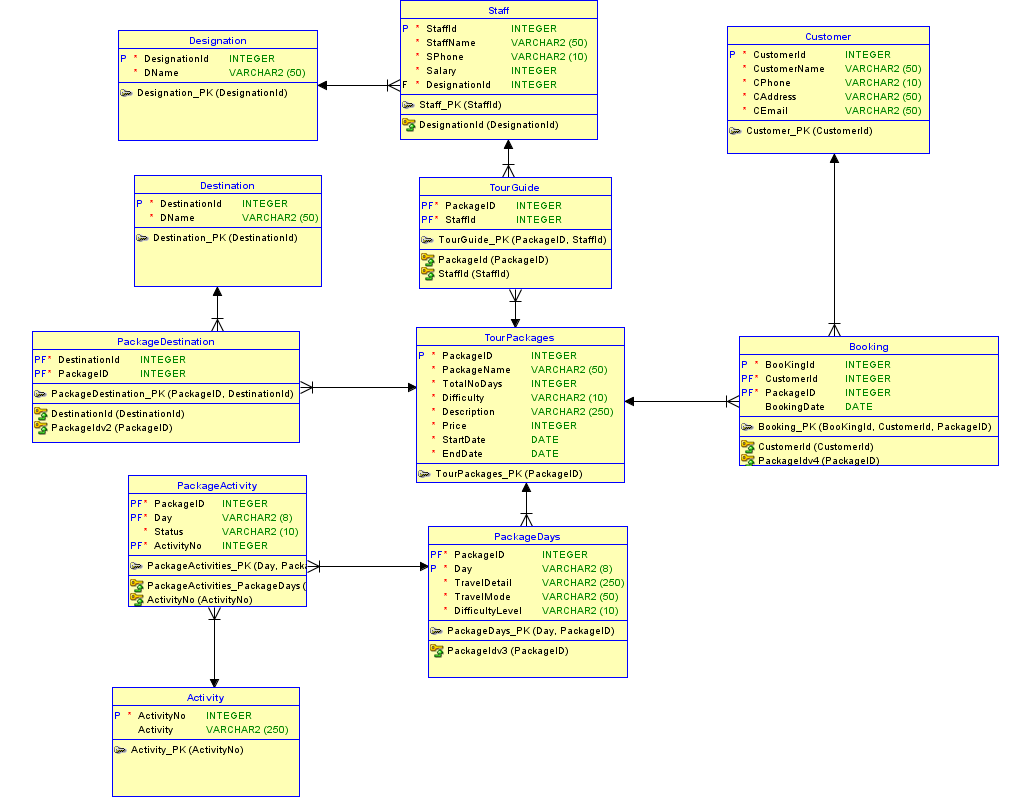


Figure 2 ERD of the System

## 3.1. Assumption

* Different package can have same destination
* Package id should be unique
* A package name can have multiple destination
* Package name can be same so to identify uniquely package id is added
* Each customer can only book one package at a time
* Customer name can be same so to identify uniquely customer id is added
* One customer has Address, Phone number, Email
* Each staff has only one designation
* One staff has Salary, Name, and Phone number
* One customer may Book the same package multiple time so we uniquely identify a booking with a booking id
* Status is dependent on Activity

# 4. Data Dictionary

Table 1 Data Dictionary

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Column\_Name** | **Mandatory** | **DataType\_Kind** | **Logical\_Type\_Name** | **Table\_Name** | **PK\_Flag** | **FK\_Flag** | **Native\_Type** | **T\_Size** |
| Activity | N | Logical Type | VARCHAR | Activity |  |  | VARCHAR2 | 250 |
| ActivityNo | Y | Logical Type | Integer | Activity | P |  | INTEGER |  |
| ActivityNo | Y | Logical Type | Integer | PackageActivity | P | F | INTEGER |  |
| BooKingId | Y | Logical Type | Integer | Booking | P |  | INTEGER |  |
| BookingDate | N | Logical Type | Date | Booking |  |  | DATE |  |
| CAddress | Y | Logical Type | VARCHAR | Customer |  |  | VARCHAR2 | 50 |
| CEmail | Y | Logical Type | VARCHAR | Customer |  |  | VARCHAR2 | 50 |
| CPhone | Y | Logical Type | VARCHAR | Customer |  |  | VARCHAR2 | 10 |
| CustomerId | Y | Logical Type | Integer | Customer | P |  | INTEGER |  |
| CustomerId | Y | Logical Type | Integer | Booking | P | F | INTEGER |  |
| CustomerName | Y | Logical Type | VARCHAR | Customer |  |  | VARCHAR2 | 50 |
| DName | Y | Logical Type | VARCHAR | Designation |  |  | VARCHAR2 | 50 |
| DName | Y | Logical Type | VARCHAR | Destination |  |  | VARCHAR2 | 50 |
| Day | Y | Logical Type | VARCHAR | PackageActivity | P | F | VARCHAR2 | 8 |
| Day | Y | Logical Type | VARCHAR | PackageDays | P |  | VARCHAR2 | 8 |
| Description | Y | Logical Type | VARCHAR | TourPackages |  |  | VARCHAR2 | 250 |
| DesignationId | Y | Logical Type | Integer | Designation | P |  | INTEGER |  |
| DesignationId | Y | Logical Type | Integer | Staff |  | F | INTEGER |  |
| DestinationId | Y | Logical Type | Integer | PackageDestination | P | F | INTEGER |  |
| DestinationId | Y | Logical Type | Integer | Destination | P |  | INTEGER |  |
| Difficulty | Y | Logical Type | VARCHAR | TourPackages |  |  | VARCHAR2 | 10 |
| DifficultyLevel | Y | Logical Type | VARCHAR | PackageDays |  |  | VARCHAR2 | 10 |
| EndDate | Y | Logical Type | Date | TourPackages |  |  | DATE |  |
| PackageID | Y | Logical Type | Integer | PackageDestination | P | F | INTEGER |  |
| PackageID | Y | Logical Type | Integer | TourPackages | P |  | INTEGER |  |
| PackageID | Y | Logical Type | Integer | PackageActivity | P | F | INTEGER |  |
| PackageID | Y | Logical Type | Integer | PackageDays | P | F | INTEGER |  |
| PackageID | Y | Logical Type | Integer | TourGuide | P | F | INTEGER |  |
| PackageID | Y | Logical Type | Integer | Booking | P | F | INTEGER |  |
| PackageName | Y | Logical Type | VARCHAR | TourPackages |  |  | VARCHAR2 | 50 |
| Price | Y | Logical Type | Integer | TourPackages |  |  | INTEGER |  |
| SPhone | Y | Logical Type | VARCHAR | Staff |  |  | VARCHAR2 | 10 |
| Salary | Y | Logical Type | Integer | Staff |  |  | INTEGER |  |
| StaffId | Y | Logical Type | Integer | Staff | P |  | INTEGER |  |
| StaffId | Y | Logical Type | Integer | TourGuide | P | F | INTEGER |  |
| StaffName | Y | Logical Type | VARCHAR | Staff |  |  | VARCHAR2 | 50 |
| StartDate | Y | Logical Type | Date | TourPackages |  |  | DATE |  |
| Status | Y | Logical Type | VARCHAR | PackageActivity |  |  | VARCHAR2 | 10 |
| TotalNoDays | Y | Logical Type | Integer | TourPackages |  |  | INTEGER |  |
| TravelDetail | Y | Logical Type | VARCHAR | PackageDays |  |  | VARCHAR2 | 250 |
| TravelMode | Y | Logical Type | VARCHAR | PackageDays |  |  | VARCHAR2 | 50 |

# 5. Generation of Database

## 5.1. Create Statements

CREATE TABLE Activity

(

ActivityNo INTEGER NOT NULL ,

Activity VARCHAR2 (250)

) ;

ALTER TABLE Activity ADD CONSTRAINT Activity\_PK PRIMARY KEY ( ActivityNo ) ;

CREATE TABLE Booking

(

BooKingId INTEGER NOT NULL ,

CustomerId INTEGER NOT NULL ,

PackageID INTEGER NOT NULL ,

BookingDate DATE

) ;

ALTER TABLE Booking ADD CONSTRAINT Booking\_PK PRIMARY KEY ( BooKingId, CustomerId, PackageID ) ;

CREATE TABLE Customer

(

CustomerId INTEGER NOT NULL ,

CustomerName VARCHAR2 (50) NOT NULL ,

CPhone VARCHAR2 (10) NOT NULL ,

CAddress VARCHAR2 (50) NOT NULL ,

CEmail VARCHAR2 (50) NOT NULL

) ;

ALTER TABLE Customer ADD CONSTRAINT Customer\_PK PRIMARY KEY ( CustomerId ) ;

CREATE TABLE Designation

(

DesignationId INTEGER NOT NULL ,

DName VARCHAR2 (50) NOT NULL

) ;

ALTER TABLE Designation ADD CONSTRAINT Designation\_PK PRIMARY KEY ( DesignationId ) ;

CREATE TABLE Destination

(

DestinationId INTEGER NOT NULL ,

DName VARCHAR2 (50) NOT NULL

) ;

ALTER TABLE Destination ADD CONSTRAINT Destination\_PK PRIMARY KEY ( DestinationId ) ;

CREATE TABLE PackageActivity

(

PackageID INTEGER NOT NULL ,

DAY VARCHAR2 (8) NOT NULL ,

Status VARCHAR2 (10) NOT NULL ,

ActivityNo INTEGER NOT NULL

) ;

ALTER TABLE PackageActivity ADD CONSTRAINT PackageActivities\_PK PRIMARY KEY ( DAY, PackageID, ActivityNo ) ;

CREATE TABLE PackageDays

(

PackageID INTEGER NOT NULL ,

DAY VARCHAR2 (8) NOT NULL ,

TravelDetail VARCHAR2 (250) NOT NULL ,

TravelMode VARCHAR2 (50) NOT NULL ,

DifficultyLevel VARCHAR2 (10) NOT NULL

) ;

ALTER TABLE PackageDays ADD CONSTRAINT PackageDays\_PK PRIMARY KEY ( DAY, PackageID ) ;

CREATE TABLE PackageDestination

(

DestinationId INTEGER NOT NULL ,

PackageID INTEGER NOT NULL

) ;

ALTER TABLE PackageDestination ADD CONSTRAINT PackageDestination\_PK PRIMARY KEY ( PackageID, DestinationId ) ;

CREATE TABLE Staff

(

StaffId INTEGER NOT NULL ,

StaffName VARCHAR2 (50) NOT NULL ,

SPhone VARCHAR2 (10) NOT NULL ,

Salary INTEGER NOT NULL ,

DesignationId INTEGER NOT NULL

) ;

ALTER TABLE Staff ADD CONSTRAINT Staff\_PK PRIMARY KEY ( StaffId ) ;

CREATE TABLE TourGuide

(

PackageID INTEGER NOT NULL ,

StaffId INTEGER NOT NULL

) ;

ALTER TABLE TourGuide ADD CONSTRAINT TourGuide\_PK PRIMARY KEY ( PackageID, StaffId ) ;

CREATE TABLE TourPackages

(

PackageID INTEGER NOT NULL ,

PackageName VARCHAR2 (50) NOT NULL ,

TotalNoDays INTEGER NOT NULL ,

Difficulty VARCHAR2 (10) NOT NULL ,

Description VARCHAR2 (250) NOT NULL ,

Price INTEGER NOT NULL ,

StartDate DATE NOT NULL ,

EndDate DATE NOT NULL

) ;

ALTER TABLE TourPackages ADD CONSTRAINT TourPackages\_PK PRIMARY KEY ( PackageID ) ;

ALTER TABLE PackageActivity ADD CONSTRAINT ActivityNo FOREIGN KEY ( ActivityNo ) REFERENCES Activity ( ActivityNo ) ;

ALTER TABLE Booking ADD CONSTRAINT CustomerId FOREIGN KEY ( CustomerId ) REFERENCES Customer ( CustomerId ) ;

ALTER TABLE Staff ADD CONSTRAINT DesignationId FOREIGN KEY ( DesignationId ) REFERENCES Designation ( DesignationId ) ;

ALTER TABLE PackageDestination ADD CONSTRAINT DestinationId FOREIGN KEY ( DestinationId ) REFERENCES Destination ( DestinationId ) ;

ALTER TABLE PackageActivity ADD CONSTRAINT PackageActivities\_PackageDays FOREIGN KEY ( DAY, PackageID ) REFERENCES PackageDays ( DAY, PackageID ) ;

ALTER TABLE TourGuide ADD CONSTRAINT PackageId FOREIGN KEY ( PackageID ) REFERENCES TourPackages ( PackageID ) ;

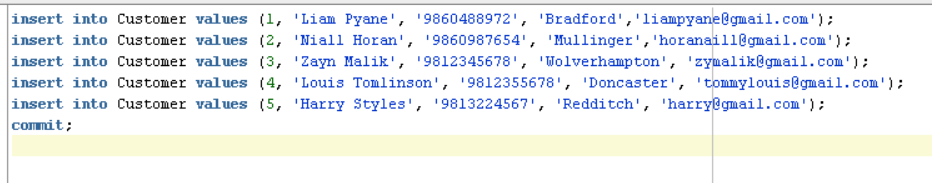
ALTER TABLE PackageDestination ADD CONSTRAINT PackageIdv2 FOREIGN KEY ( PackageID ) REFERENCES TourPackages ( PackageID ) ;

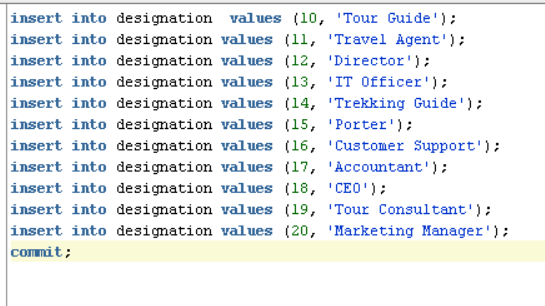
ALTER TABLE PackageDays ADD CONSTRAINT PackageIdv3 FOREIGN KEY ( PackageID ) REFERENCES TourPackages ( PackageID ) ;

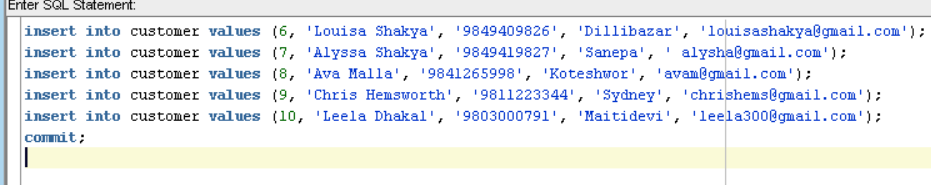
ALTER TABLE Booking ADD CONSTRAINT PackageIdv4 FOREIGN KEY ( PackageID ) REFERENCES TourPackages ( PackageID ) ;

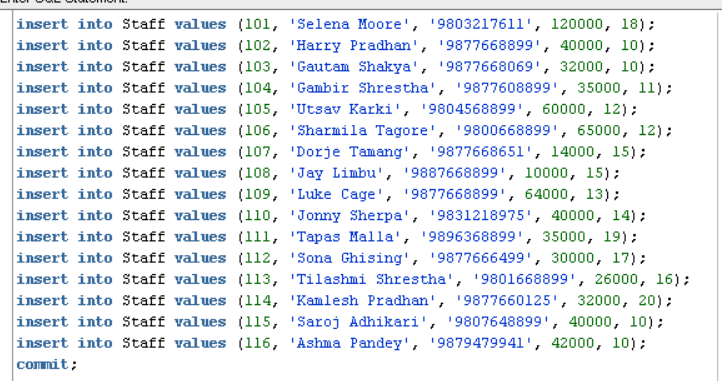
ALTER TABLE TourGuide ADD CONSTRAINT StaffId FOREIGN KEY ( StaffId ) REFERENCES Staff ( StaffId ) ;

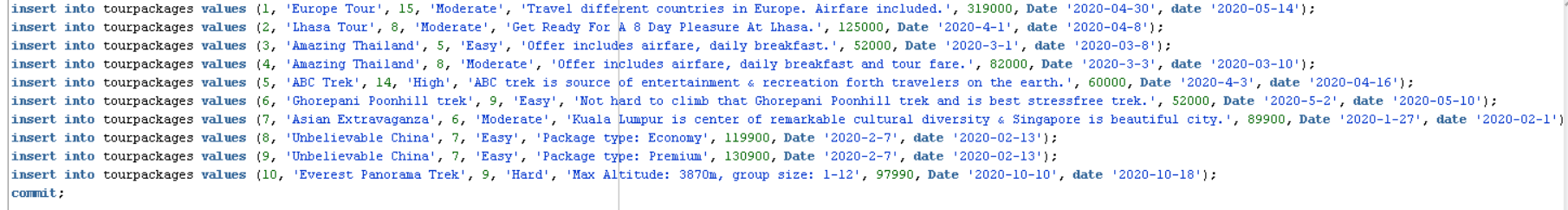
## 5.2. Insert Statements

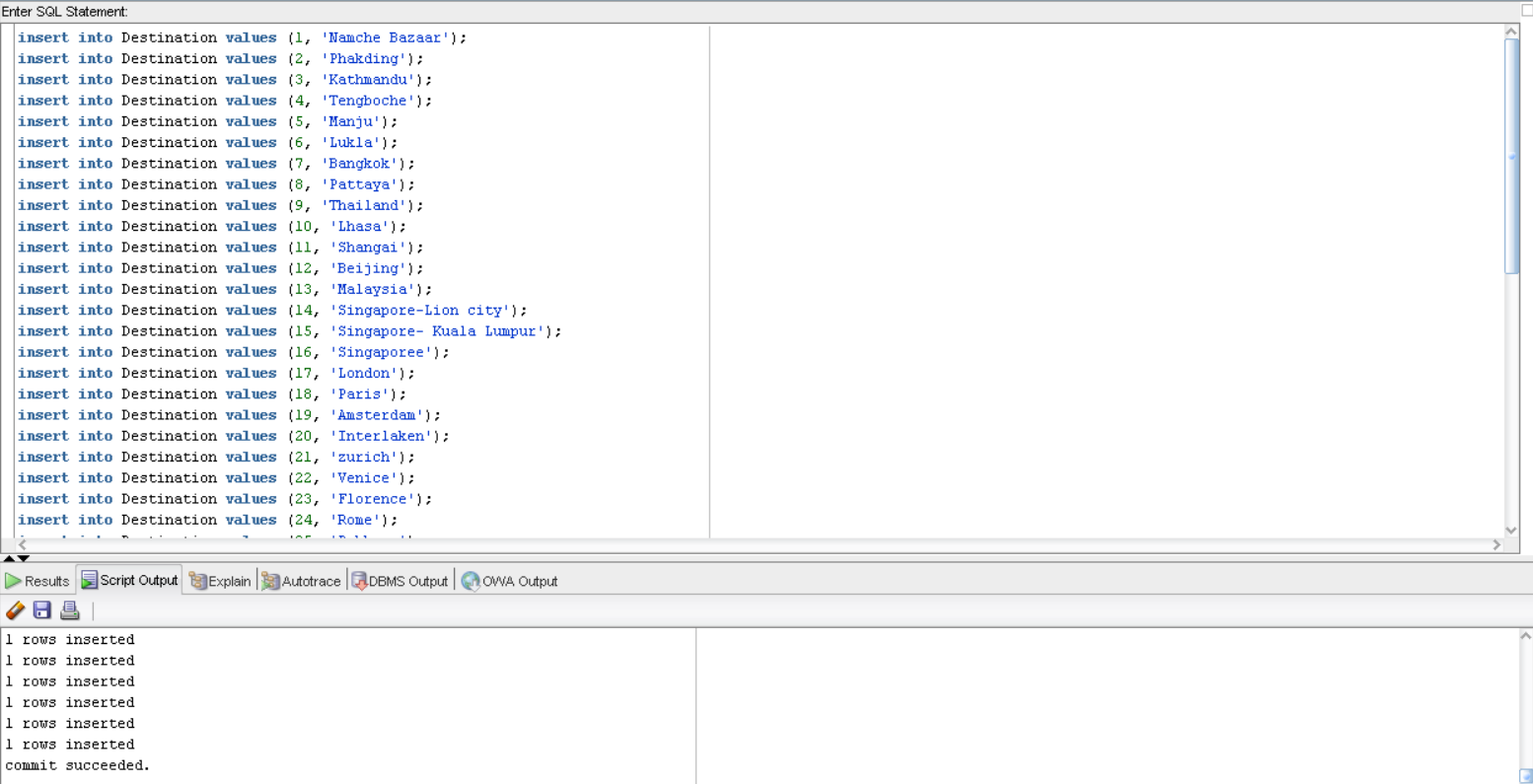


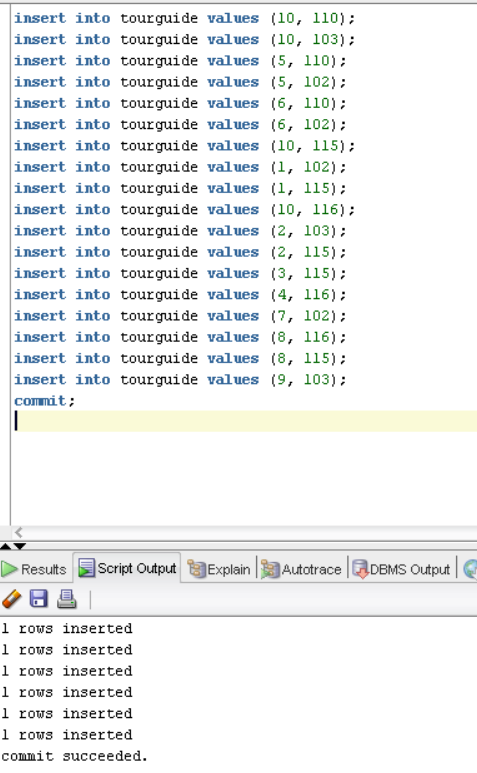


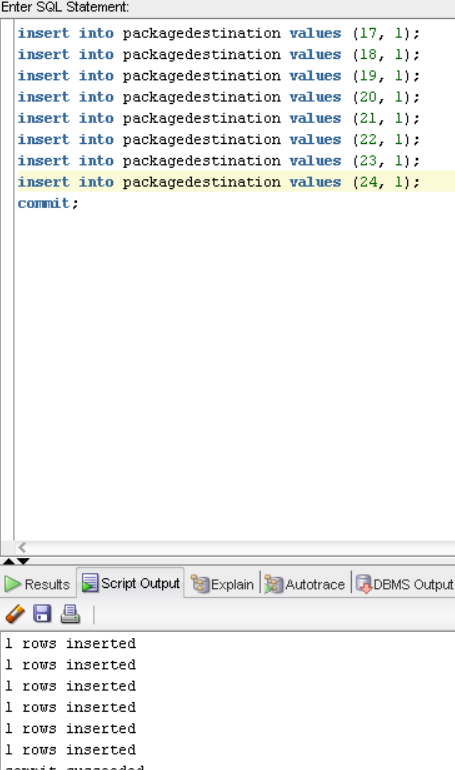


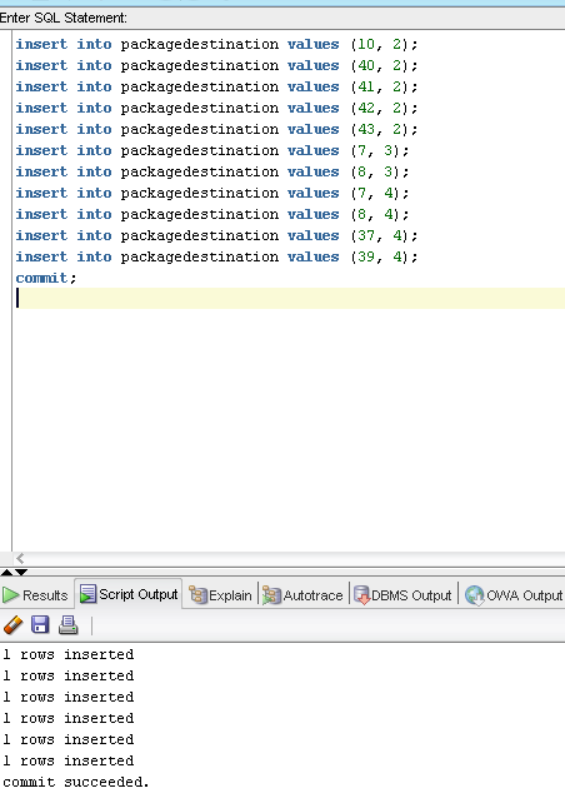


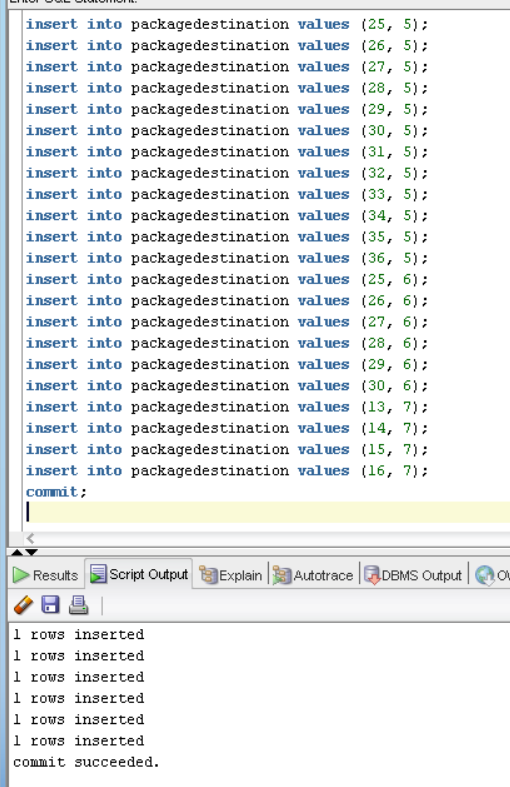


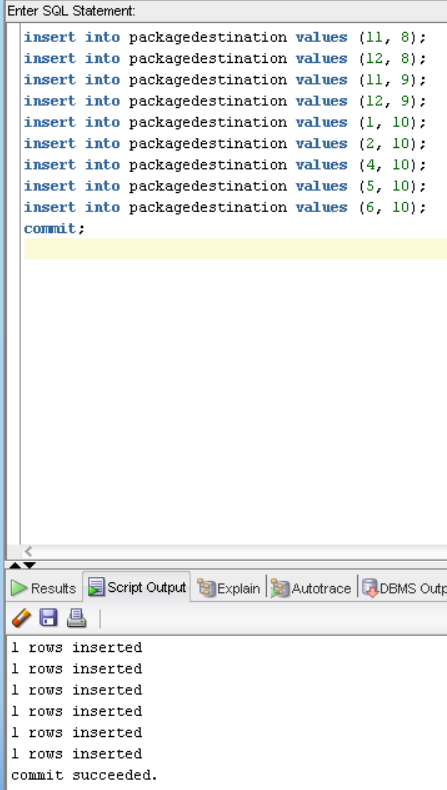




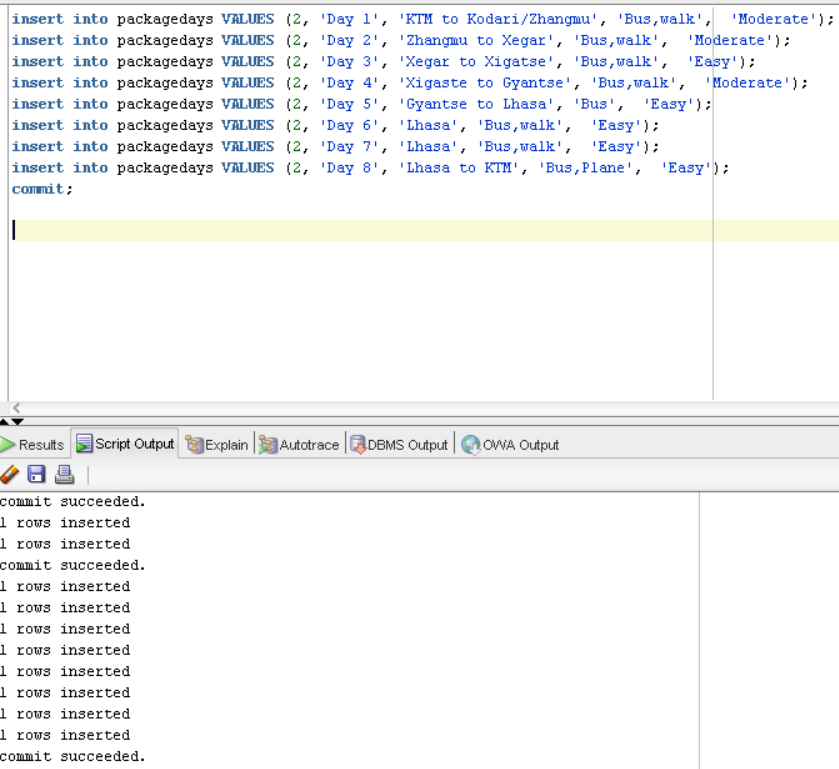


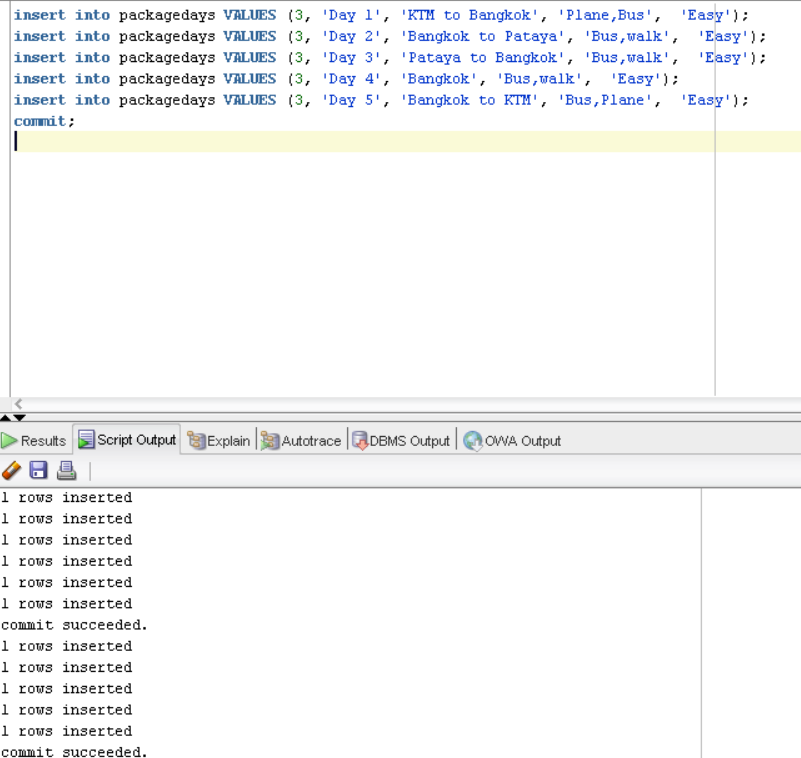




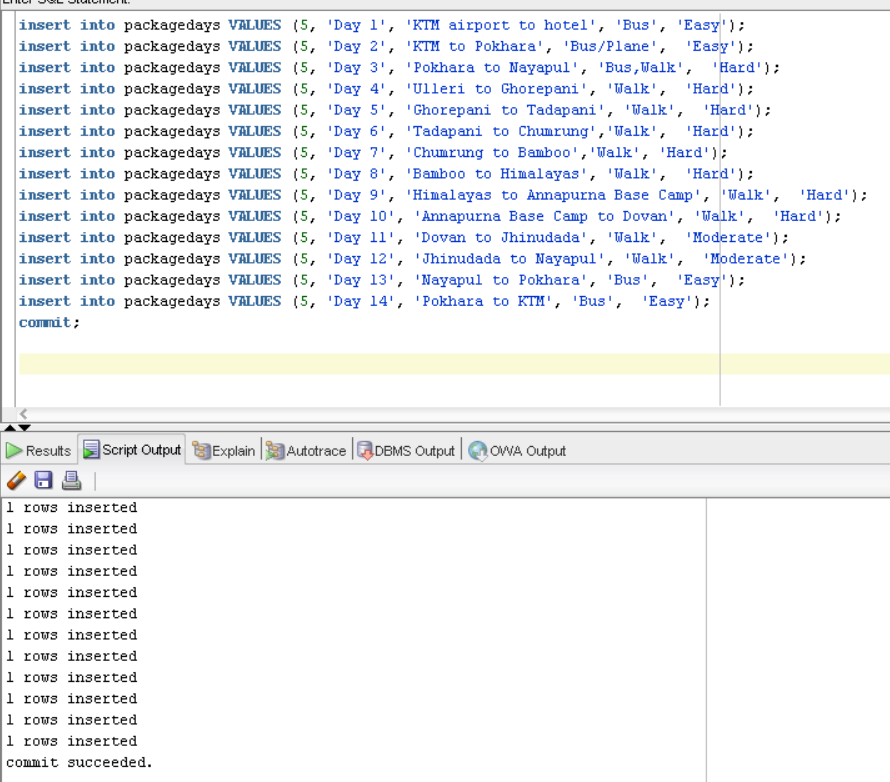


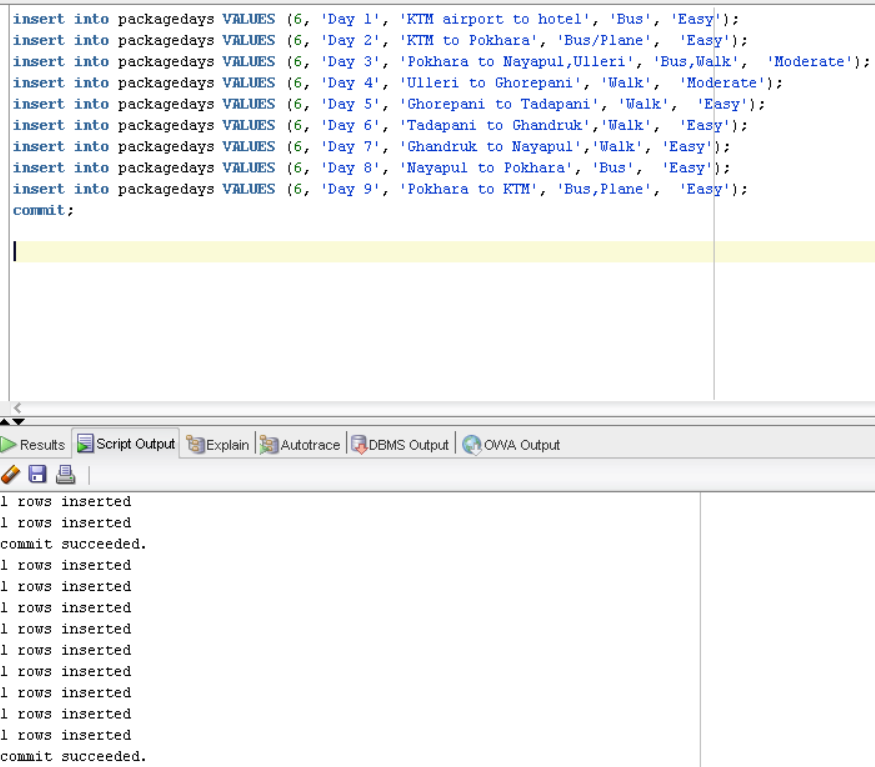


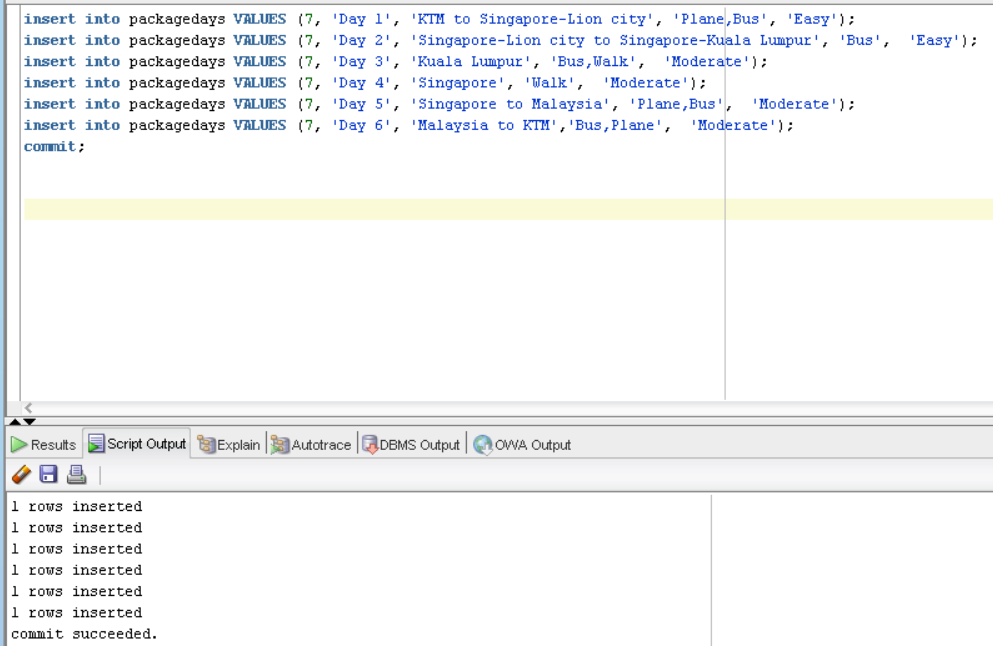




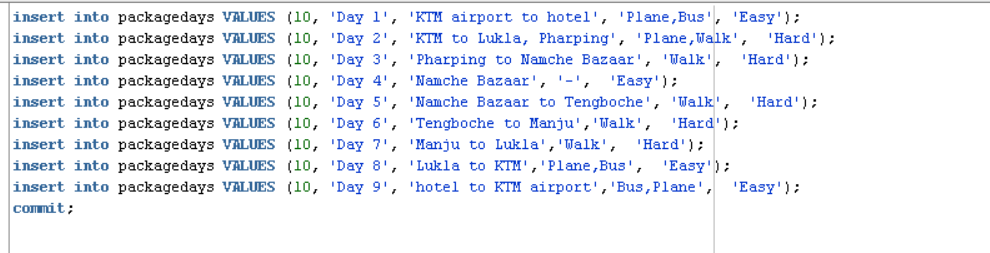


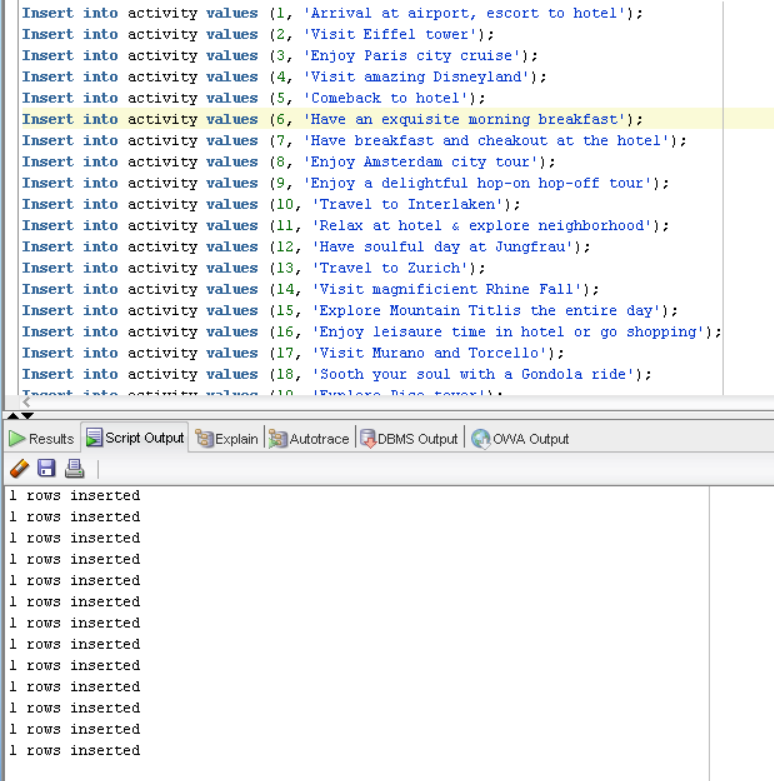


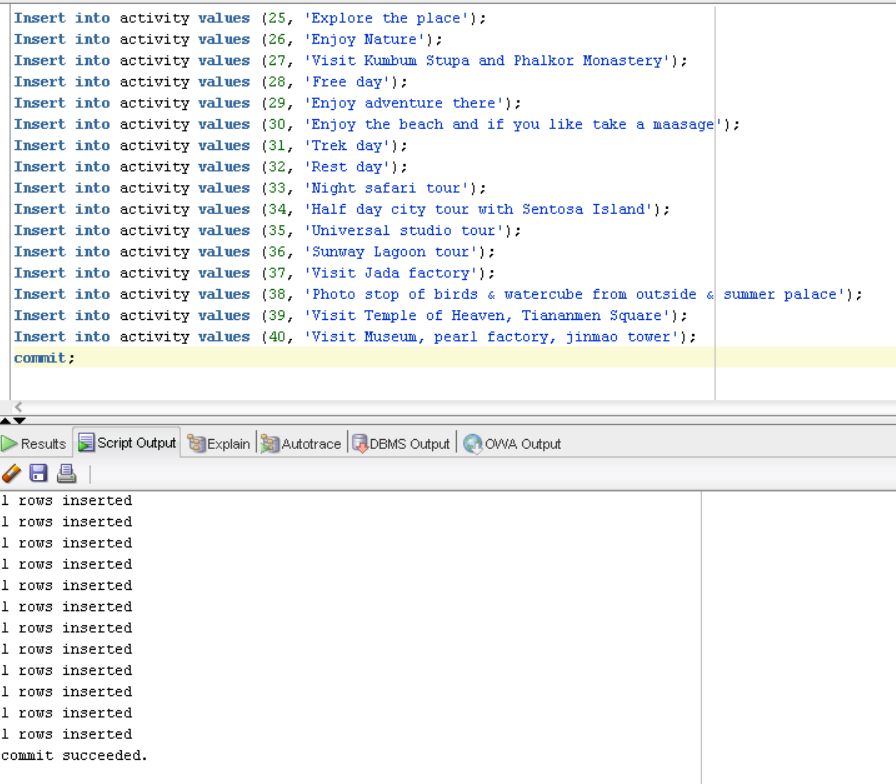


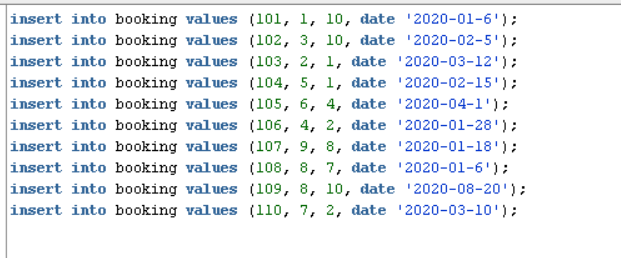


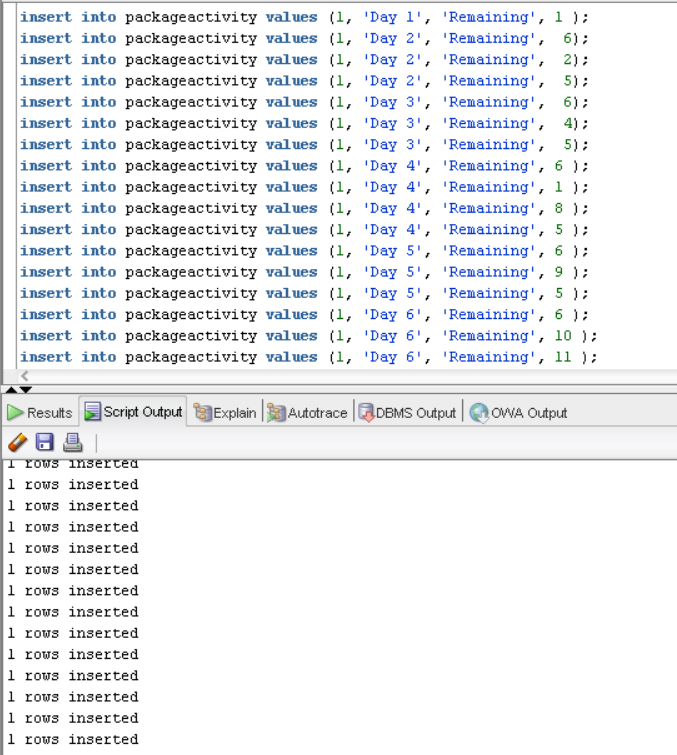












insert into packageactivity values (1, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (1, 'Day 2', 'Remaining', 6);

insert into packageactivity values (1, 'Day 2', 'Remaining', 2);

insert into packageactivity values (1, 'Day 2', 'Remaining', 5);

insert into packageactivity values (1, 'Day 3', 'Remaining', 6);

insert into packageactivity values (1, 'Day 3', 'Remaining', 4);

insert into packageactivity values (1, 'Day 3', 'Remaining', 5);

insert into packageactivity values (1, 'Day 4', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 4', 'Remaining', 1 );

insert into packageactivity values (1, 'Day 4', 'Remaining', 8 );

insert into packageactivity values (1, 'Day 4', 'Remaining', 5 );

insert into packageactivity values (1, 'Day 5', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 5', 'Remaining', 9 );

insert into packageactivity values (1, 'Day 5', 'Remaining', 5 );

insert into packageactivity values (1, 'Day 6', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 6', 'Remaining', 10 );

insert into packageactivity values (1, 'Day 6', 'Remaining', 11 );

insert into packageactivity values (1, 'Day 7', 'Remaining', 12 );

insert into packageactivity values (1, 'Day 7', 'Remaining', 5 );

insert into packageactivity values (1, 'Day 8', 'Remaining', 7 );

insert into packageactivity values (1, 'Day 8', 'Remaining', 13 );

insert into packageactivity values (1, 'Day 8', 'Remaining', 14 );

insert into packageactivity values (1, 'Day 8', 'Remaining', 16 );

insert into packageactivity values (1, 'Day 9', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 9', 'Remaining', 15);

insert into packageactivity values (1, 'Day 9', 'Remaining', 5);

insert into packageactivity values (1, 'Day 10', 'Remaining', 7 );

insert into packageactivity values (1, 'Day 10', 'Remaining',28 );

insert into packageactivity values (1, 'Day 11', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 11', 'Remaining', 17 );

insert into packageactivity values (1, 'Day 11', 'Remaining', 18 );

insert into packageactivity values (1, 'Day 12', 'Remaining', 7 );

insert into packageactivity values (1, 'Day 12', 'Remaining', 1 );

insert into packageactivity values (1, 'Day 12', 'Remaining', 19 );

insert into packageactivity values (1, 'Day 13', 'Remaining', 7 );

insert into packageactivity values (1, 'Day 13', 'Remaining', 20 );

insert into packageactivity values (1, 'Day 13', 'Remaining', 11 );

insert into packageactivity values (1, 'Day 14', 'Remaining', 6 );

insert into packageactivity values (1, 'Day 14', 'Remaining', 21 );

insert into packageactivity values (1, 'Day 14', 'Remaining', 22 );

insert into packageactivity values (1, 'Day 14', 'Remaining', 23 );

insert into packageactivity values (1, 'Day 15', 'Remaining', 7);

insert into packageactivity values (1, 'Day 15', 'Remaining', 24 );

insert into packageactivity values (2, 'Day 1', 'Remaining', 24 );

insert into packageactivity values (2, 'Day 1', 'Remaining', 25 );

insert into packageactivity values (2, 'Day 2', 'Remaining', 26 );

insert into packageactivity values (2, 'Day 2', 'Remaining', 28);

insert into packageactivity values (2, 'Day 3', 'Remaining', 23);

insert into packageactivity values (2, 'Day 3', 'Remaining', 26 );

insert into packageactivity values (2, 'Day 4', 'Remaining', 27 );

insert into packageactivity values (2, 'Day 5', 'Remaining', 28 );

insert into packageactivity values (2, 'Day 6', 'Remaining',25 );

insert into packageactivity values (2, 'Day 7', 'Remaining', 16 );

insert into packageactivity values (2, 'Day 8', 'Remaining', 24);

insert into packageactivity values (3, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (3, 'Day 1', 'Remaining', 11 );

insert into packageactivity values (3, 'Day 2', 'Remaining', 6 );

insert into packageactivity values (3, 'Day 2', 'Remaining', 29 );

insert into packageactivity values (3, 'Day 2', 'Remaining', 30 );

insert into packageactivity values (3, 'Day 3', 'Remaining', 24 );

insert into packageactivity values (3, 'Day 3', 'Remaining', 1 );

insert into packageactivity values (3, 'Day 3', 'Remaining', 28 );

insert into packageactivity values (3, 'Day 4', 'Remaining', 11 );

insert into packageactivity values (3, 'Day 5', 'Remaining', 7 );

insert into packageactivity values (4, 'Day 1', 'Remaining', 11 );

insert into packageactivity values (4, 'Day 2', 'Remaining', 6 );

insert into packageactivity values (4, 'Day 2', 'Remaining', 29 );

insert into packageactivity values (4, 'Day 2', 'Remaining', 30 );

insert into packageactivity values (4, 'Day 3', 'Remaining', 24 );

insert into packageactivity values (4, 'Day 3', 'Remaining', 1 );

insert into packageactivity values (4, 'Day 3', 'Remaining', 28 );

insert into packageactivity values (4, 'Day 4', 'Remaining', 11 );

insert into packageactivity values (4, 'Day 5', 'Remaining', 7 );

insert into packageactivity values (4, 'Day 5', 'Remaining', 1 );

insert into packageactivity values (4, 'Day 5', 'Remaining', 29 );

insert into packageactivity values (4, 'Day 5', 'Remaining', 30);

insert into packageactivity values (4, 'Day 6', 'Remaining', 7 );

insert into packageactivity values (4, 'Day 6', 'Remaining', 29 );

insert into packageactivity values (4, 'Day 6', 'Remaining', 30 );

insert into packageactivity values (4, 'Day 7', 'Remaining', 7 );

insert into packageactivity values (4, 'Day 7', 'Remaining', 28 );

insert into packageactivity values (4, 'Day 8', 'Remaining', 24 );

insert into packageactivity values (5, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (5, 'Day 2', 'Remaining', 1 );

insert into packageactivity values (5, 'Day 2', 'Remaining', 32 );

insert into packageactivity values (5, 'Day 3', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 3', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 4', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 4', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 4', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 5', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 5', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 5', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 6', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 6', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 6', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 7', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 7', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 7', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 8', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 8', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 8', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 9', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 9', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 9', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 10', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 10', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 10', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 11', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 11', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 11', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 12', 'Remaining', 6 );

insert into packageactivity values (5, 'Day 12', 'Remaining', 31 );

insert into packageactivity values (5, 'Day 12', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 13', 'Remaining', 26 );

insert into packageactivity values (5, 'Day 13', 'Remaining', 28 );

insert into packageactivity values (5, 'Day 14', 'Remaining', 28 );

insert into packageactivity values (6, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (6, 'Day 2', 'Remaining', 1 );

insert into packageactivity values (6, 'Day 3', 'Remaining', 25 );

insert into packageactivity values (6, 'Day 3', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 3', 'Remaining', 32 );

insert into packageactivity values (6, 'Day 4', 'Remaining', 31 );

insert into packageactivity values (6, 'Day 4', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 4', 'Remaining', 25 );

insert into packageactivity values (6, 'Day 5', 'Remaining', 31 );

insert into packageactivity values (6, 'Day 5', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 5', 'Remaining', 25 );

insert into packageactivity values (6, 'Day 6', 'Remaining', 31 );

insert into packageactivity values (6, 'Day 6', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 6', 'Remaining', 25 );

insert into packageactivity values (6, 'Day 7', 'Remaining', 31 );

insert into packageactivity values (6, 'Day 7', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 7', 'Remaining', 25 );

insert into packageactivity values (6, 'Day 8', 'Remaining', 26 );

insert into packageactivity values (6, 'Day 9', 'Remaining', 24 );

insert into packageactivity values (7, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (7, 'Day 1', 'Remaining', 33 );

insert into packageactivity values (7, 'Day 2', 'Remaining', 6 );

insert into packageactivity values (7, 'Day 2', 'Remaining', 34 );

insert into packageactivity values (7, 'Day 3', 'Remaining', 6 );

insert into packageactivity values (7, 'Day 3', 'Remaining', 35 );

insert into packageactivity values (7, 'Day 4', 'Remaining', 7 );

insert into packageactivity values (7, 'Day 4', 'Remaining', 1 );

insert into packageactivity values (7, 'Day 4', 'Remaining', 28 );

insert into packageactivity values (7, 'Day 5', 'Remaining', 7 );

insert into packageactivity values (7, 'Day 5', 'Remaining', 1 );

insert into packageactivity values (7, 'Day 5', 'Remaining', 11);

insert into packageactivity values (7, 'Day 5', 'Remaining', 33 );

insert into packageactivity values (7, 'Day 6', 'Remaining', 36 );

insert into packageactivity values (7, 'Day 7', 'Remaining', 7 );

insert into packageactivity values (7, 'Day 7', 'Remaining', 24 );

insert into packageactivity values (8, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (8, 'Day 2', 'Remaining', 6 );

insert into packageactivity values (8, 'Day 2', 'Remaining', 37 );

insert into packageactivity values (8, 'Day 2', 'Remaining', 38 );

insert into packageactivity values (8, 'Day 3', 'Remaining', 6);

insert into packageactivity values (8, 'Day 3', 'Remaining', 39);

insert into packageactivity values (8, 'Day 3', 'Remaining', 23 );

insert into packageactivity values (8, 'Day 4', 'Remaining', 7 );

insert into packageactivity values (8, 'Day 4', 'Remaining', 1 );

insert into packageactivity values (8, 'Day 4', 'Remaining', 28 );

insert into packageactivity values (8, 'Day 5', 'Remaining', 6 );

insert into packageactivity values (8, 'Day 5', 'Remaining', 40 );

insert into packageactivity values (8, 'Day 6', 'Remaining', 28 );

insert into packageactivity values (8, 'Day 7', 'Remaining', 7 );

insert into packageactivity values (8, 'Day 7', 'Remaining', 24 );

insert into packageactivity values (9, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (9, 'Day 2', 'Remaining', 6 );

insert into packageactivity values (9, 'Day 2', 'Remaining', 37 );

insert into packageactivity values (9, 'Day 2', 'Remaining', 38 );

insert into packageactivity values (9, 'Day 3', 'Remaining', 6);

insert into packageactivity values (9, 'Day 3', 'Remaining', 39);

insert into packageactivity values (9, 'Day 3', 'Remaining', 23 );

insert into packageactivity values (9, 'Day 4', 'Remaining', 7 );

insert into packageactivity values (9, 'Day 4', 'Remaining', 1 );

insert into packageactivity values (9, 'Day 4', 'Remaining', 28 );

insert into packageactivity values (9, 'Day 5', 'Remaining', 6 );

insert into packageactivity values (9, 'Day 5', 'Remaining', 40 );

insert into packageactivity values (9, 'Day 6', 'Remaining', 28 );

insert into packageactivity values (9, 'Day 7', 'Remaining', 7 );

insert into packageactivity values (9, 'Day 7', 'Remaining', 24 );

insert into packageactivity values (10, 'Day 1', 'Remaining', 1 );

insert into packageactivity values (10, 'Day 2', 'Remaining', 1 );

insert into packageactivity values (10, 'Day 2', 'Remaining', 31 );

insert into packageactivity values (10, 'Day 2', 'Remaining', 26 );

insert into packageactivity values (10, 'Day 2', 'Remaining', 25 );

insert into packageactivity values (10, 'Day 3', 'Remaining', 31 );

insert into packageactivity values (10, 'Day 3', 'Remaining', 26 );

insert into packageactivity values (10, 'Day 3', 'Remaining', 25 );

insert into packageactivity values (10, 'Day 4', 'Remaining', 28 );

insert into packageactivity values (10, 'Day 5', 'Remaining', 31 );

insert into packageactivity values (10, 'Day 5', 'Remaining', 26 );

insert into packageactivity values (10, 'Day 5', 'Remaining', 25 );

insert into packageactivity values (10, 'Day 6', 'Remaining', 31 );

insert into packageactivity values (10, 'Day 6', 'Remaining', 26 );

insert into packageactivity values (10, 'Day 6', 'Remaining', 25 );

insert into packageactivity values (10, 'Day 7', 'Remaining', 31 );

insert into packageactivity values (10, 'Day 7', 'Remaining', 26 );

insert into packageactivity values (10, 'Day 7', 'Remaining', 25 );

insert into packageactivity values (10, 'Day 8', 'Remaining', 24 );

insert into packageactivity values (10, 'Day 8', 'Remaining', 1 );

insert into packageactivity values (10, 'Day 8', 'Remaining', 24 );

## 5.3. Select Statements

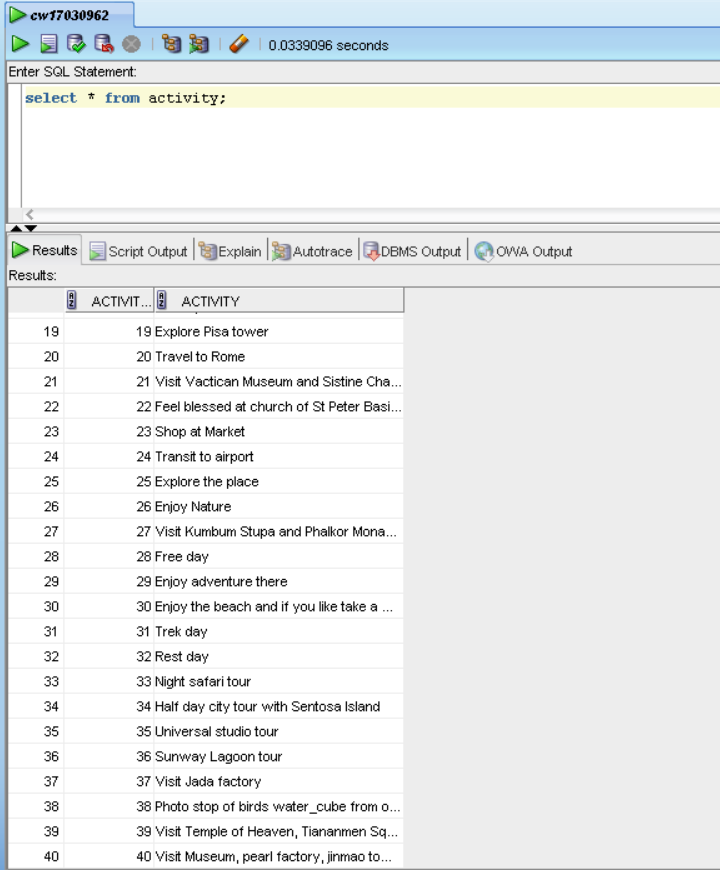


Figure 3 Select Statement for Activity

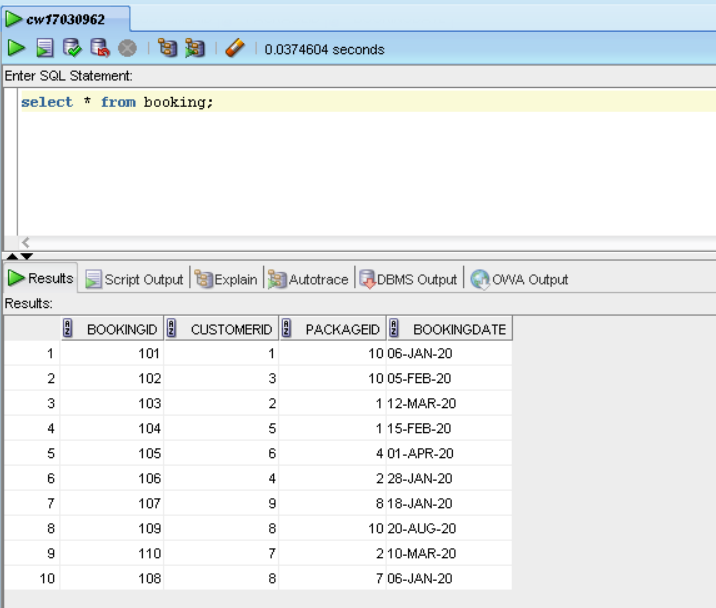


Figure 4 Select Statement for Booking

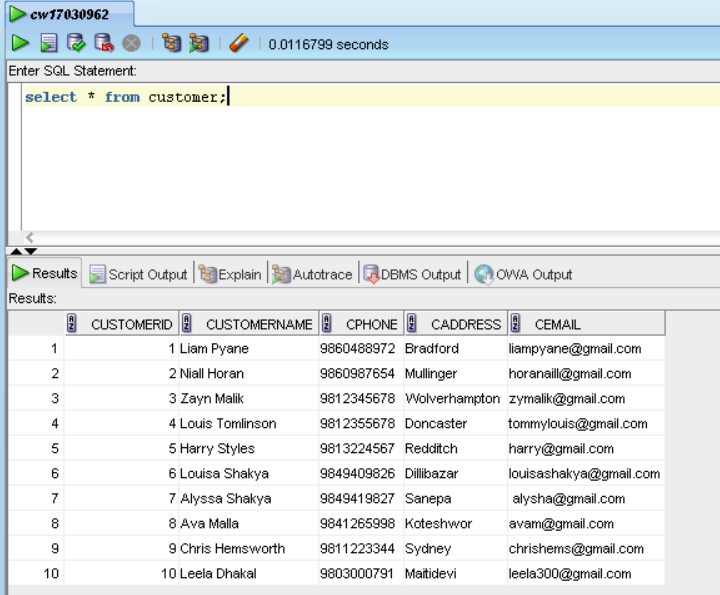


Figure 5 Select Statement for Customer

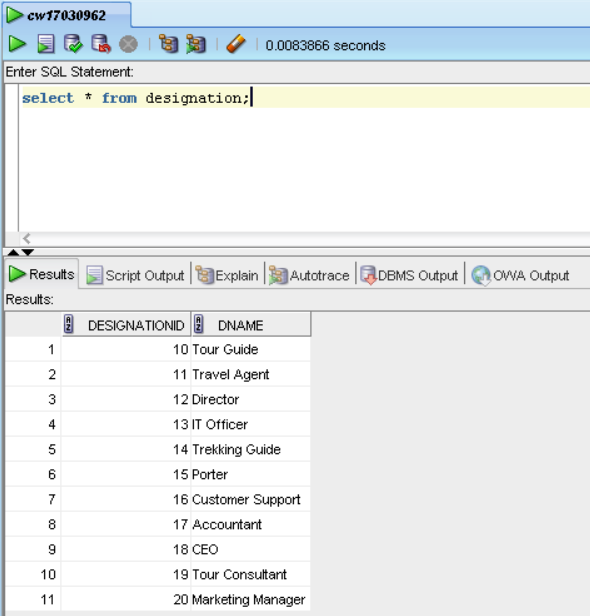


Figure 6 Select Statement for Designation

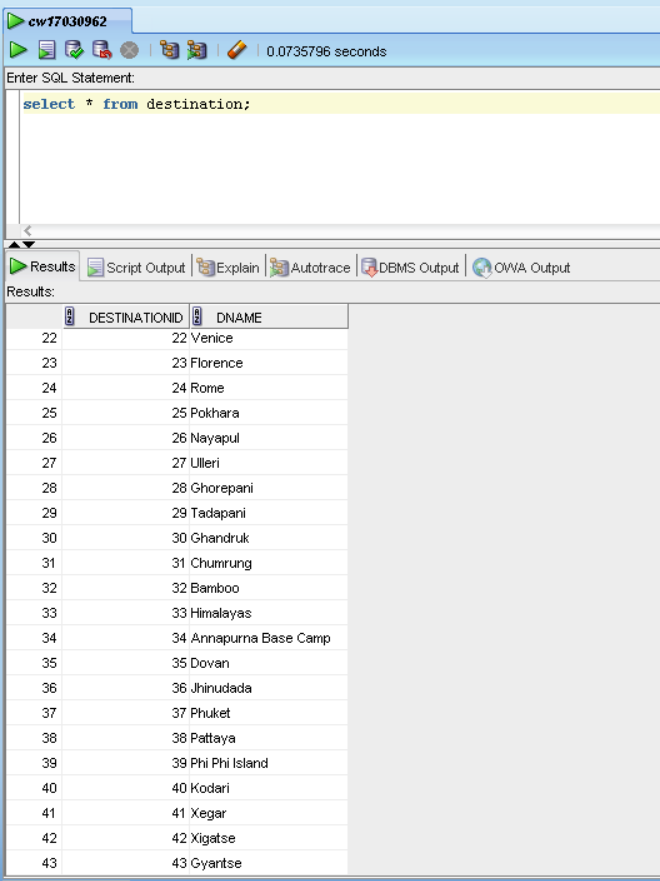


Figure 7 Select Statement for Destination

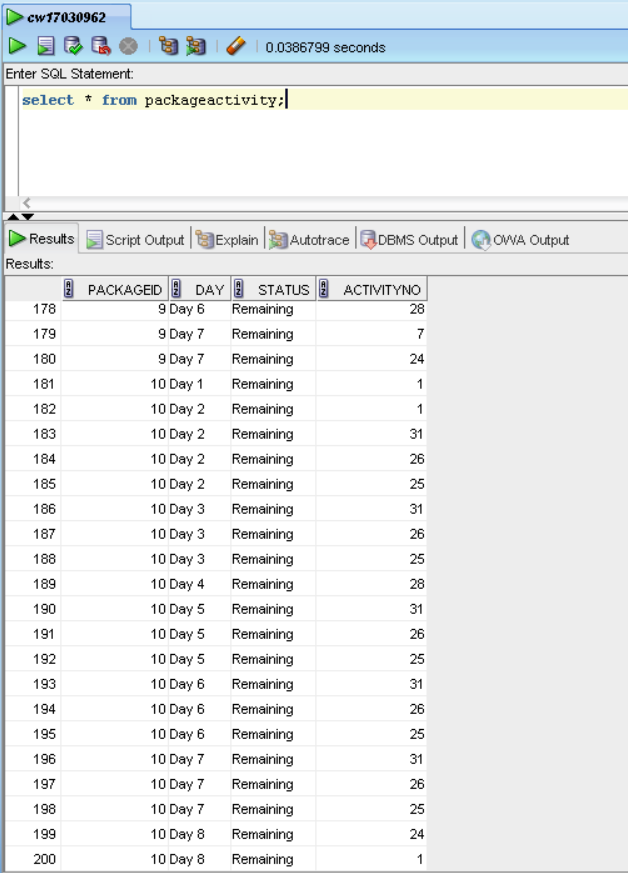


Figure 8 Select Statement for PackageActivity

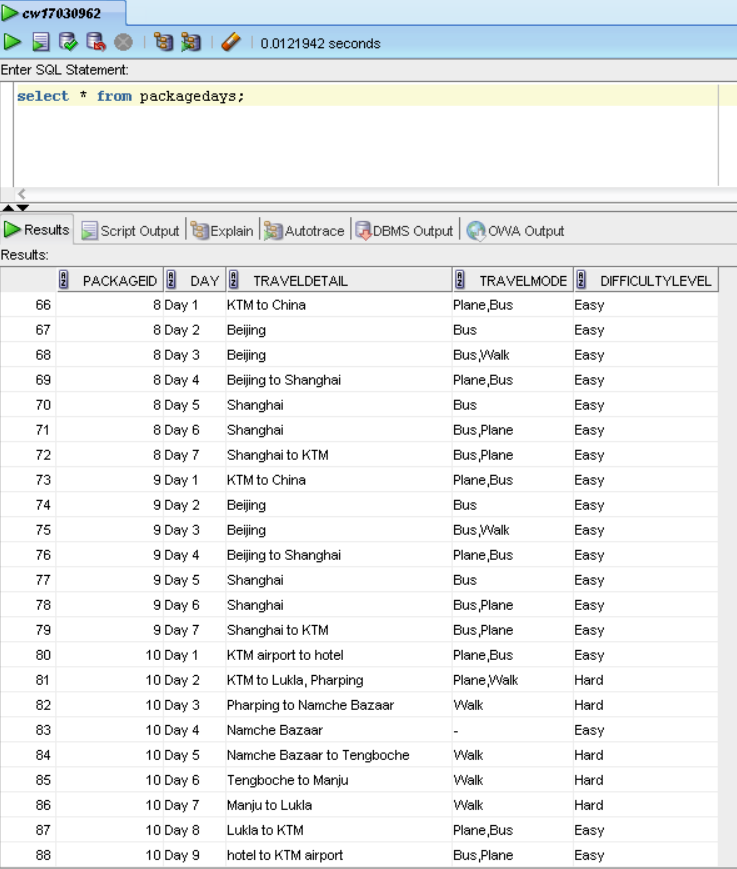


Figure 9 Select Statement for PackageDays

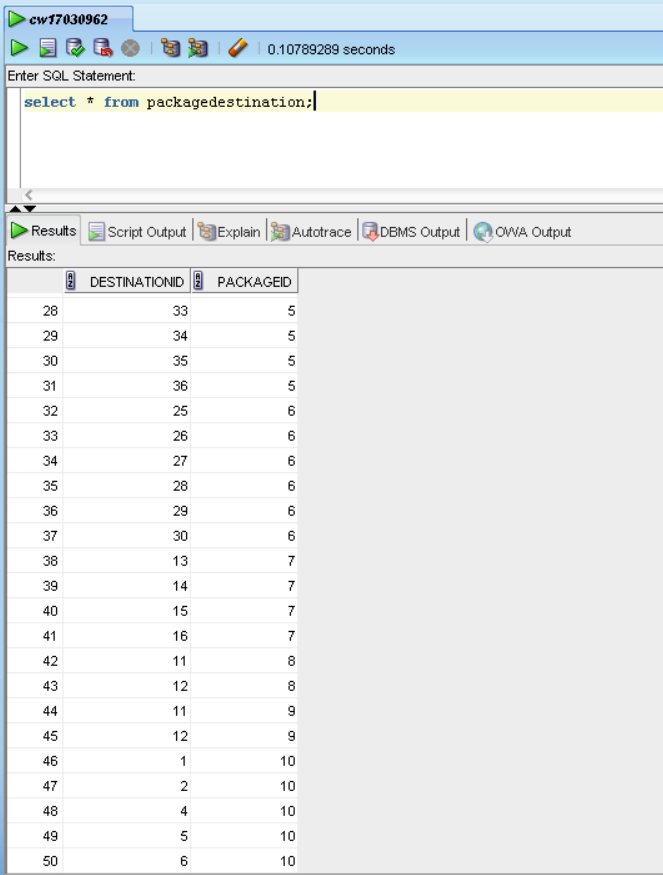


Figure 10 Select Statement for PackageDestination

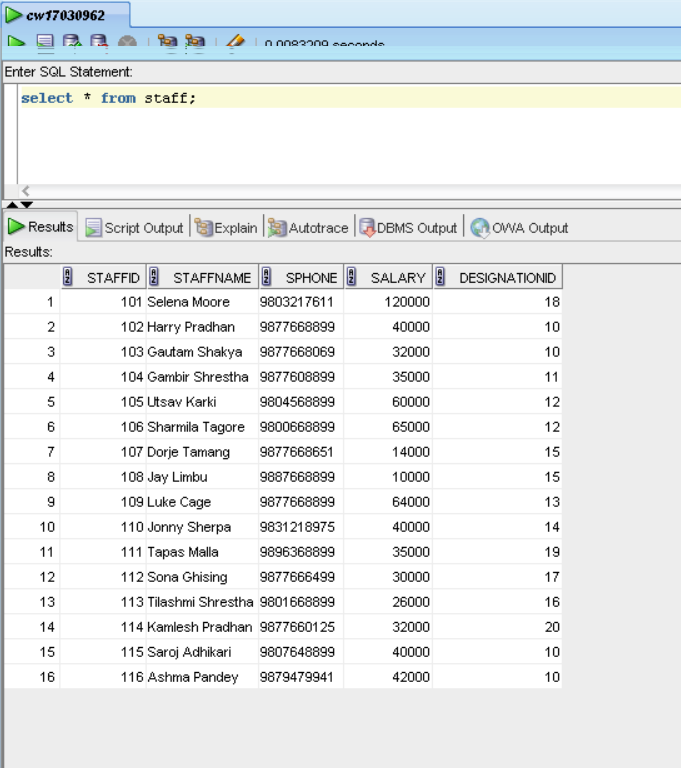


Figure 11 Select Statement for Staff

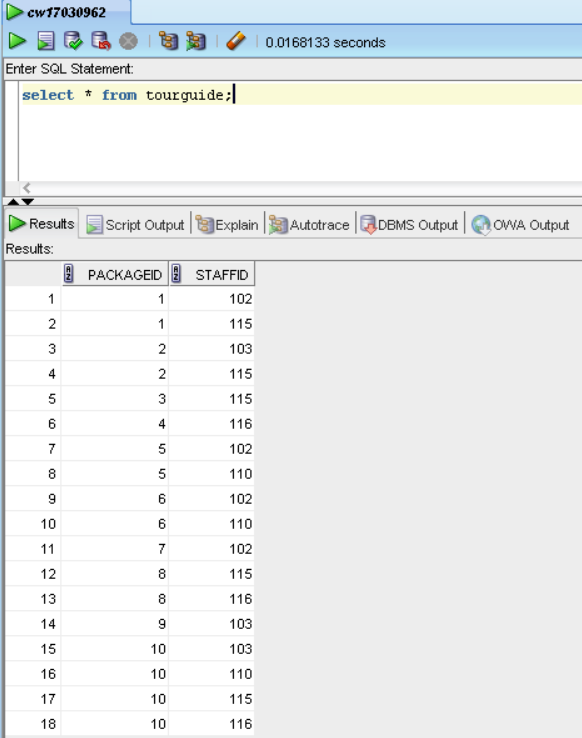


Figure 12 Select Statement for TourGuide

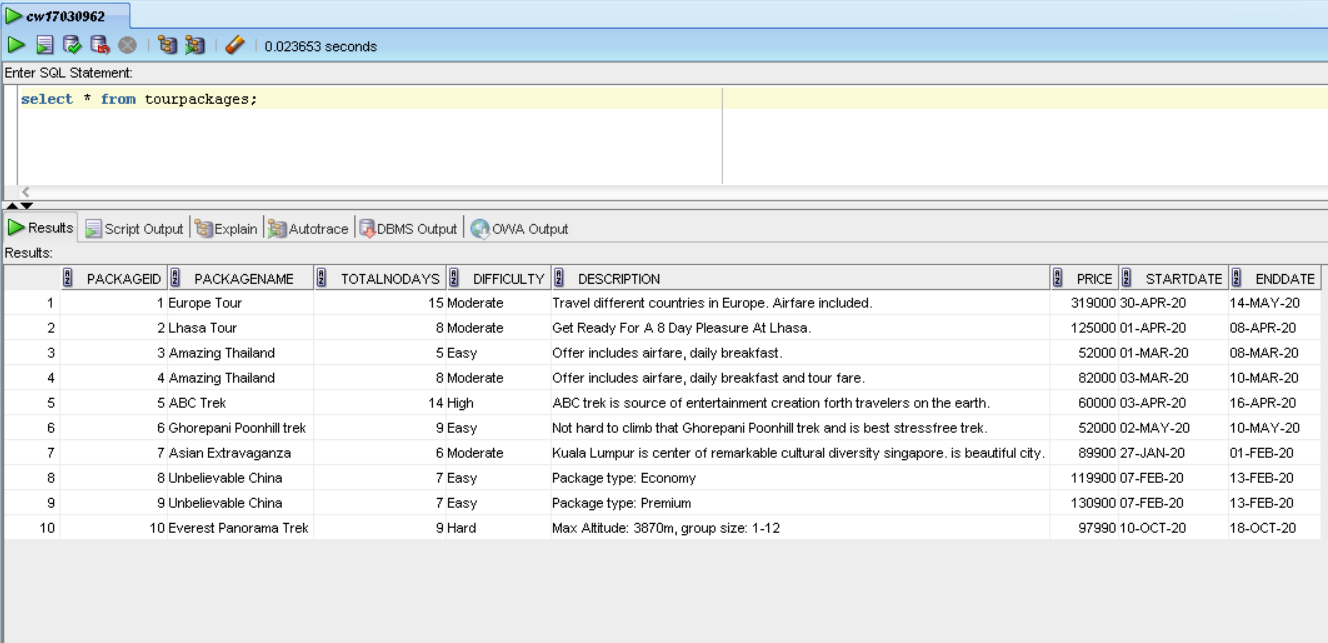


Figure 13 Select Statement for TourPackages

# 6. Implementation of Web-Based Database Application

## 6.1. Basic Webforms

### 6.1.1. Staff Details

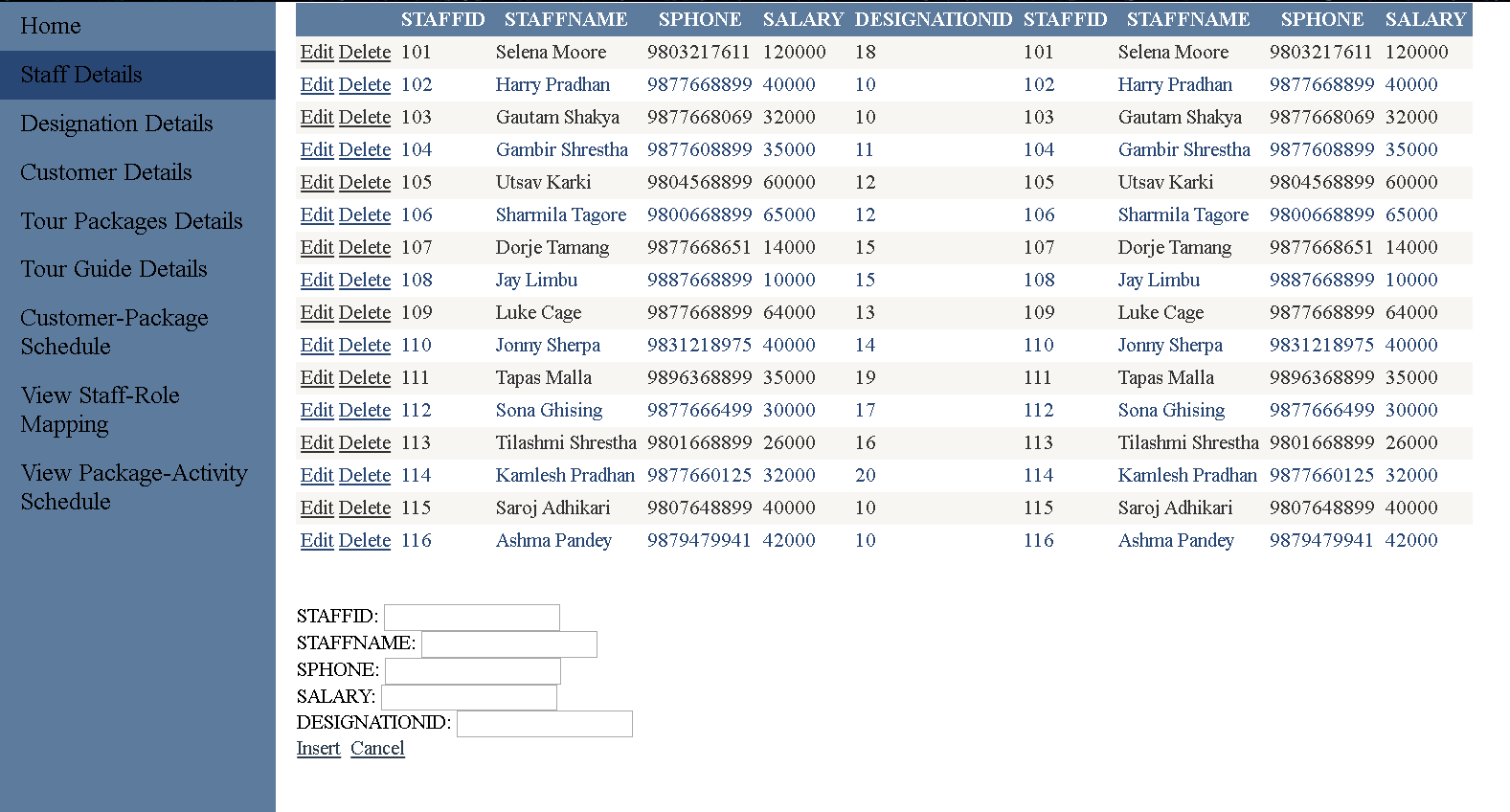


Figure 14 Staff Detail Page

### 6.1.2. Customer Details



Figure 15 Customer Details Page

### 6.1.3. Package Details

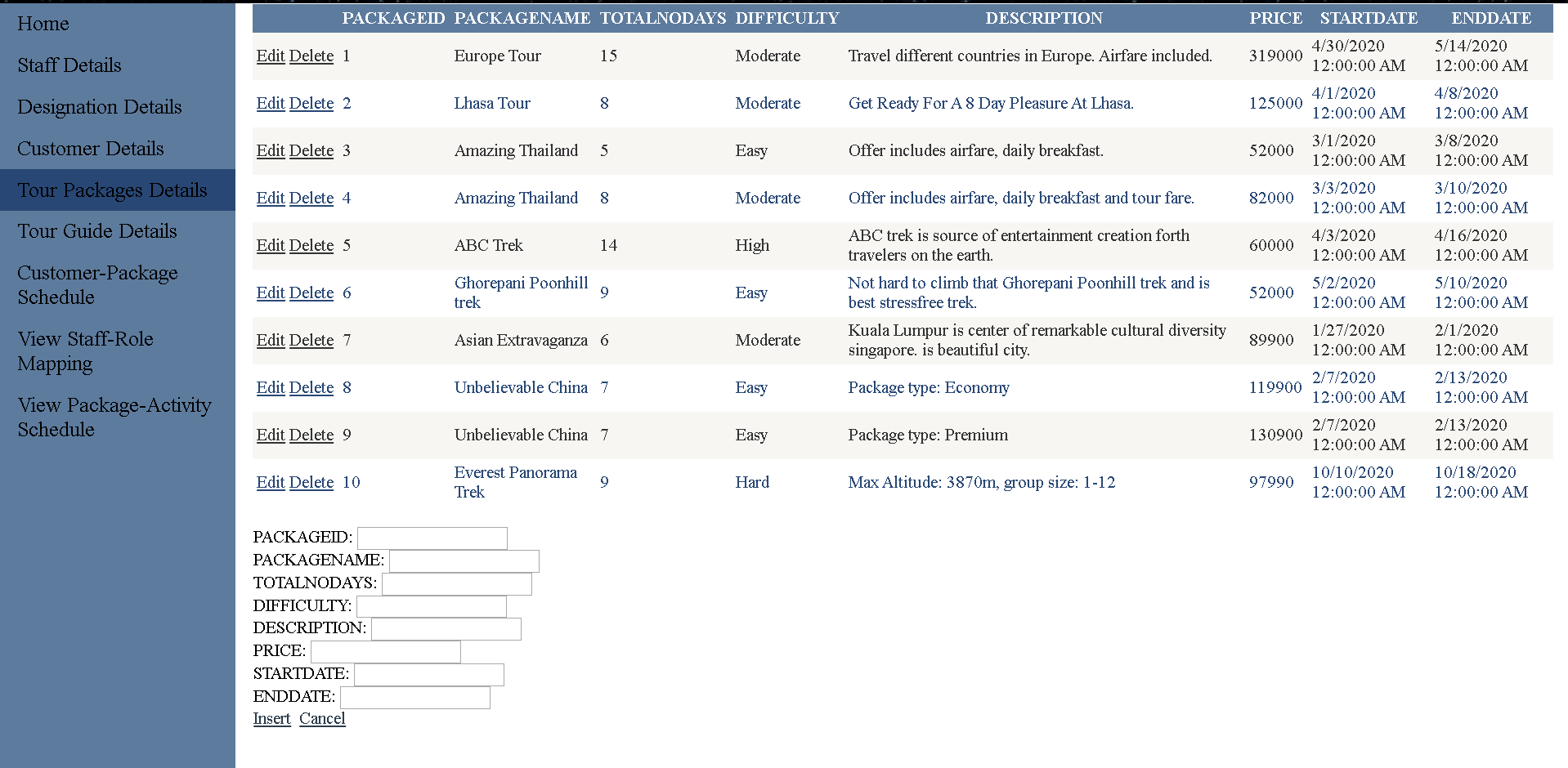


Figure 16 Tour Packages Details Page

### 6.1.4. Tour Guide Details

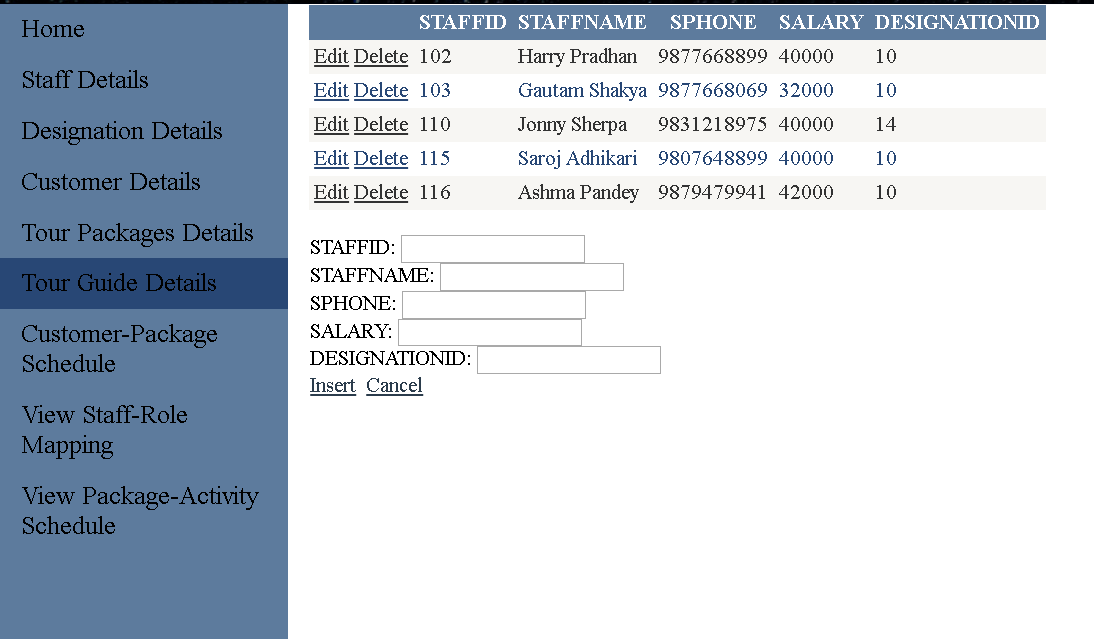


Figure 17 Tour Guide Details Page

### 6.1.5. Designation Details

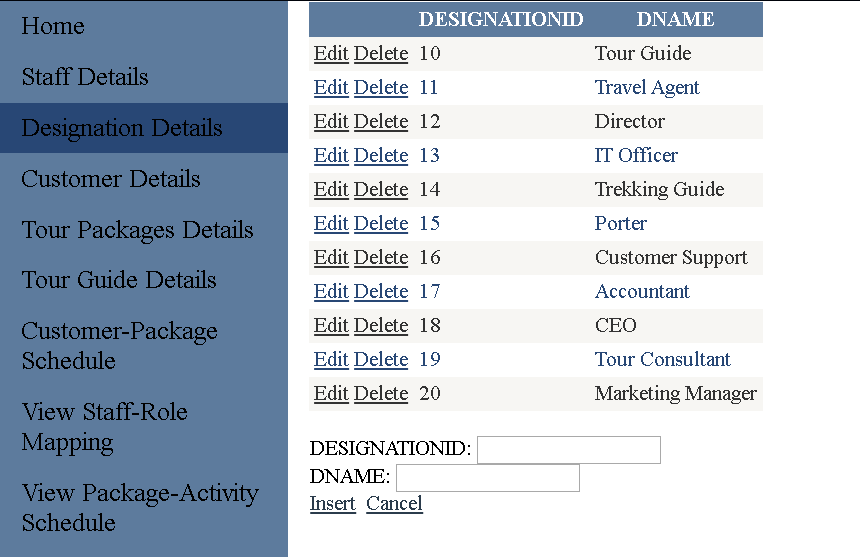


Figure 18 Designation Details Page

## 6.2. Customer-Package Schedule Form

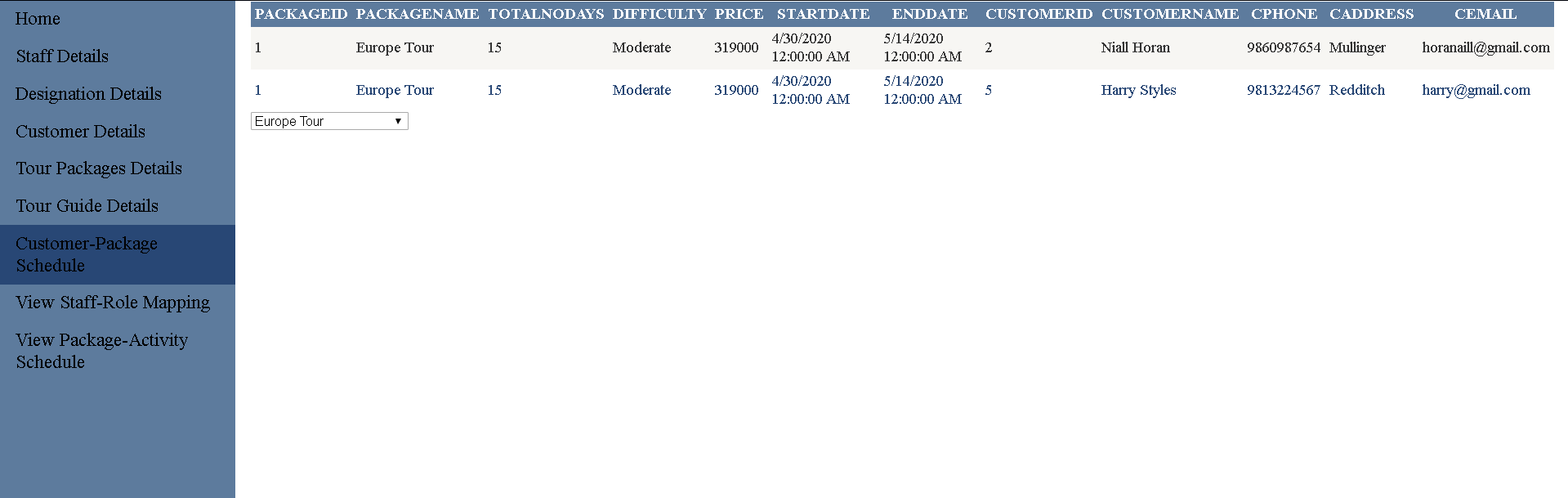


Figure 19 Customer-Package Schedule Page

## 6.3. Staff-Role Schedule Form

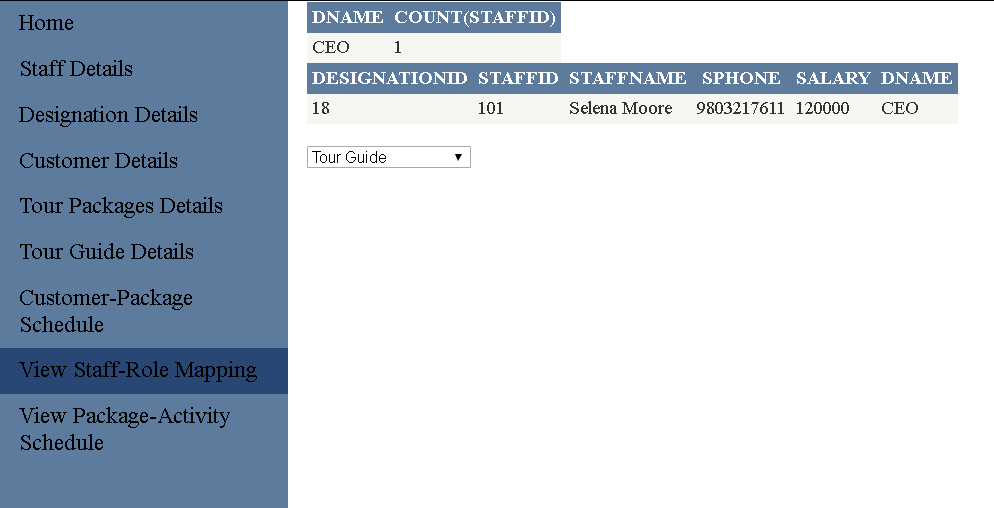


Figure 20 View Staff-Role Mapping Page

## 6.4. Package-Activity Schedule Form



Figure 21 View Package-Activity Schedule

# 7. Testing

# 8. User Manual

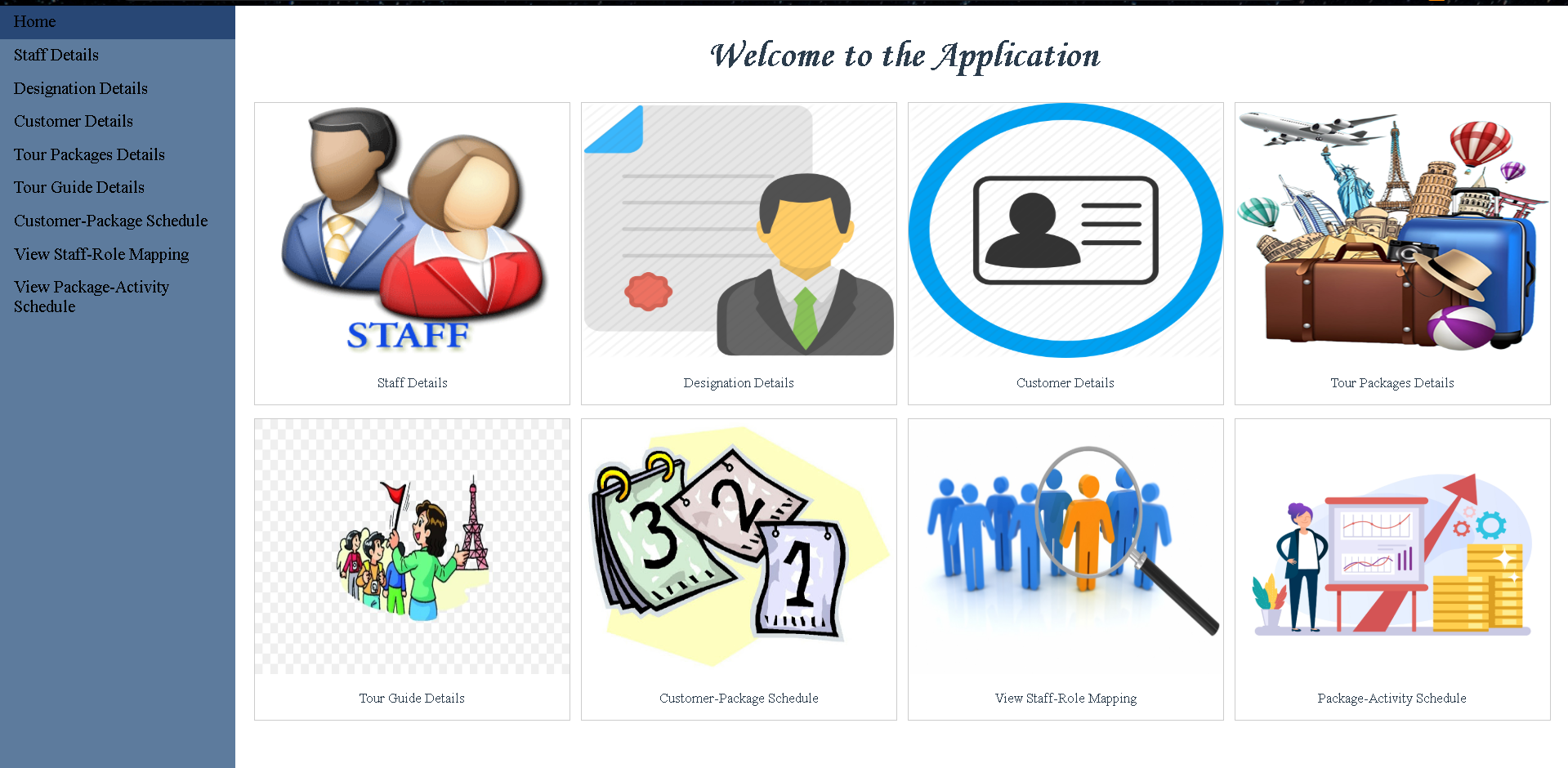


Figure 22 Home Page

The above figure is the Home page, user can select tab from the navigation or select

the page according to the pictures and text shown in home page.

# 9. Further Discussion

The given coursework is advantageous for it is helpful for forthcoming studies and also for future projects as database is crucial and is important link in the creation as well for management of data. Database is a flexible module as it can convey any fields in IT and also it helps to understand other modules too.

After the completion of the coursework I have learned how database system can be developed and how important all the small processes are before developing the database system. Analyze, design and developing a database system has been cleared by this coursework.

# 10. References

Coronel, C. & Morris, S. (2018) *Database Systems: Design, Implementation, & Management*. Cengage Learning. Available at: <https://books.google.com.np/books?id=j69EDwAAQBAJ&printsec=frontcover&dq=what+is+normalization+in+database&hl=en&sa=X&ved=0ahUKEwjxj82yuOrfAhXJK48KHYlyBiYQ6AEIMjAC#v=onepage&q=normalization&f=false> [accessed 28 January 2020].

Coronel, C. & Morris, S. (2018) *Database Systems: Design, Implementation, & Management*. Cengage Learning. Available at: <https://books.google.com.np/books?id=j69EDwAAQBAJ&printsec=frontcover&dq=what+is+normalization+in+database&hl=en&sa=X&ved=0ahUKEwjxj82yuOrfAhXJK48KHYlyBiYQ6AEIMjAC#v=onepage&q=normalization&f=false> [accessed 8 January 2019].

KERDPRASOP, & KERDPRASOP , N. (2011) Functional Dependency Discovery via Bayes Net Analysis., 2011. Research Gate.

Nishadha. (2017) *Ultimate Guide to ER Diagrams ( Entity Relationship Diagrams )* [Online]. Available from: <https://creately.com/blog/diagrams/er-diagrams-tutorial/> [Accessed 28 January 2020].

Oracle. (2020) *Data Modeling with Oracle SQL Developer* [Online]. Available from: <https://www.oracle.com/in/database/technologies/appdev/datamodeler.html> [Accessed 28 January 2020].

# 11. Appendix