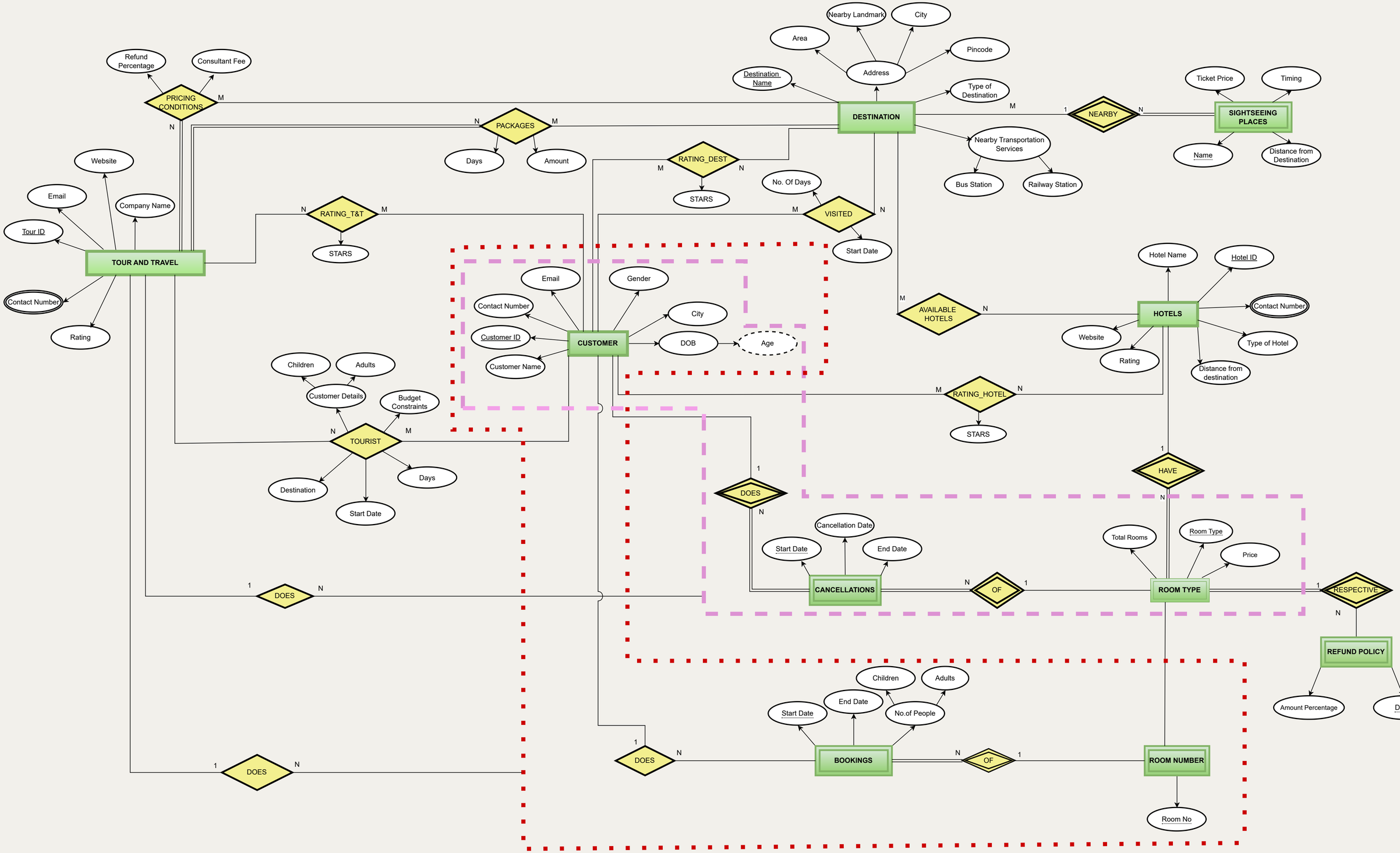
IT-214

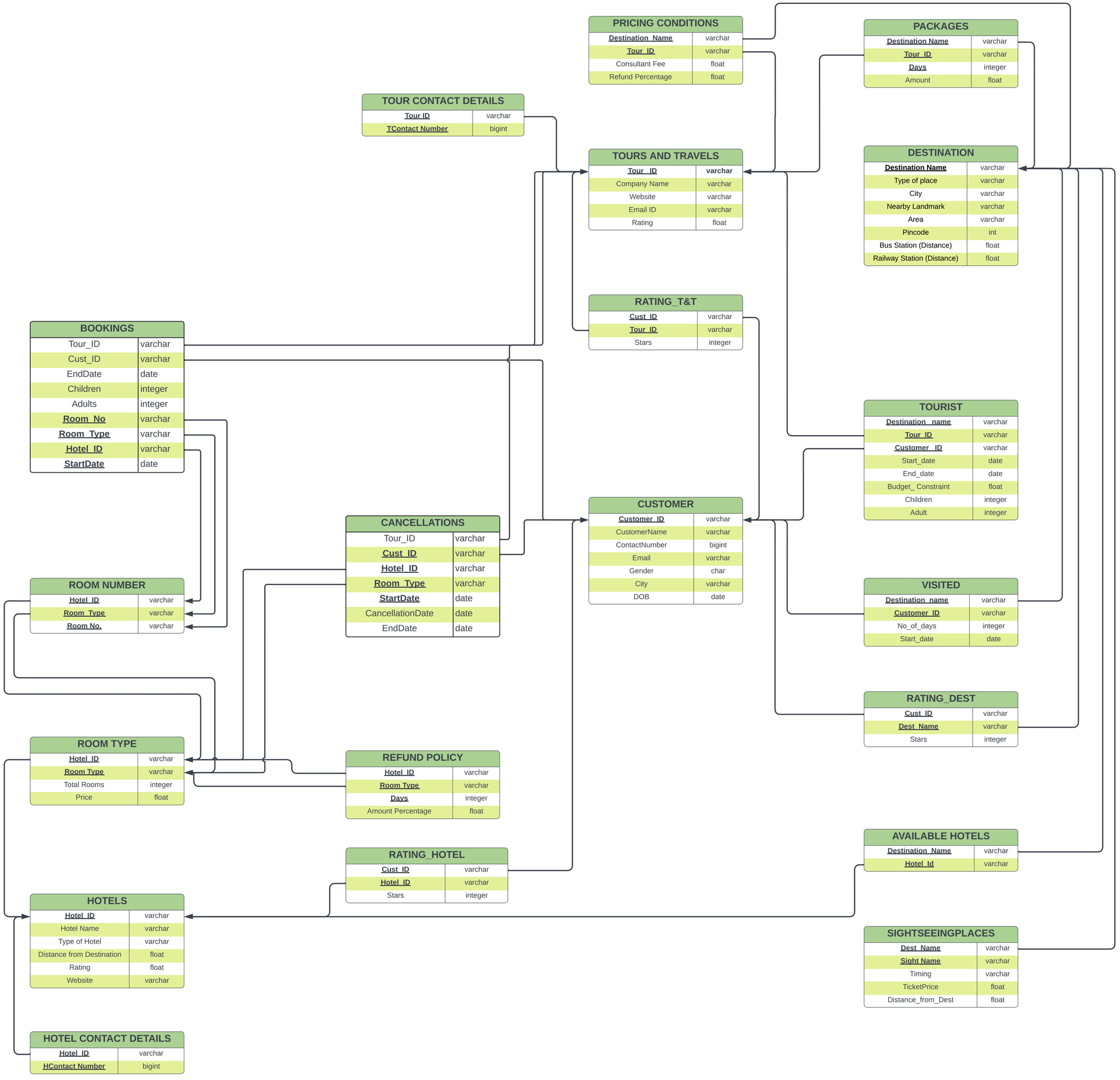
Database Management System

PROJECT TITLE:

“TOURISM GUIDE DATABASE”

Student ID	Name
202201017	Om Lachake (Representative)
202201041	Dhriti Goenka
202201060	Chintan Bhara
202201071	Nirva Patel
202201076	Vidhi Dhanani


Refund Percentage
Consultant Fee
PRICING CONDITIONS
M
N
Website
Email
Company Name
Tour ID
TOUR AND TRAVEL
Contact Number
Rating
N
PACKAGES
M
Days
Amount
N
RATING_T&T
M
STARS
Nearby Landmark
City
Area
Pincode
Address
Destination Name
DESTINATION
Type of Destination
M
Nearby Transportation Services
Bus Station
Railway Station
1
NEARBY
N
Ticket Price
Timing
SIGHTSEEING PLACES
Name
Distance from Destination
M
RATING_DEST
N
STARS
No. Of Days
M
VISITED
N
Start Date
Email
Gender
Contact Number
City
Customer ID
CUSTOMER
DOB
Age
Customer Name
Children
Adults
Customer Details
Budget Constraints
N
TOURIST
M
Destination
Days
Start Date
1
DOES
N
1
DOES
N
1
DOES
N
M
AVAILABLE HOTELS
N
Hotel Name
Hotel ID
HOTELS
Contact Number
Website
Type of Hotel
Rating
Distance from destination
M
RATING_HOTEL
N
STARS
1
HAVE
N
Cancellation Date
Start Date
End Date
CANCELLATIONS
N
OF
1
Total Rooms
Room Type
Price
ROOM TYPE
1
RESPECTIVE
N
REFUND POLICY
Amount Percentage
Days
Children
Adults
End Date
Start Date
No.of People
BOOKINGS
N
OF
1
ROOM NUMBER
Room No


PRICING CONDITIONS
Destination_Name varchar
Tour_ID varchar
Consultant Fee float
Refund Percentage float
PACKAGES
Destination Name varchar
Tour_ID varchar
Days integer
Amount float
TOUR CONTACT DETAILS
Tour ID varchar
TContact Number bigint
TOURS AND TRAVELS
Tour_ID varchar
Company Name varchar
Website varchar
Email ID varchar
Rating float
DESTINATION
Destination Name varchar
Type of place varchar
City varchar
Nearby Landmark varchar
Area varchar
Pincode int
Bus Station (Distance) float
Railway Station (Distance) float
RATING_T&T
Cust_ID varchar
Tour_ID varchar
Stars integer
BOOKINGS
Tour_ID varchar
Cust_ID varchar
EndDate date
Children integer
Adults integer
Room_No varchar
Room_Type varchar
Hotel_ID varchar
StartDate date
TOURIST
Destination_name varchar
Tour_ID varchar
Customer_ID varchar
Start_date date
End_date date
Budget_ Constraint float
Children integer
Adult integer
CANCELLATIONS
Tour_ID varchar
Cust_ID varchar
Hotel_ID varchar
Room_Type varchar
StartDate date
CancellationDate date
EndDate date
CUSTOMER
Customer_ID varchar
CustomerName varchar
ContactNumber bigint
Email varchar
Gender char
City varchar
DOB date
ROOM NUMBER
Hotel_ID varchar
Room_Type varchar
Room No. varchar
VISITED
Destination_name varchar
Customer_ID varchar
No_of_days integer
Start_date date
RATING_DEST
Cust_ID varchar
Dest_Name varchar
Stars integer
ROOM TYPE
Hotel_ID varchar
Room Type varchar
Total Rooms integer
Price float
REFUND POLICY
Hotel_ID varchar
Room Type varchar
Days integer
Amount Percentage float
AVAILABLE HOTELS
Destination_Name varchar
Hotel_Id varchar
RATING_HOTEL
Cust_ID varchar
Hotel_ID varchar
Stars integer
HOTELS
Hotel_ID varchar
Hotel Name varchar
Type of Hotel varchar
Distance from Destination float
Rating float
Website varchar
SIGHTSEEINGPLACES
Dest_Name varchar
Sight Name varchar
Timing varchar
TicketPrice float
Distance_from_Dest float
HOTEL CONTACT DETAILS
Hotel_ID varchar
HContact Number bigint

FUNCTIONAL DEPENDENCIES AND NORMALIZATION

FUNCTIONAL DEPENDENCY SET:

Customer_ID → CustomerName
Customer_ID → ContactNumber
Customer_ID → Email
Customer_ID → Gender
Customer_ID → City
Customer_ID → DOB
ContactNumber → Customer_ID
ContactNumber → CustomerName
ContactNumber → Email
ContactNumber → Gender
ContactNumber → City
ContactNumber → DOB
Destination_Name → Type of Place
Destination_Name → City
Destination_Name → Nearby Landmark
Destination_Name → Area
Destination_Name → Pincode
Destination_Name → Bus_Station(Distance)
Destination_Name → Railway_Station(Distance)
Pincode → Area
Pincode → City
{City, Pincode} → Area
{Area, Pincode} → City
{Nearby Landmark, Pincode} → City
{Nearby Landmark, Pincode} → Area
{Landmark, Area, City} → Pincode
{Dest_Name, Sight_Name} → Timing
{Dest_Name, Sight_Name} → TicketPrice
{Dest_Name, Sight_Name} → Distance_from_Dest
{Cust_ID, Destination_Name} → Start_Date
{Cust_ID, Destination_Name} → No_of_Days
Tour_ID → Company Name
Tour_ID → Website
Tour_ID → Email_ID
Tour_ID → Rating
{Destination_Name, Tour_ID, Customer_ID} → StartDate
{Destination_Name, Tour_ID, Customer_ID} → EndDate
{Destination_Name, Tour_ID, Customer_ID} → Budget_Constraint
{Destination_Name, Tour_ID, Customer_ID} → Children
{Destination_Name, Tour_ID, Customer_ID} → Adults
{Tour_ID, Destination_Name} → Consultant_Fee
{Tour_ID, Destination_Name} → Refund Percentage
{Tour_ID, Destination_Name, Days} → Amount

Hotel_ID → Hotel Name
Hotel_ID → Type of Hotel
Hotel_ID → Distance from Destination
Hotel_ID → Rating
Hotel_ID → Website
Website→ Hotel_ID
Website → Type of Hotel
Website → Distance from Destination
Website → Hotel Name
Website → Rating
{Hotel_ID, Room_Type} → Total Rooms
{Hotel_ID, Room_Type} → Price
{RoomType, Hotel_ID, Days} → Amount Percentage
Hotel_ID → ContactNumber
{Hotel_ID, Room_Type, Room_No, Startdate} → Cust_ID
{Hotel_ID, Room_Type, Room_No, Startdate} → EndDate
{Hotel_ID, Room_Type, Room_No, Startdate} → Tour_ID
{Hotel_ID, Room_Type, Room_No, Startdate} → Children
{Hotel_ID, Room_Type, Room_No, Startdate} → Adults
{Hotel_ID, Room_Type, Startdate, Cust_ID} → Tour_ID
{Hotel_ID, Room_Type, StartDate, Cust_ID} → CancellationDate
{Hotel_ID, Room_Type, StartDate, Cust_ID} → EndDate
{Hotel_ID, Cust_ID} → Stars
{Dest_Name, Cust_ID} → Stars
{Tour_ID, Cust_ID} → Stars
{Cust_ID, StartDate} → Hotel_ID

MINIMAL FUNCTIONAL DEPENDENCY SET:

Customer_ID → CustomerName
Customer_ID → ContactNumber
Customer_ID → Email
Customer_ID → Gender
Customer_ID → City
Customer_ID → DOB
ContactNumber → Customer_ID
ContactNumber → CustomerName
ContactNumber → Email
ContactNumber → Gender
ContactNumber → City
ContactNumber → DOB
Destination_Name → Type of Place
Destination_Name → City
Destination_Name → Nearby Landmark
Destination_Name → Area
Destination_Name → Pincode
Destination_Name → Bus_Station(Distance)
Destination_Name → Railway_Station(Distance)
Pincode → Area
Pincode → City
{Landmark, Area, City} → Pincode
{Dest_Name, Sight_Name} → Timing
{Dest_Name, Sight_Name} → TicketPrice
{Dest_Name, Sight_Name} → Distance_from_Dest
{Cust_ID, Destination_Name} → Start_Date
{Cust_ID, Destination_Name} → No_of_Days
Tour_ID → Company Name
Tour_ID → Website
Tour_ID → Email_ID
Tour_ID → Rating
{Destination_Name, Tour_ID, Customer_ID} → StartDate
{Destination_Name, Tour_ID, Customer_ID} → EndDate
{Destination_Name, Tour_ID, Customer_ID} → Budget_Constraint
{Destination_Name, Tour_ID, Customer_ID} → Children
{Destination_Name, Tour_ID, Customer_ID} → Adults
{Tour_ID, Destination_Name} → Consultant_Fee
{Tour_ID, Destination_Name} → Refund Percentage
{Tour_ID, Destination_Name, Days} → Amount
Hotel_ID → Hotel Name
Hotel_ID → Type of Hotel
Hotel_ID → Distance from Destination
Hotel_ID → Rating

Hotel_ID → Website
Website→ Hotel_ID
Website → Type of Hotel
Website → Distance from Destination
Website → Hotel Name
Website → Rating
{Hotel_ID, Room_Type} → Total Rooms
{Hotel_ID, Room_Type} → Price
{RoomType, Hotel_ID, Days} → Amount Percentage
Hotel_ID → ContactNumber
{Hotel_ID, Room_Type, Room_No, Startdate} → Cust_ID
{Hotel_ID, Room_Type, Room_No, Startdate} → EndDate
{Hotel_ID, Room_Type, Room_No, Startdate} → Tour_ID
{Hotel_ID, Room_Type, Room_No, Startdate} → Children
{Hotel_ID, Room_Type, Room_No, Startdate} → Adults
{Hotel_ID, Room_Type, Startdate, Cust_ID} → Tour_ID
{Hotel_ID, Room_Type, StartDate, Cust_ID} → CancellationDate
{Hotel_ID, Room_Type, StartDate, Cust_ID} → EndDate
{Hotel_ID, Cust_ID} → Stars
{Dest_Name, Cust_ID} → Stars
{Tour_ID, Cust_ID} → Stars
{Cust_ID, StartDate} → Hotel_ID

NORMALIZATION PROOFS:

CUSTOMER (Customer_ID, CustomerName,ContactNumber, Email, Gender, City, DOB) **:**
Customer_ID → {CustomerName, ContactNumber,Email,Gender, City, DOB}
ContactNumber → {Customer_ID,CustomerName, Email,Gender, City, DOB}

Computing the primary key:
$\{Customer\_ID\}^+$ → {Customer_ID,CustomerName, ContactNumber,Email,Gender, City, DOB}
$\{ContactNumber\}^+$ → {Customer_ID,CustomerName, ContactNumber,Email,Gender, City, DOB}

Customer_ID and ContactNumber can both be the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have Customer_ID or ContactNumber on the left.

DESTINATION (Destination_Name, Type of Place, City, Nearby Landmark, Area, Pincode, Bus_Station(Distance), Railway_Station(Distance))**:**
Destination_Name → {Type of Place, City, Nearby Landmark, Area, Pincode, Bus_Station(Distance), Railway_Station(Distance)}
Pincode → Area
Pincode → City
{Landmark, Area, City} → Pincode

Computing the primary key:
$\{Destination\_Name\}^+$ → {Destination_Name, Type of Place, City, Nearby Landmark, Area, Pincode, Bus_Station(Distance), Railway_Station(Distance)}
{Destination_Name} is the primary key.

BCNF Proof:
The relation is not in BCNF because the last three functional dependencies do not have the primary key on left.

SIGHTSEEING PLACES (Dest_Name, Sight_Name, Timing, Ticket_price, Distance_from_Dest)**:**
{Dest_Name, Sight_Name} → {Timing, TicketPrice, Distance_from_Dest}

Computing the primary key:
$\{Dest\_Name, Sight\_Name\}^+$ → {Dest_Name, Sight_Name, Timing, TicketPrice, Distance_from_Dest}
{Dest_Name, Sight_Name} is the primary key.

BCNF Proof:

The relation is in BCNF because all functional dependencies have the primary key {Dest_Name, Sight_Name} on the left.

VISITED (Destination_Name, Customer_ID, No_of_Days, StartDate)**:**
{Destination_Name, Customer_ID} → {No_of_Days, StartDate}

Computing the primary key:
$\{Destination\_Name, Customer\_ID\}^{+}$ → {Destination_Date, Customer_ID, No_of_Days, StartDate}
{Desination_Name, Customer_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Desination_Name, Customer_ID} on the left.

TOURS & TRAVELS (Tour_ID, Company Name, Website, Email_ID, Rating):
Tour_ID → {Company Name, Website, Email_ID, Rating}

Computing the primary key:
$\{Tour\_ID\}^{+}$ → {Tour_ID, Company Name, Website, Email_ID, Rating}
{Tour_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Tour_ID} on the left.

TOURIST (Destination_Name, Tour_ID, Customer_ID, StartDate, EndDate, Budget_Constraint, Children, Adults):
{Destination_Name, Tour_ID, Customer_ID} → {StartDate, EndDate, Budget_Constraint, Children, Adults

Computing the primary key:
$\{Destination\_Name, Tour\_ID, Customer\_ID\}^{+}$ → {Destination_Name, Tour_ID, Customer_ID, StartDate, EndDate, Budget_Constraint, Children, Adults}
{Destination_Name, Tour_ID, Customer_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Destination_Name, Tour_ID, Customer_ID} on the left.

PRICING CONDITIONS (Tour_ID, Destination_Name, Consultant_Fee, Refund_Percentage):
{Tour_ID, Destination_Name} → {Consultant_Fee, Refund_Percentage}

Computing the primary key:
$\{Tour\_ID, Destination\_Name\}^{+}$ → {Tour_ID, Destination_Name, Consultant_Fee, Refund_Percentage}
{Tour_ID, Destination_Name} is the primary key.

BCNF Proof:

The relation is in BCNF because all functional dependencies have the primary key {Destination_Name, Tour_ID} on the left.

PACKAGES (Tour_ID, Destination_Name, Day, Amount):
{Tour_ID, Destination_Name, Days} → Amount

Computing the primary key:
$\{Tour\_ID, Destination\_Name, Days\}^+ \rightarrow Amount$
{Tour_ID, Destination_Name, Days} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Tour_ID, Destination_Name, Days} on the left.

HOTELS (Hotel_ID, Hotel Name, Type of Hotel, Distance from Destination, Rating, Website):
Hotel_ID → {Hotel Name, Type of Hotel, Distance from Destination, Rating, Website}
Website→ {Hotel_ID, Hotel Name, Type of Hotel, Distance from Destination, Rating}

Computing the primary key:
$\{Hotel\_ID\}^+ \rightarrow$ {Hotel_ID, Hotel Name, Type of Hotel, Distance from Destination, Rating, Website}
$\{Website\}^+ \rightarrow$ {Hotel_ID, Hotel Name, Type of Hotel, Distance from Destination, Rating, Website}
{Hotel_ID} and {Website} can both be the primary keys.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Hotel_ID} or {Website} on the left.

ROOM TYPE (Hotel_ID, Room_Type, Total Rooms, Price):
{Hotel_ID, Room_Type} → {Total Rooms, Price}

Computing the primary key:
$\{Hotel\_ID, Room\_Type\}^+ \rightarrow$ {Hotel_ID, Room_Type, Total Rooms, Price}
{Hotel_ID, Room_Type} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Hotel_ID,Room_Type} on the left.

REFUND POLICY (RoomType, Hotel_ID, Days, Amount Percentage):
{RoomType, Hotel_ID, Days} → Amount Percentage

Computing the primary key:

{RoomType, Hotel_ID, Days}$^+$ → {RoomType, Hotel_ID, Days, Amount Percentage}
{RoomType, Hotel_ID, Days} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {RoomType, Hotel_ID, Days} on the left.

BOOKINGS (Hotel_ID, Room_Type, Room_No, Startdate, Cust_ID, Enddate, Tour_ID, Children, Adults):
{Hotel_ID, Room_Type, Room_No, Startdate} → {Cust_ID, Enddate, Tour_ID, Children, Adults}

Computing the primary key:
{Hotel_ID, Room_Type, Room_No, Startdate}$^+$ → {Hotel_ID, Room_Type, Room_No, Startdate, Cust_ID, Enddate, Tour_ID, Children, Adults}
{Hotel_ID, Room_Type, Room_No, Startdate} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Hotel_ID, Room_Type, Room_No, Startdate} on the left.

CANCELLATIONS (Hotel_ID, Room_Type, Startdate, Cust_ID, Tour_ID,CancellationDate, EndDate):
{Hotel_ID, Room_Type, Startdate, Cust_ID} → {Tour_ID, CancellationDate, EndDate}

Computing the primary key:
{Hotel_ID, Room_Type, Startdate, Cust_ID}$^+$ → {Hotel_ID, Room_Type, Startdate, Cust_ID, Tour_ID, CancellationDate, EndDate}
{Hotel_ID, Room_Type, Startdate, Cust_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Hotel_ID, Room_Type, Startdate, Cust_ID} on the left.

RATING_HOTEL (Hotel_ID, Cust_ID, Stars):
{Hotel_ID, Cust_ID} → Stars

Computing the primary key:
{Hotel_ID, Cust_ID}$^+$ → {Hotel_ID, Cust_ID, Stars}
{Hotel_ID, Cust_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Hotel_ID, Cust_ID} on the left.

RATING_DEST (Dest_Name, Cust_ID, Stars):
{Dest_Name, Cust_ID} → Stars

Computing the primary key:
{Dest_Name, Cust_ID}$^+$ → {Hotel_ID, Cust_ID, Stars}
{Dest_Name, Cust_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Dest_Name, Cust_ID} on the left.

RATING_T&T (Tour_ID, Cust_ID, Stars):
{Tour_ID, Cust_ID} → Stars

Computing the primary key:
{Tour_ID, Cust_ID}$^+$ → {Hotel_ID, Cust_ID, Stars}
{Tour_ID, Cust_ID} is the primary key.

BCNF Proof:
The relation is in BCNF because all functional dependencies have the primary key {Tour_ID, Cust_ID} on the left.

ROOM NUMBER (Hotel_ID, Room_Type, Room_No):

Computing the primary key:
{Hotel_ID, Room_Type, Room_No}$^+$ → {Hotel_ID, Room_Type, Room_No}
{Hotel_ID, Room_Type, Room_No} is the primary key.

This relation only has a non-trivial functional dependency. So, it is in BCNF, as all functional dependencies only have the primary key on the left.

TOUR CONTACT DETAILS (Tour_ID, TContactNumber):

Computing the primary key:
{Tour_ID, TContactNumber}$^+$ → {Tour_ID, ContactNumber}
{Tour_ID, TContactNumber} is the primary key.

This relation only has a non-trivial functional dependency. So, it is in BCNF, as all functional dependencies only have the primary key on the left.

AVAILABLE HOTEL(Destination Name , Hotel_ID) :

Computing the primary key:
{Destination Name, Hotel_ID}$^+$ → {Destination Name , Hotel_ID}
{Destination Name, Hotel_ID} is the primary key.

This relation only has a non-trivial functional dependency. So, it is in BCNF, as all functional dependencies only have the primary key on the left.

HOTEL CONTACT DETAILS (Hotel_ID,HContactNumber):

Computing the primary key:
{Hotel_ID, HContactNumber}$^+$ → {Hotel_ID, ContactNumber}

{Hotel_ID, HContactNumber} is the primary key.

This relation only has a non-trivial functional dependency. So, it is in BCNF, as all functional dependencies only have the primary key on the left.