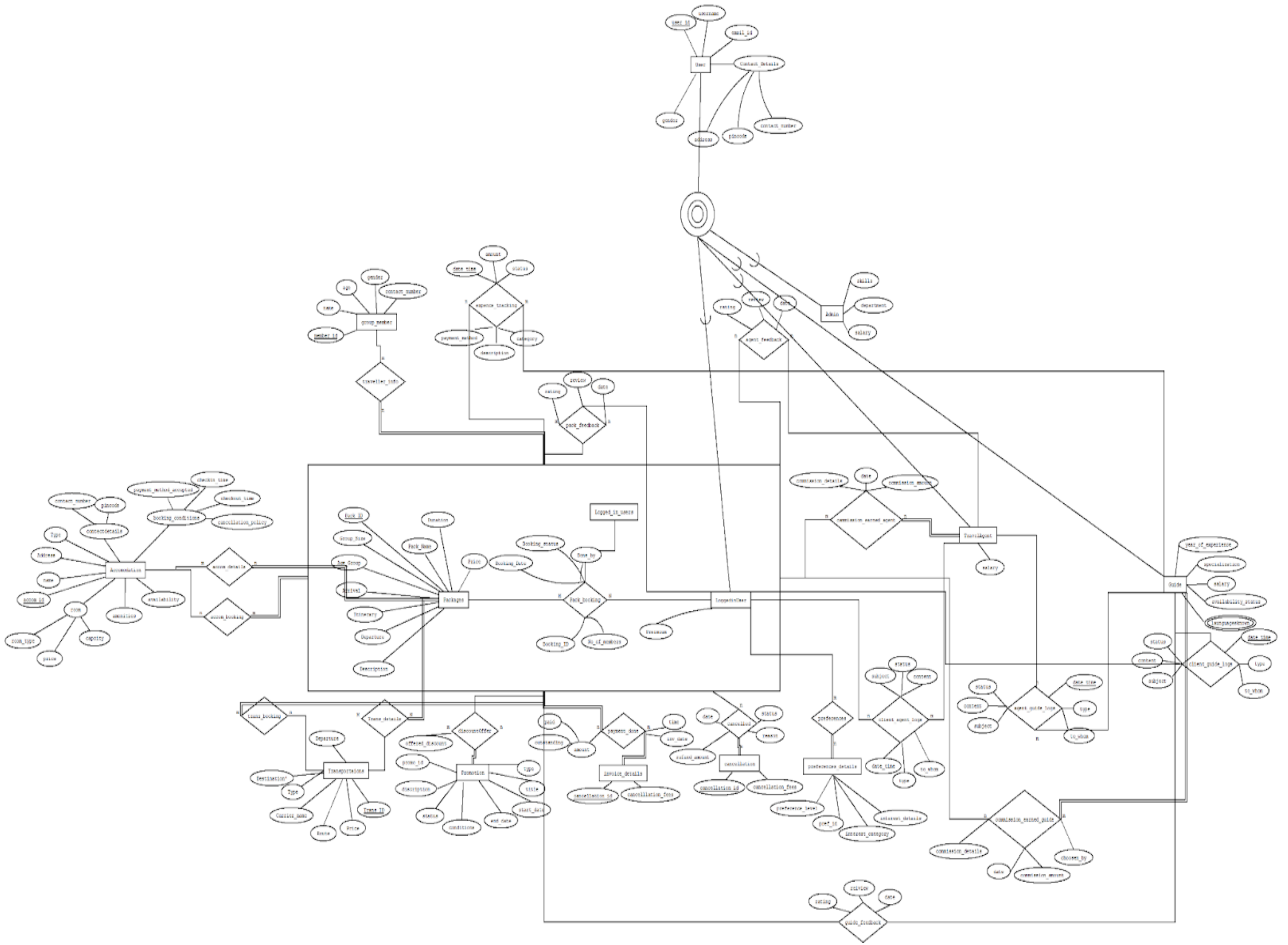
DBMS
WonderLust Heaven

Jenish Vadodariya - 202301202
Archan Maru - 202301217
Dwarkesh Vaghasiya - 202301225
Manan Ghonia - 202301240
Charvik Tejani - 202301242

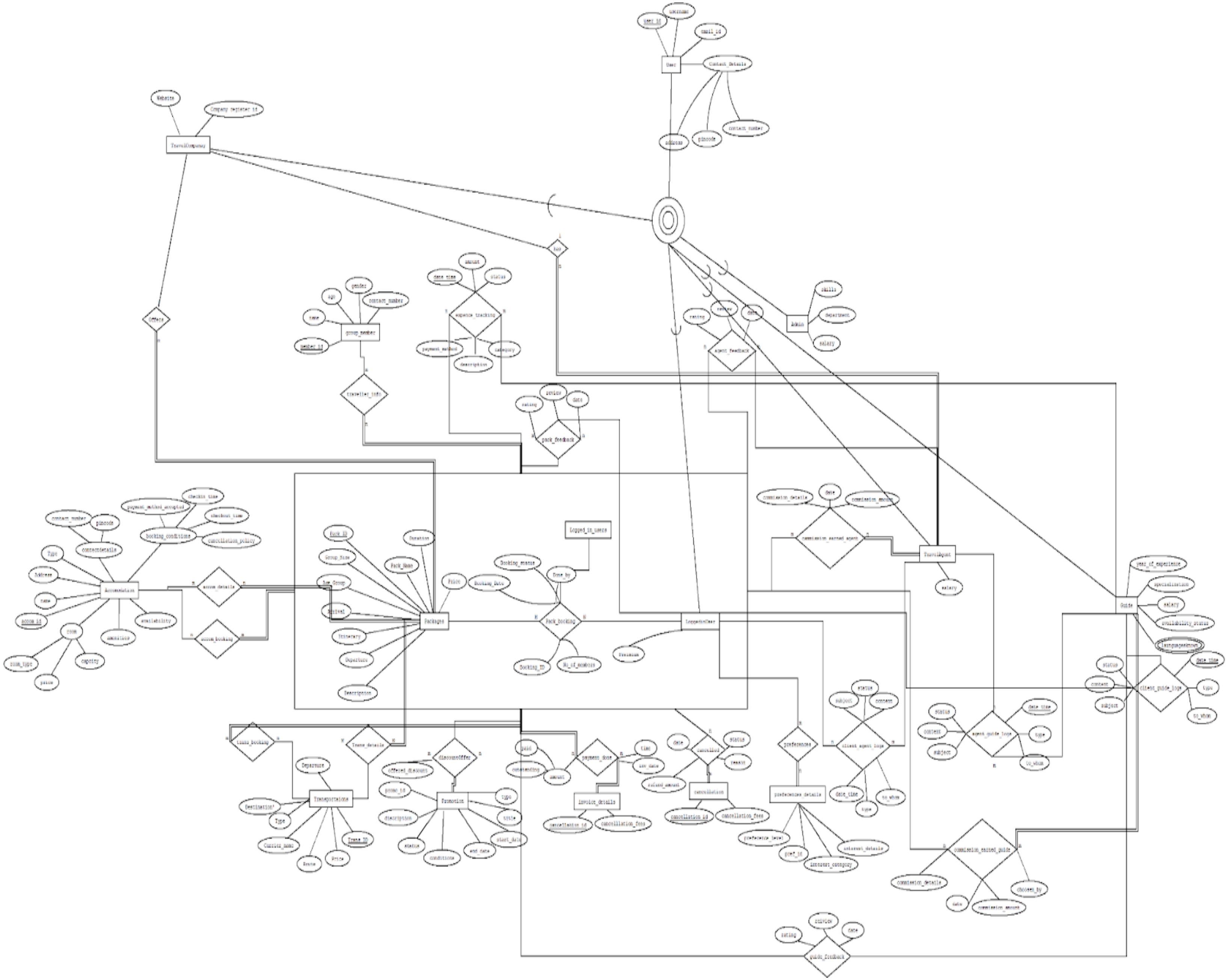
ERD to Relational mapping:

Here, in this file, we are attaching old ERD as well as modified ERD as we have made some changes in the ERD like we have added common attributes of subclass to the superclass (i.e., contact_number, pincode, address) and added TravelCompany new subclass added and new relation and some attribute changes made like we removed gender from user since there is no gender of company.

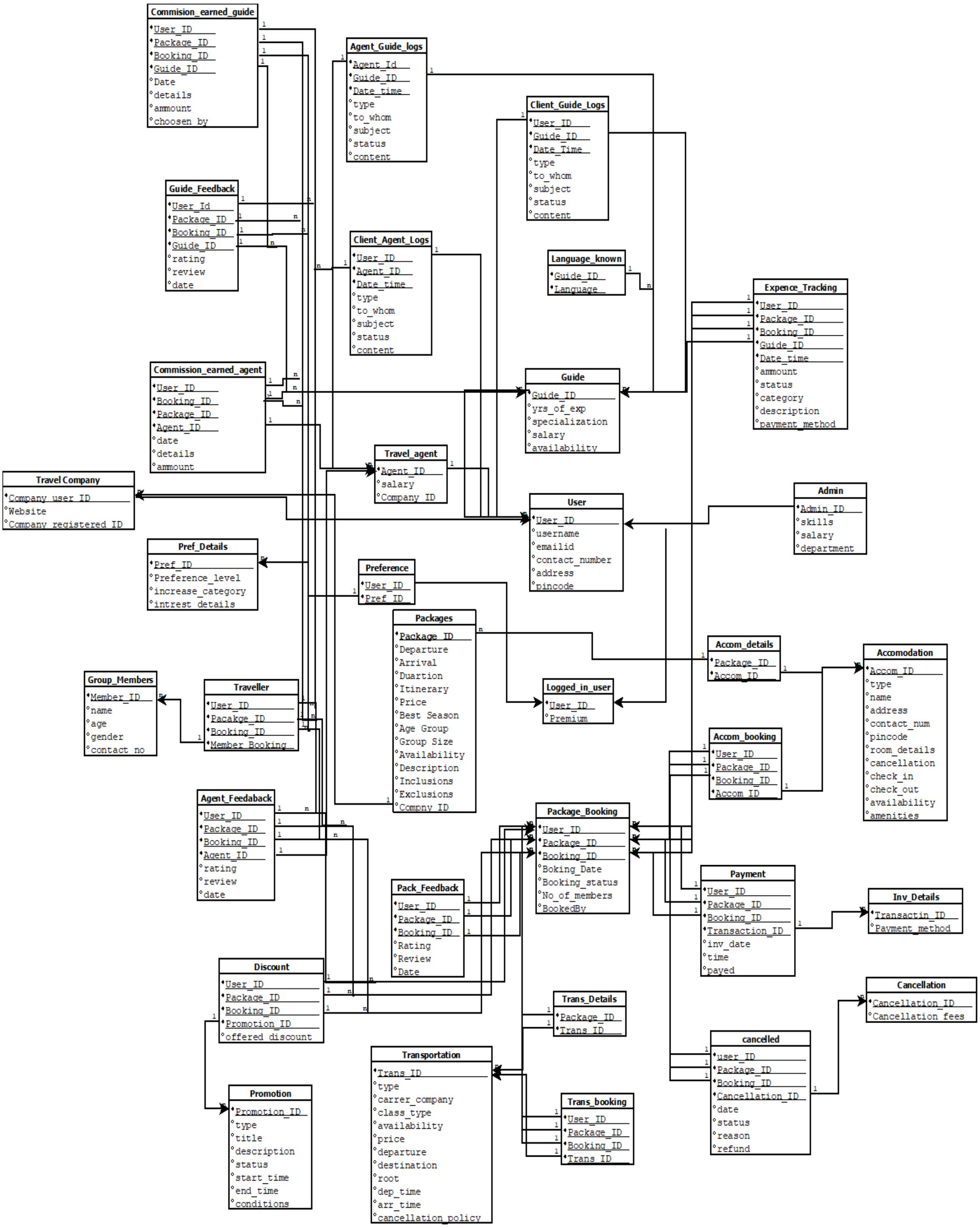
OLD Schema


User
user_id
username
email_id
Contact_Details
gender
address
pincode
contact_number
Admin
skills
department
salary
agent_feedback
rating
review
date
group_member
member_id
name
age
gender
contact_number
traveller_info
expence_tracking
date_time
amount
status
payment_method
description
category
pack_feedback
rating
review
date
Accommodation
accom_id
name
Address
Type
contactdetails
contact_number
pincode
booking_conditions
payment_method_accepted
checkin_time
checkout_time
cancellation_policy
availability
amenities
room
room_type
price
capcity
accom_details
accom_booking
Packages
Pack_ID
Group_Size
Age_Group
Arrival
Itinerary
Departure
Description
Pack_Name
Duration
Price
Pack_booking
Booking_status
Booking_Date
Done_by
Logged_in_users
Booking_ID
No_of_members
LoggedinUser
Premium
trans_booking
Trans_details
Transportations
Departure
Destination
Type
Carrier_name
Route
Price
Trans_ID
discountOffer
offered_discount
Promotion
promo_id
description
status
conditions
end_date
start_date
title
type
payment_done
paid
outstanding
amount
time
inv_date
invoice_details
cancellation_id
cancellation_fees
cancelled
date
status
reason
refund_amount
cancellation
cancellation_id
cancellation_fees
preferences
preferences_details
preference_level
pref_id
interest_category
interest_details
client_agent_logs
subject
status
content
date_time
type
to_whom
commission_earned_agent
commission_details
date
commission_amount
TravelAgent
salary
agent_guide_logs
status
content
subject
date_time
type
to_whom
Guide
year_of_experience
specialization
salary
availability_status
languagesknown
client_guide_logs
status
content
subject
date_time
type
to_whom
commission_earned_guide
commission_details
date
commission_amount
choosen_by
guide_feedback
rating
review
date
n
m
M
N

NEW Schema : added Company and change in some attribute



Relational Schema According To new ER Diagram



DDL Script

```
CREATE SCHEMA WonderLustHeaven;

SET SEARCH_PATH TO WonderLustHeaven;

CREATE TABLE User_
(
    user_id int,
    username varchar(40),
    email_id varchar(40),
    contact_number varchar(10),
    pincode varchar(6),
    address_ varchar(100),
    PRIMARY KEY (user_id)
);

CREATE TABLE Company(
    user_id int,
    website VARCHAR(40),
    companyRegesterid int,
    FOREIGN KEY (user_id) REFERENCES User_(user_id) ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE LoggedInUser
(
    user_id int,
    premium varchar(3),
    PRIMARY KEY (user_id),
    FOREIGN KEY (user_id) REFERENCES User_(user_id) ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE Travel_Agent
(
    agent_id int PRIMARY KEY,
    companyid int,
    salary numeric(7,2),
    FOREIGN KEY (agent_id) REFERENCES User_(user_id) ON DELETE CASCADE
ON UPDATE CASCADE,
    FOREIGN KEY (companyid) REFERENCES Company(user_id) ON DELETE
CASCADE ON UPDATE CASCADE
```

```sql
);

CREATE TABLE Guide
(
    guide_id int,
    yrs_of_experience int,
    specialization varchar(100),
    salary numeric(7,2),
    availability_status varchar(15),
    PRIMARY KEY (guide_id),
    FOREIGN KEY (guide_id) REFERENCES User_(user_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Languages_Known
(
    guide_id int,
    language_ varchar(15),
    PRIMARY KEY (guide_id,language_),
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Admin
(
    admin_id int,
    skills varchar(100),
    salary numeric(7,2),
    department varchar(20),
    PRIMARY KEY (admin_id)
);

CREATE TABLE Packages
(
    package_id int,
    companyid int,
    package_name varchar(50),
    departure_dest varchar(25),
    arrival_dest varchar(25),
    duration varchar(30),
    itinerary varchar(150),
    price numeric(7,2),
    best_season varchar(20),
    age_grp varchar(10),
    grp_size varchar(12),
    availability_ varchar(15),
    description_ varchar(1000),
    inclusions varchar(500),
    exclusions varchar(500),
    PRIMARY KEY (package_id),
    FOREIGN KEY (companyid) REFERENCES Company(user_id) ON DELETE
CASCADE ON UPDATE CASCADE
);


CREATE TABLE Package_Booking
(
    user_id int,
    package_id int,
    booking_id int,
    booking_date date,
    booking_status varchar(15),
    no_of_members varchar(20),
    done_by varchar(15),
    PRIMARY KEY (user_id,package_id,booking_id),
    FOREIGN KEY (user_id) REFERENCES LoggedInUser(user_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (package_id) REFERENCES Packages(package_id) ON DELETE
CASCADE ON UPDATE CASCADE
);

CREATE TABLE Accommodation
(
    accom_id int,
    type_ varchar(15),
    name_ varchar(50),
    address_ varchar(100),
    contact_no varchar(10),
    pincode varchar(6),
    room_type varchar(20),
    room_price numeric(7,2),
    room_capacity int,
    payment_method_accepted varchar(80),
    cancellation_policy varchar(300),
    checkin_time varchar(10),
    checkout_time varchar(10),
    availability_ varchar(10),
    amenities varchar(500),
    PRIMARY KEY (accom_id)
);

CREATE TABLE Accom_details
(
    package_id int,
    accom_id int,
    PRIMARY KEY (package_id,accom_id),
    FOREIGN KEY (package_id) REFERENCES Packages(package_id) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (accom_id) REFERENCES Accommodation(accom_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Accom_booking
(
    user_id int,
    package_id int,
    booking_id int,
    accom_id int,
    PRIMARY KEY (user_id,package_id,booking_id,accom_id),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (accom_id) REFERENCES Accommodation(accom_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Transportation
(
    trans_id int,
    type_ varchar(10),
    carrier_company varchar(50),
    classtype varchar(15),
    available_seats int,
    price numeric(7,2),
    depart_location varchar(25),
    dest_location varchar(25),
    route_ varchar(50),
    depart_time varchar(15),
    arrival_time varchar(15),
```

```sql
    payment_method_accepted varchar(80),
    cancellation_policy varchar(300),
    amenities varchar(500),
    PRIMARY KEY (trans_id)
);

CREATE TABLE Trans_details
(
    package_id int,
    trans_id int,
    PRIMARY KEY (package_id,trans_id),
    FOREIGN KEY (package_id) REFERENCES Packages(package_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (trans_id) REFERENCES Transportation(trans_id) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Trans_Booking
(
    user_id int,
    package_id int,
    booking_id int,
    trans_id int,
    PRIMARY KEY (user_id,package_id,booking_id,trans_id),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES
Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (trans_id) REFERENCES Transportation(trans_id) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE ExpenseTracking
(
    user_id int,
    package_id int,
    booking_id int,
    guide_id int,
    date_time timestamp,
    amount numeric(7,2),
    status_ varchar(10),
    category varchar(10),
    payment_method varchar(15),
    description_ varchar(150),
    PRIMARY KEY (user_id,package_id,booking_id,guide_id,date_time),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES
Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE Group_members
(
    member_id int PRIMARY KEY,
    name_ varchar(40),
    age int,
    gender varchar(6),
    contact_no varchar(10)
);

CREATE TABLE Traveller_info
(
    user_id int,
    package_id int,
    booking_id int,
    member_id int,
    PRIMARY KEY (user_id, package_id, booking_id, member_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (member_id) REFERENCES Group_members(member_id) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Invoice_details
(
    transaction_id int,
    payment_method varchar(20),
    PRIMARY KEY (transaction_id)
);

CREATE TABLE Payment_done
(
    user_id int,
    package_id int,
    booking_id int,
    transaction_id int,
    inv_date date,
    time_ varchar(15),
    paid numeric(7,2),
    outstanding numeric(7,2),
    PRIMARY KEY (user_id,package_id,booking_id,transaction_id),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES
Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (transaction_id) REFERENCES
Invoice_details(transaction_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Cancellation
(
    cancellation_id int,
    cancellation_fees numeric(7,2),
    PRIMARY KEY (cancellation_id)
);

CREATE TABLE Cancelled
(
    user_id int,
    package_id int,
    booking_id int,
    cancellation_id int,
    date_ date,
    status_ varchar(15),
    reason varchar(150),
    refund_amount numeric(7,2),
    PRIMARY KEY (user_id,package_id,booking_id,cancellation_id),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES
Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (cancellation_id) REFERENCES
Cancellation(cancellation_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Pack_Feedback
(
    user_id int,
    package_id int,
    booking_id int,
    rating int,
    review varchar(300),
    date_ date,
    PRIMARY KEY (user_id,package_id,booking_id),
    FOREIGN KEY (user_id,package_id,booking_id) REFERENCES
Package_Booking(user_id,package_id,booking_id) ON DELETE CASCADE ON
UPDATE CASCADE
);

CREATE TABLE Agent_feedback
(
    user_id int,
    package_id int,
    booking_id int,
    agent_id int,
    rating int,
    review varchar(300),
    date_ date,
    PRIMARY KEY (user_id, package_id, booking_id, agent_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (agent_id) REFERENCES Travel_Agent(agent_id) ON DELETE
CASCADE ON UPDATE CASCADE
);

CREATE TABLE Guide_feedback
(
    user_id int,
    package_id int,
    booking_id int,
    guide_id int,
    rating int,
    review varchar(300),
    date_ date,
    PRIMARY KEY (user_id, package_id, booking_id, guide_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE
ON UPDATE CASCADE
);
```

```sql
CREATE TABLE Commission_earned_agent
(
    user_id int,
    package_id int,
    booking_id int,
    agent_id int,
    date_ date,
    details varchar(250),
    amount numeric(7,2),
    PRIMARY KEY (user_id, package_id, booking_id, agent_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (agent_id) REFERENCES Travel_Agent(agent_id) ON DELETE
CASCADE ON UPDATE CASCADE
);

CREATE TABLE Commission_earned_guide
(
    user_id int,
    package_id int,
    booking_id int,
    guide_id int,
    date_ date,
    details varchar(250),
    amount numeric(7,2),
    chosen_by varchar(20),
    PRIMARY KEY (user_id, package_id, booking_id, guide_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE Preference_details
(
    pref_id int PRIMARY KEY,
    preference_level varchar(20),
    interest_category varchar(20),
    interest_details varchar(300)
);
```

```sql
CREATE TABLE Preferences
(
    user_id int,
    pref_id int,
    PRIMARY KEY (user_id, pref_id),
    FOREIGN KEY (user_id) REFERENCES LoggedInUser(user_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (pref_id) REFERENCES Preference_details(pref_id) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Promotion
(
    promotion_id varchar(20) PRIMARY KEY,
    type_ varchar(10),
    title varchar(25),
    description_ varchar(1000),
    status_ varchar(10),
    startdate varchar(15),
    enddate varchar(15),
    conditions varchar(300)
);

CREATE TABLE Discount_offer
(
    user_id int,
    package_id int,
    booking_id int,
    promotion_id varchar(20),
    offered_discount numeric(7,2),
    PRIMARY KEY (user_id, package_id, booking_id, promotion_id),
    FOREIGN KEY (user_id, package_id, booking_id) REFERENCES
Package_Booking(user_id, package_id, booking_id) ON DELETE CASCADE ON
UPDATE CASCADE,
    FOREIGN KEY (promotion_id) REFERENCES Promotion(promotion_id) ON
DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE Client_Agent_logs
(
    user_id int,
    agent_id int,
    date_time timestamp,
    type_ varchar(15),
    to_whom varchar(6),
    subject_ varchar(20),
    status_ varchar(15),
    content varchar(300),
    PRIMARY KEY (user_id, agent_id, date_time),
    FOREIGN KEY (user_id) REFERENCES LoggedInUser(user_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (agent_id) REFERENCES Travel_Agent(agent_id) ON DELETE
CASCADE ON UPDATE CASCADE
);

CREATE TABLE Client_Guide_logs
(
    user_id int,
    guide_id int,
    date_time timestamp,
    type_ varchar(15),
    to_whom varchar(6),
    subject_ varchar(20),
    status_ varchar(15),
    content varchar(300),
    PRIMARY KEY (user_id,guide_id,date_time),
    FOREIGN KEY (user_id) REFERENCES LoggedInUser(user_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE
ON UPDATE CASCADE
);

CREATE TABLE Agent_Guide_logs
(
    agent_id int,
    guide_id int,
    date_time timestamp,
    type_ varchar(15),
    to_whom varchar(6),
    subject_ varchar(20),
    status_ varchar(15),
    content varchar(300),
    PRIMARY KEY (agent_id, guide_id, date_time),
    FOREIGN KEY (agent_id) REFERENCES Travel_Agent(agent_id) ON DELETE
CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (guide_id) REFERENCES Guide(guide_id) ON DELETE CASCADE
ON UPDATE CASCADE
);
```

Projected FD SET

FDs of Relation: User

user_id → user_name
user_id → email_id
user_id → contact_number
user_id → pincode
user_id → address

As closure of {user_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Company

user_id → user_name
user_id → email_id
user_id → contact_number
user_id → pincode
user_id → address
user_id → website
user_id → companyregisterid

As closure of {user_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: LoggedInUser

user_id → user_name
user_id → email_id

user_id → contact_number
user_id → pincode
user_id → address
user_id → gender
user_id → premium

As closure of {user_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Travel Agent

agent_id → user_name
agent_id → email_id
agent_id → contact_number
agent_id → pincode
agent_id → address
agent_id → gender
agent_id → salary

As closure of {agent_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Guide

guide_id → user_name
guide_id → email_id
guide_id → contact_number
guide_id → pincode
guide_id → address
guide_id → gender
guide_id → salary
guide_id → yrs_of_experience
guide_id → specialization
guide_id → availability_status

As closure of {guide_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Admin

admin_id → user_name
admin_id → email_id
admin_id → contact_number
admin_id → pincode
admin_id → address
admin_id → gender
admin_id → salary
admin_id → skills
admin_id → department

As closure of {admin_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Packages

package_id → package_name
package_id → departure_dest
package_id → arrival_dest
package_id → duration
package_id → itinerary
package_id → price
package_id → best_season
package_id → age_group
package_id → grp_size
package_id → availability
package_id → description
package_id → inclusions
package_id → exclusions

As closure of {package_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Package_booking

(user_id, package_id, booking_id) → booking_date
(user_id, package_id, booking_id) → booking_status
(user_id, package_id, booking_id) → no_of_members
(user_id, package_id, booking_id) → done_by

As closure of {user_id, package_id, booking_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Accommodation

accom_id → type_
accom_id → name_
accom_id → address_
accom_id → contact_number
accom_id → pincode
accom_id → room_type
accom_id → room_price
accom_id → room_capacity
accom_id → payment_method_accepted
accom_id → cancellation_policy
accom_id → checkin_time
accom_id → checkout_time
accom_id → availability_
accom_id → amenities

As closure of {accom_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Transportation

trans_id → type_
trans_id → carrier_company
trans_id → classtype

trans_id → available_seats
trans_id → price
trans_id → depart_location
trans_id → dest_location
trans_id → route_
trans_id → depart_time
trans_id → arrival_time
trans_id → payment_method_accepted
trans_id → cancellation_policy
trans_id → amenities

As closure of {trans_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Expense_tracking

(user_id, package_id, booking_id, guide_id, date_time) → amount
(user_id, package_id, booking_id, guide_id, date_time) → status
(user_id, package_id, booking_id, guide_id, date_time) → category
(user_id, package_id, booking_id, guide_id, date_time) → payment_method
(user_id, package_id, booking_id, guide_id, date_time) → description_

As closure of {user_id, package_id, booking_id, guide_id, date_time} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Group members

member_id → name_
member_id → age
member_id → gender
member_id → contact_number

As closure of {member_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Invoice_details

transaction_id → payment_method

As closure of {transaction_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Payment_done

(user_id, package_id, booking_id, transaction_id) → inv_date
(user_id, package_id, booking_id, transaction_id) → time_
(user_id, package_id, booking_id, transaction_id) → paid
(user_id, package_id, booking_id, transaction_id) → outstanding

As closure of {user_id, package_id, booking_id, transaction_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Cancellation

cancellation_id → cancellation_fees

As closure of {cancellation_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Cancelled

(user_id, package_id, booking_id, cancellation_id) → date_
(user_id, package_id, booking_id, cancellation_id) → status_
(user_id, package_id, booking_id, cancellation_id) → reason
(user_id, package_id, booking_id, cancellation_id) → refund_amount
As closure of {user_id, package_id, booking_id, cancellation_id} determines

all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Pack_feedback

(user_id, package_id, booking_id) → rating
(user_id, package_id, booking_id) → review
(user_id, package_id, booking_id) → date_

As closure of {user_id, package_id, booking_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Agent_feedback

(user_id, package_id, booking_id, agent_id) → rating
(user_id, package_id, booking_id, agent_id) → review
(user_id, package_id, booking_id, agent_id) → date_

As closure of {user_id, package_id, booking_id, agent_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Guide_feedback

(user_id, package_id, booking_id, guide_id) → rating
(user_id, package_id, booking_id, guide_id) → review
(user_id, package_id, booking_id, guide_id) → date_

As closure of {user_id, package_id, booking_id, guide_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Commission_earned_agent

(user_id, package_id, booking_id, agent_id) → date_
(user_id, package_id, booking_id, agent_id) → details

(user_id, package_id, booking_id, agent_id) → amount

As closure of {user_id, package_id, booking_id, agent_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Commission_earned_guide

(user_id, package_id, booking_id, guide_id) → date_
(user_id, package_id, booking_id, guide_id) → details
(user_id, package_id, booking_id, guide_id) → amount
(user_id, package_id, booking_id, guide_id) → chosen_by

As closure of {user_id, package_id, booking_id, guide_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Preference_details

pref_id → preference_level
pref_id → interest_category
pref_id → interest_details

As closure of {pref_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Promotion

promotion_id → type_
promotion_id → title
promotion_id → description_
promotion_id → status_
promotion_id → startdate
promotion_id → enddate
promotion_id → conditions

As closure of {promotion_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Discount_offer

(user_id, package_id, booking_id, promotion_id) → offered_discount
As closure of {user_id, package_id, booking_id, promotion_id} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Client_agent_logs

(user_id, agent_id, date_time) → type_
(user_id, agent_id, date_time) → to_whom
(user_id, agent_id, date_time) → subject_
(user_id, agent_id, date_time) → status_
(user_id, agent_id, date_time) → content

As closure of {user_id, agent_id, date_time} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Client_guide_logs

(user_id, guide_id, date_time) → type_
(user_id, guide_id, date_time) → to_whom
(user_id, guide_id, date_time) → subject_
(user_id, guide_id, date_time) → status_
(user_id, guide_id, date_time) → content

As closure of {user_id, guide_id, date_time} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

FDs of Relation: Agent_guide_logs

(agent_id, guide_id, date_time) → type_
(agent_id, guide_id, date_time) → to_whom
(agent_id, guide_id, date_time) → subject_
(agent_id, guide_id, date_time) → status_
(agent_id, guide_id, date_time) → content

As closure of {agent_id, guide_id, date_time} determines all the attributes of this relation, so it is a super-key. Hence, this relation is in **BCNF**.

Minimal FD Set for All Relations (BCNF Proof)

FDs of Relation: User
user_id → user_name, email_id, contact_number, pincode, address
As closure of {user_id} determines all attributes, it is a super-key ⇒ BCNF.

FDs of Relation: Company
user_id → user_name, email_id, contact_number, pincode, address, website, companyregisterid

FDs of Relation: LoggedInUser
user_id → user_name, email_id, contact_number, pincode, address, gender, premium

FDs of Relation: Travel Agent
agent_id → user_name, email_id, contact_number, pincode, address, gender, salary

FDs of Relation: Guide
guide_id → user_name, email_id, contact_number, pincode, address, gender, salary, yrs_of_experience, specialization, availability_status
Super-key ⇒ BCNF.

FDs of Relation: Admin
admin_id → user_name, email_id, contact_number, pincode, address, gender, salary, skills, department

FDs of Relation: Packages
package_id → package_name, departure_dest, arrival_dest, duration, itinerary, price, best_season, age_group, grp_size, availability, description,

inclusions, exclusions
Super-key ⇒ BCNF.

FDs of Relation: Package_booking
(user_id, package_id, booking_id) → booking_date, booking_status, no_of_members, done_by
Super-key ⇒ BCNF.

FDs of Relation: Accommodation
accom_id → type_, name_, address_, contact_number, pincode, room_type, room_price, room_capacity, payment_method_accepted, cancellation_policy, checkin_time, checkout_time, availability_, amenities
Super-key ⇒ BCNF.

FDs of Relation: Transportation
trans_id → type_, carrier_company, classtype, available_seats, price, depart_location, dest_location, route_, depart_time, arrival_time, payment_method_accepted, cancellation_policy, amenities
Super-key ⇒ BCNF.

FDs of Relation: Expense_tracking
(user_id, package_id, booking_id, guide_id, date_time) → amount, status, category, payment_method, description_

FDs of Relation: Group members
member_id → name_, age, gender, contact_number

FDs of Relation: Invoice_details
transaction_id → payment_method

FDs of Relation: Payment_done
(user_id, package_id, booking_id, transaction_id) → inv_date, time_, paid, outstanding

FDs of Relation: Cancellation
cancellation_id → cancellation_fees

FDs of Relation: Cancelled
(user_id, package_id, booking_id, cancellation_id) → date_, status_, reason, refund_amount

FDs of Relation: Pack_feedback
(user_id, package_id, booking_id) → rating, review, date_

FDs of Relation: Agent_feedback
(user_id, package_id, booking_id, agent_id) → rating, review, date_

FDs of Relation: Guide_feedback
(user_id, package_id, booking_id, guide_id) → rating, review, date_

FDs of Relation: Commission_earned_agent
(user_id, package_id, booking_id, agent_id) → date_, details, amount

FDs of Relation: Commission_earned_guide
(user_id, package_id, booking_id, guide_id) → date_, details, amount, chosen_by

FDs of Relation: Preference_details
pref_id → preference_level, interest_category, interest_details

FDs of Relation: Promotion
promotion_id → type_, title, description_, status_, startdate, enddate, conditions

FDs of Relation: Discount_offer
(user_id, package_id, booking_id, promotion_id) → offered_discount

FDs of Relation: Client_agent_logs
(user_id, agent_id, date_time) → type_, to_whom, subject_, status_, content

FDs of Relation: Client_guide_logs
(user_id, guide_id, date_time) → type_, to_whom, subject_, status_, content

FDs of Relation: Agent_guide_logs
(agent_id, guide_id, date_time) → type_, to_whom, subject_, status_, content

Conclusion:
All relations are in BCNF. Thus, the relational schema is well-normalized with no redundancy

INSERT Script

```
INSERT INTO User_ (user_id, username, email_id, contact_number, pincode, address_) VALUES
  (1, 'alice', 'alice@example.com', '1111111111', '560001', '123 A Street'),
  (2, 'bob', 'bob@example.com', '2222222222', '560002', '234 B Avenue'),
  (3, 'carol', 'carol@example.com', '3333333333', '560003', '345 C Road'),
  (4, 'dave', 'dave@example.com', '4444444444', '560004', '456 D Blvd'),
  (5, 'eve', 'eve@example.com', '5555555555', '560005', '567 E Lane'),
  (6, 'frank', 'frank@example.com', '6666666666', '560006', '678 F Court'),
  (7, 'grace', 'grace@example.com', '7777777777', '560007', '789 G Drive'),
  (8, 'heidi', 'heidi@example.com', '8888888888', '560008', '890 H Terrace'),
  (9, 'ivan', 'ivan@example.com', '9999999999', '560009', '901 I Circle'),
  (10, 'judy', 'judy@example.com', '1010101010', '560010', '101 J Plaza');

INSERT INTO Company (user_id, website, companyRegesterid) VALUES
  (1, 'www.aliceco.com', 1001),
  (2, 'www.bobco.com', 1002),
  (3, 'www.carolco.com', 1003),
  (4, 'www.daveco.com', 1004),
  (5, 'www.eveco.com', 1005),
  (6, 'www.frankco.com', 1006),
  (7, 'www.graceco.com', 1007),
  (8, 'www.heidico.com', 1008),
  (9, 'www.ivanco.com', 1009),
  (10, 'www.judyco.com', 1010);

INSERT INTO LoggedInUser (user_id, premium) VALUES
  (1, 'yes'),
  (2, 'no'),
  (3, 'yes'),
  (4, 'no'),
  (5, 'yes'),
  (6, 'no'),
  (7, 'yes'),
  (8, 'no'),
  (9, 'yes'),
  (10, 'no');

INSERT INTO Travel_Agent (agent_id, companyid, salary) VALUES
  (1, 1, 3000.00),
  (2, 2, 3200.00),
  (3, 3, 3100.00),
  (4, 4, 3300.00),
  (5, 5, 3400.00),
  (6, 6, 3500.00),
  (7, 7, 3600.00),
  (8, 8, 3700.00),
  (9, 9, 3800.00),
  (10, 10, 3900.00);

INSERT INTO Guide (guide_id, yrs_of_experience, specialization, salary,
availability_status) VALUES
  (1, 5, 'History', 2500.00, 'Available'),
  (2, 6, 'Nature', 2600.00, 'Busy'),
  (3, 7, 'Architecture', 2700.00, 'Available'),
  (4, 8, 'Art', 2800.00, 'Busy'),
  (5, 4, 'Wildlife', 2900.00, 'Available'),
  (6, 3, 'Food', 3000.00, 'Available'),
  (7, 10, 'Culture', 3100.00, 'Busy'),
  (8, 9, 'Photography', 3200.00, 'Available'),
  (9, 11, 'Adventure', 3300.00, 'Busy'),
  (10, 2, 'Local History', 2400.00, 'Available');

INSERT INTO Languages_Known (guide_id, language_) VALUES
  (1, 'English'),
  (2, 'Spanish'),
  (3, 'French'),
  (4, 'German'),
  (5, 'Italian'),
  (6, 'Mandarin'),
  (7, 'Japanese'),
  (8, 'Korean'),
  (9, 'Russian'),
  (10, 'Hindi');
```

```
INSERT INTO Admin (admin_id, skills, salary, department) VALUES
  (1, 'Management', 5000.00, 'Operations'),
  (2, 'IT', 5200.00, 'Tech'),
  (3, 'HR', 5100.00, 'Human Resources'),
  (4, 'Finance', 5300.00, 'Accounting'),
  (5, 'Logistics', 5400.00, 'Operations'),
  (6, 'Support', 5500.00, 'Customer Service'),
  (7, 'Security', 5600.00, 'Admin'),
  (8, 'Legal', 5700.00, 'Compliance'),
  (9, 'Marketing', 5800.00, 'Sales'),
  (10, 'IT Security', 5900.00, 'Tech');

INSERT INTO Packages (package_id, companyid, package_name,
departure_dest, arrival_dest, duration, itinerary, price, best_season,
age_grp, grp_size, availability_, description_, inclusions, exclusions)
VALUES
  (1, 1, 'City Explorer', 'CityA', 'CityB', '3 days', 'Day1: CityA
tour; Day2: Museum visit; Day3: Shopping', 4999.99, 'Spring', 'Adult',
'Small', 'Yes', 'Explore urban landmarks', 'Hotel, transport',
'Meals'),
  (2, 2, 'Mountain Adventure', 'TownA', 'TownB', '5 days', 'Hiking and
camping', 6999.99, 'Summer', 'All', 'Medium', 'Yes', 'Enjoy the
mountains', 'Guided trek', 'Extras'),
  (3, 3, 'Beach Holiday', 'CoastA', 'CoastB', '7 days', 'Resort stay
and water sports', 8999.99, 'Winter', 'Family', 'Large', 'No', 'Relax
by the sea', 'Resort, meals', 'Drinks'),
  (4, 4, 'Cultural Journey', 'CityC', 'CityD', '4 days', 'City tours
and cultural events', 5999.99, 'Autumn', 'Adult', 'Small', 'Yes',
'Immerse in local culture', 'Tickets included', 'Exhibits'),
  (5, 5, 'Safari Expedition', 'BaseA', 'BaseB', '6 days', 'Wildlife
viewing', 10999.99, 'Summer', 'All', 'Medium', 'Yes', 'Experience the
wild', 'Safari jeep, guide', 'Insurance'),
  (6, 6, 'Historical Tour', 'OldTown', 'HeritageSite', '2 days',
'Historic site visits', 3999.99, 'Spring', 'Adult', 'Small', 'Yes',
'Step back in time', 'Museum tickets', 'Snacks'),
  (7, 7, 'Foodie Delight', 'GourmetCity', 'CuisineTown', '3 days',
'Food tours and cooking classes', 4999.99, 'Winter', 'Family', 'Small',
'Yes', 'Taste local cuisine', 'Cooking session', 'Drinks'),
  (8, 8, 'Winter Wonderland', 'SnowCity', 'IcyTown', '5 days', 'Skiing
and winter sports', 7999.99, 'Winter', 'All', 'Large', 'No', 'Enjoy
snow adventures', 'Ski pass', 'Equipment'),
  (9, 9, 'Desert Escape', 'DesertStart', 'OasisEnd', '4 days', 'Camel
ride and desert camp', 5999.99, 'Autumn', 'Adult', 'Medium', 'Yes',
'Escape to the desert', 'Camp, food', 'Extra costs'),
  (10, 10, 'Island Getaway', 'PortA', 'PortB', '7 days', 'Island
hopping', 9999.99, 'Summer', 'Family', 'Large', 'Yes', 'Relax on
islands', 'Boat ride, hotel', 'Taxes');

INSERT INTO Package_Booking (user_id, package_id, booking_id,
booking_date, booking_status, no_of_members, done_by) VALUES
  (1, 1, 1, '2025-04-01', 'Confirmed', '2', 'online'),
  (2, 2, 1, '2025-04-02', 'Pending', '4', 'agent'),
  (3, 3, 1, '2025-04-03', 'Cancelled', '3', 'online'),
  (4, 4, 1, '2025-04-04', 'Confirmed', '1', 'agent'),
  (5, 5, 1, '2025-04-05', 'Confirmed', '5', 'online'),
  (6, 6, 1, '2025-04-06', 'Pending', '2', 'agent'),
  (7, 7, 1, '2025-04-07', 'Confirmed', '3', 'online'),
  (8, 8, 1, '2025-04-08', 'Cancelled', '4', 'agent'),
  (9, 9, 1, '2025-04-09', 'Confirmed', '2', 'online'),
  (10, 10, 1, '2025-04-10', 'Confirmed', '6', 'agent');

INSERT INTO Accommodation (accom_id, type_, name_, address_,
contact_no, pincode, room_type, room_price, room_capacity,
payment_method_accepted, cancellation_policy, checkin_time,
checkout_time, availability_, amenities) VALUES
  (1, 'Hotel', 'Hotel One', 'Address 1', '1111111111', '560001',
'Deluxe', 2000.00, 2, 'Cash, Card', '24h before checkin', '14:00',
'12:00', 'Yes', 'WiFi, TV'),
  (2, 'Resort', 'Resort Two', 'Address 2', '2222222222', '560002',
'Suite', 3000.00, 4, 'Card', '48h before checkin', '15:00', '11:00',
'No', 'Pool, Spa'),
  (3, 'Hostel', 'Hostel Three', 'Address 3', '3333333333', '560003',
'Standard', 1500.00, 1, 'Cash', 'No refund', '13:00', '10:00', 'Yes',
'Breakfast'),
  (4, 'Hotel', 'Hotel Four', 'Address 4', '4444444444', '560004',
'Business', 2500.00, 2, 'Cash, Card', '24h before checkin', '14:00',
'12:00', 'Yes', 'Gym, WiFi'),
  (5, 'Resort', 'Resort Five', 'Address 5', '5555555555', '560005',
'Deluxe', 3500.00, 3, 'Card', '48h before checkin', '15:00', '11:00',
'No', 'Pool, Bar'),
  (6, 'Guest House', 'Guest Six', 'Address 6', '6666666666', '560006',
'Standard', 1800.00, 2, 'Cash, Card', '24h before checkin', '12:00',
'10:00', 'Yes', 'Breakfast, WiFi'),
  (7, 'Hotel', 'Hotel Seven', 'Address 7', '7777777777', '560007', 'Suite', 4000.00, 4, 'Cash, Card', '48h before checkin', '16:00', '12:00', 'Yes', 'Gym, Spa'),
  (8, 'Resort', 'Resort Eight', 'Address 8', '8888888888', '560008', 'Deluxe', 3200.00, 3, 'Card', '24h before checkin', '14:00', '11:00', 'No', 'Pool, TV'),
  (9, 'Hostel', 'Hostel Nine', 'Address 9', '9999999999', '560009', 'Standard', 1200.00, 1, 'Cash', 'No refund', '13:00', '10:00', 'Yes', 'Breakfast'),
  (10, 'Guest House', 'Guest Ten', 'Address 10', '1010101010', '560010', 'Business', 2800.00, 2, 'Cash, Card', '24h before checkin', '14:00', '12:00', 'Yes', 'WiFi, Breakfast');

INSERT INTO Accom_details (package_id, accom_id) VALUES
  (1, 1),
  (2, 2),
  (3, 3),
  (4, 4),
  (5, 5),
  (6, 6),
  (7, 7),
  (8, 8),
  (9, 9),
  (10, 10);

INSERT INTO Accom_booking (user_id, package_id, booking_id, accom_id) VALUES
  (1, 1, 1, 1),
  (2, 2, 1, 2),
  (3, 3, 1, 3),
  (4, 4, 1, 4),
  (5, 5, 1, 5),
  (6, 6, 1, 6),
  (7, 7, 1, 7),
  (8, 8, 1, 8),
  (9, 9, 1, 9),
  (10, 10, 1, 10);

INSERT INTO Transportation (trans_id, type_, carrier_company, classtype, available_seats, price, depart_location, dest_location, route_, depart_time, arrival_time, payment_method_accepted, cancellation_policy, amenities) VALUES
```

```
  (1, 'Bus', 'Carrier A', 'Economy', 40, 500.00, 'CityX', 'CityY',
'Route1', '08:00', '12:00', 'Cash, Card', '24h cancellation', 'WiFi'),
  (2, 'Train', 'Carrier B', 'AC', 100, 750.00, 'CityY', 'CityZ',
'Route2', '09:00', '13:00', 'Card', '48h cancellation', 'Meals'),
  (3, 'Flight', 'Airline A', 'Economy', 180, 2500.00, 'Airport1',
'Airport2', 'Route3', '06:00', '08:00', 'Card', 'Non-refundable',
'In-flight Entertainment'),
  (4, 'Bus', 'Carrier C', 'Business', 30, 600.00, 'CityA', 'CityB',
'Route4', '10:00', '14:00', 'Cash, Card', '24h cancellation', 'WiFi,
AC'),
  (5, 'Train', 'Carrier D', 'Sleeper', 120, 800.00, 'CityB', 'CityC',
'Route5', '07:00', '11:00', 'Card', '48h cancellation', 'Meals, AC'),
  (6, 'Flight', 'Airline B', 'Business', 150, 3500.00, 'Airport3',
'Airport4', 'Route6', '12:00', '15:00', 'Card', 'Refundable', 'Lounge
access'),
  (7, 'Bus', 'Carrier E', 'Economy', 50, 550.00, 'CityC', 'CityD',
'Route7', '11:00', '15:00', 'Cash', '24h cancellation', 'WiFi'),
  (8, 'Train', 'Carrier F', 'AC', 110, 780.00, 'CityD', 'CityE',
'Route8', '05:00', '09:00', 'Card', '48h cancellation', 'Meals'),
  (9, 'Flight', 'Airline C', 'Economy', 160, 2700.00, 'Airport5',
'Airport6', 'Route9', '14:00', '17:00', 'Card', 'Non-refundable',
'In-flight Entertainment'),
  (10, 'Bus', 'Carrier G', 'Business', 35, 620.00, 'CityE', 'CityF',
'Route10', '13:00', '17:00', 'Cash, Card', '24h cancellation', 'WiFi,
AC');

INSERT INTO Trans_details (package_id, trans_id) VALUES
  (1, 1),
  (2, 2),
  (3, 3),
  (4, 4),
  (5, 5),
  (6, 6),
  (7, 7),
  (8, 8),
  (9, 9),
  (10, 10);

INSERT INTO Trans_Booking (user_id, package_id, booking_id, trans_id)
VALUES
  (1, 1, 1, 1),
  (2, 2, 1, 2),
  (3, 3, 1, 3),
```

```sql
  (4, 4, 1, 4),
  (5, 5, 1, 5),
  (6, 6, 1, 6),
  (7, 7, 1, 7),
  (8, 8, 1, 8),
  (9, 9, 1, 9),
  (10, 10, 1, 10);

INSERT INTO ExpenseTracking (user_id, package_id, booking_id, guide_id,
date_time, amount, status_, category, payment_method, description_)
VALUES
  (1, 1, 1, 1, '2025-04-01 10:00:00', 500.00, 'Paid', 'Food', 'Card',
'Lunch expense'),
  (2, 2, 1, 2, '2025-04-02 11:00:00', 600.00, 'Pending', 'Transport',
'Cash', 'Taxi fare'),
  (3, 3, 1, 3, '2025-04-03 12:00:00', 700.00, 'Paid', 'Tickets',
'Card', 'Museum entry'),
  (4, 4, 1, 4, '2025-04-04 13:00:00', 800.00, 'Paid', 'Meal', 'Cash',
'Dinner expense'),
  (5, 5, 1, 5, '2025-04-05 14:00:00', 900.00, 'Pending', 'Guide',
'Card', 'Local guide fee'),
  (6, 6, 1, 6, '2025-04-06 15:00:00', 550.00, 'Paid', 'Transport',
'Cash', 'Bus fare'),
  (7, 7, 1, 7, '2025-04-07 16:00:00', 650.00, 'Paid', 'Food', 'Card',
'Snacks'),
  (8, 8, 1, 8, '2025-04-08 17:00:00', 750.00, 'Pending', 'Tickets',
'Cash', 'Attraction entry'),
  (9, 9, 1, 9, '2025-04-09 18:00:00', 850.00, 'Paid', 'Meal', 'Card',
'Lunch expense'),
  (10, 10, 1, 10, '2025-04-10 19:00:00', 950.00, 'Paid', 'Misc',
'Cash', 'Other expense');

INSERT INTO Group_members (member_id, name_, age, gender, contact_no)
VALUES
  (1, 'Member1', 25, 'Male', '1111000001'),
  (2, 'Member2', 30, 'Female', '1111000002'),
  (3, 'Member3', 22, 'Male', '1111000003'),
  (4, 'Member4', 28, 'Female', '1111000004'),
  (5, 'Member5', 35, 'Male', '1111000005'),
  (6, 'Member6', 27, 'Female', '1111000006'),
  (7, 'Member7', 32, 'Male', '1111000007'),
  (8, 'Member8', 24, 'Female', '1111000008'),
  (9, 'Member9', 29, 'Male', '1111000009'),
  (10, 'Member10', 31, 'Female', '1111000010');

INSERT INTO Traveller_info (user_id, package_id, booking_id, member_id)
VALUES
  (1, 1, 1, 1),
  (2, 2, 1, 2),
  (3, 3, 1, 3),
  (4, 4, 1, 4),
  (5, 5, 1, 5),
  (6, 6, 1, 6),
  (7, 7, 1, 7),
  (8, 8, 1, 8),
  (9, 9, 1, 9),
  (10, 10, 1, 10);

INSERT INTO Invoice_details (transaction_id, payment_method) VALUES
  (1, 'Card'),
  (2, 'Cash'),
  (3, 'Card'),
  (4, 'Online'),
  (5, 'Card'),
  (6, 'Cash'),
  (7, 'Online'),
  (8, 'Card'),
  (9, 'Cash'),
  (10, 'Online');

INSERT INTO Payment_done (user_id, package_id, booking_id,
transaction_id, inv_date, time_, paid, outstanding) VALUES
  (1, 1, 1, 1, '2025-04-01', '10:00', 4999.99, 0.00),
  (2, 2, 1, 2, '2025-04-02', '11:00', 6999.99, 0.00),
  (3, 3, 1, 3, '2025-04-03', '12:00', 8999.99, 0.00),
  (4, 4, 1, 4, '2025-04-04', '13:00', 5999.99, 0.00),
  (5, 5, 1, 5, '2025-04-05', '14:00', 10999.99, 0.00),
  (6, 6, 1, 6, '2025-04-06', '15:00', 3999.99, 0.00),
  (7, 7, 1, 7, '2025-04-07', '16:00', 4999.99, 0.00),
  (8, 8, 1, 8, '2025-04-08', '17:00', 7999.99, 0.00),
  (9, 9, 1, 9, '2025-04-09', '18:00', 5999.99, 0.00),
  (10, 10, 1, 10, '2025-04-10', '19:00', 9999.99, 0.00);

INSERT INTO Cancellation (cancellation_id, cancellation_fees) VALUES
  (1, 100.00),
  (2, 150.00),
  (3, 200.00),
  (4, 120.00),
  (5, 130.00),
  (6, 110.00),
  (7, 160.00),
  (8, 180.00),
  (9, 140.00),
  (10, 170.00);

INSERT INTO Cancelled (user_id, package_id, booking_id,
cancellation_id, date_, status_, reason, refund_amount) VALUES
  (1, 1, 1, 1, '2025-04-02', 'Processed', 'Change of plans', 4899.99),
  (2, 2, 1, 2, '2025-04-03', 'Processed', 'Personal reasons', 6849.99),
  (3, 3, 1, 3, '2025-04-04', 'Pending', 'Illness', 8799.99),
  (4, 4, 1, 4, '2025-04-05', 'Processed', 'Schedule conflict',
5879.99),
  (5, 5, 1, 5, '2025-04-06', 'Processed', 'Weather issues', 10869.99),
  (6, 6, 1, 6, '2025-04-07', 'Pending', 'Family emergency', 3889.99),
  (7, 7, 1, 7, '2025-04-08', 'Processed', 'Health reasons', 4889.99),
  (8, 8, 1, 8, '2025-04-09', 'Processed', 'Other', 7819.99),
  (9, 9, 1, 9, '2025-04-10', 'Pending', 'Change of plans', 5899.99),
  (10, 10, 1, 10, '2025-04-11', 'Processed', 'Personal reasons',
9829.99);

INSERT INTO Pack_Feedback (user_id, package_id, booking_id, rating,
review, date_) VALUES
  (1, 1, 1, 5, 'Excellent package!', '2025-04-02'),
  (2, 2, 1, 4, 'Very good experience', '2025-04-03'),
  (3, 3, 1, 3, 'Average service', '2025-04-04'),
  (4, 4, 1, 5, 'Loved the cultural touch', '2025-04-05'),
  (5, 5, 1, 4, 'Great safari!', '2025-04-06'),
  (6, 6, 1, 3, 'Could be better', '2025-04-07'),
  (7, 7, 1, 5, 'Delicious and fun', '2025-04-08'),
  (8, 8, 1, 4, 'Enjoyed the winter sports', '2025-04-09'),
  (9, 9, 1, 3, 'Good, but had issues', '2025-04-10'),
  (10, 10, 1, 5, 'A perfect getaway!', '2025-04-11');

INSERT INTO Agent_feedback (user_id, package_id, booking_id, agent_id,
rating, review, date_) VALUES
  (1, 1, 1, 1, 5, 'Very helpful agent', '2025-04-02'),
  (2, 2, 1, 2, 4, 'Responsive and kind', '2025-04-03'),
  (3, 3, 1, 3, 3, 'Okay service', '2025-04-04'),
  (4, 4, 1, 4, 5, 'Excellent experience', '2025-04-05'),
  (5, 5, 1, 5, 4, 'Good service', '2025-04-06'),
  (6, 6, 1, 6, 3, 'Could be improved', '2025-04-07'),
  (7, 7, 1, 7, 5, 'Outstanding assistance', '2025-04-08'),
  (8, 8, 1, 8, 4, 'Very professional', '2025-04-09'),
  (9, 9, 1, 9, 3, 'Satisfactory service', '2025-04-10'),
  (10, 10, 1, 10, 5, 'Great job', '2025-04-11');

INSERT INTO Guide_feedback (user_id, package_id, booking_id, guide_id,
rating, review, date_) VALUES
  (1, 1, 1, 1, 5, 'Amazing guide!', '2025-04-02'),
  (2, 2, 1, 2, 4, 'Very knowledgeable', '2025-04-03'),
  (3, 3, 1, 3, 3, 'Average guide', '2025-04-04'),
  (4, 4, 1, 4, 5, 'Perfect tour guide', '2025-04-05'),
  (5, 5, 1, 5, 4, 'Good insights', '2025-04-06'),
  (6, 6, 1, 6, 3, 'Not very engaging', '2025-04-07'),
  (7, 7, 1, 7, 5, 'Outstanding tour', '2025-04-08'),
  (8, 8, 1, 8, 4, 'Very informative', '2025-04-09'),
  (9, 9, 1, 9, 3, 'It was okay', '2025-04-10'),
  (10, 10, 1, 10, 5, 'Fantastic guide', '2025-04-11');

INSERT INTO Commission_earned_agent (user_id, package_id, booking_id,
agent_id, date_, details, amount) VALUES
  (1, 1, 1, 1, '2025-04-02', 'Commission for sale', 100.00),
  (2, 2, 1, 2, '2025-04-03', 'Commission for sale', 150.00),
  (3, 3, 1, 3, '2025-04-04', 'Commission for sale', 200.00),
  (4, 4, 1, 4, '2025-04-05', 'Commission for sale', 120.00),
  (5, 5, 1, 5, '2025-04-06', 'Commission for sale', 130.00),
  (6, 6, 1, 6, '2025-04-07', 'Commission for sale', 110.00),
  (7, 7, 1, 7, '2025-04-08', 'Commission for sale', 160.00),
  (8, 8, 1, 8, '2025-04-09', 'Commission for sale', 180.00),
  (9, 9, 1, 9, '2025-04-10', 'Commission for sale', 140.00),
  (10, 10, 1, 10, '2025-04-11', 'Commission for sale', 170.00);

INSERT INTO Commission_earned_guide (user_id, package_id, booking_id,
guide_id, date_, details, amount, chosen_by) VALUES
  (1, 1, 1, 1, '2025-04-02', 'Commission for guide', 90.00, 'Client'),
  (2, 2, 1, 2, '2025-04-03', 'Commission for guide', 95.00, 'Client'),
  (3, 3, 1, 3, '2025-04-04', 'Commission for guide', 85.00, 'Client'),
  (4, 4, 1, 4, '2025-04-05', 'Commission for guide', 100.00, 'Client'),
  (5, 5, 1, 5, '2025-04-06', 'Commission for guide', 105.00, 'Client'),
  (6, 6, 1, 6, '2025-04-07', 'Commission for guide', 80.00, 'Client'),
  (7, 7, 1, 7, '2025-04-08', 'Commission for guide', 110.00, 'Client'),
  (8, 8, 1, 8, '2025-04-09', 'Commission for guide', 115.00, 'Client'),
  (9, 9, 1, 9, '2025-04-10', 'Commission for guide', 90.00, 'Client'),
  (10, 10, 1, 10, '2025-04-11', 'Commission for guide', 120.00,
'Client');

INSERT INTO Preference_details (pref_id, preference_level,
interest_category, interest_details) VALUES
  (1, 'High', 'Adventure', 'Mountains and trekking'),
  (2, 'Medium', 'Culture', 'Historical sites'),
  (3, 'Low', 'Food', 'Local cuisines'),
  (4, 'High', 'Relaxation', 'Beach resorts'),
  (5, 'Medium', 'Wildlife', 'Safari and nature'),
  (6, 'Low', 'Shopping', 'Local markets'),
  (7, 'High', 'Sports', 'Water sports'),
  (8, 'Medium', 'Art', 'Museums and galleries'),
  (9, 'Low', 'Music', 'Local concerts'),
  (10, 'High', 'Nightlife', 'Clubs and bars');

INSERT INTO Preferences (user_id, pref_id) VALUES
  (1, 1),
  (2, 2),
  (3, 3),
  (4, 4),
  (5, 5),
  (6, 6),
  (7, 7),
  (8, 8),
  (9, 9),

INSERT INTO Promotion (promotion_id, type_, title, description_,
status_, startdate, enddate, conditions) VALUES
  ('PROMO1', 'Discount', 'Spring Sale', '20% off for early bookings',
'Active', '2025-03-01', '2025-03-31', 'Min booking $5000'),
  ('PROMO2', 'Cashback', 'Summer Cashback', '10% cashback on every
booking', 'Active', '2025-06-01', '2025-06-30', 'One time use'),
  ('PROMO3', 'Discount', 'Winter Special', '15% off on select
packages', 'Inactive', '2025-12-01', '2025-12-31', 'No combine'),
  ('PROMO4', 'Bonus', 'Holiday Bonus', 'Free meal vouchers', 'Active',
'2025-11-20', '2025-11-30', 'Applicable on weekends'),
  ('PROMO5', 'Discount', 'Flash Sale', '25% discount for 24 hours',
'Active', '2025-04-15', '2025-04-16', 'First come first served'),
  ('PROMO6', 'Cashback', 'New Year Offer', '5% cashback on all
bookings', 'Active', '2025-01-01', '2025-01-10', 'No minimum'),
  ('PROMO7', 'Discount', 'Festival Offer', '10% off during festival',
'Active', '2025-08-01', '2025-08-07', 'Limited seats'),
  ('PROMO8', 'Bonus', 'Referral Bonus', 'Earn extra points for
referrals', 'Active', '2025-05-01', '2025-05-31', 'Refer 2 friends'),
  ('PROMO9', 'Discount', 'Early Bird', 'Early bookings get 15% off',
'Active', '2025-02-01', '2025-02-28', 'Booking 30 days ahead'),
  ('PROMO10', 'Cashback', 'Loyalty Offer', 'Exclusive cashback for
repeat customers', 'Active', '2025-07-01', '2025-07-31', 'Only for
loyalty members');

INSERT INTO Discount_offer (user_id, package_id, booking_id,
promotion_id, offered_discount) VALUES
  (1, 1, 1, 'PROMO1', 20.00),
  (2, 2, 1, 'PROMO2', 10.00),
  (3, 3, 1, 'PROMO3', 15.00),
  (4, 4, 1, 'PROMO4', 0.00),
  (5, 5, 1, 'PROMO5', 25.00),
  (6, 6, 1, 'PROMO6', 5.00),
  (7, 7, 1, 'PROMO7', 10.00),
  (8, 8, 1, 'PROMO8', 0.00),
  (9, 9, 1, 'PROMO9', 15.00),
  (10, 10, 1, 'PROMO10', 0.00);

INSERT INTO Client_Agent_logs (user_id, agent_id, date_time, type_,
to_whom, subject_, status_, content) VALUES
  (1, 1, '2025-04-01 09:00:00', 'Inquiry', 'Agent', 'Booking Query',
'Resolved', 'Asked about package details'),
  (2, 2, '2025-04-02 09:15:00', 'Complaint', 'Agent', 'Delay Issue',
'Pending', 'Travel delay complaint'),
  (3, 3, '2025-04-03 09:30:00', 'Inquiry', 'Agent', 'Rate Info',
'Resolved', 'Asked about fare rates'),
  (4, 4, '2025-04-04 09:45:00', 'Feedback', 'Agent', 'Service
Feedback', 'Resolved', 'Positive feedback given'),
  (5, 5, '2025-04-05 10:00:00', 'Inquiry', 'Agent', 'Booking Process',
'Pending', 'Clarification on booking'),
  (6, 6, '2025-04-06 10:15:00', 'Complaint', 'Agent', 'Cancellation',
'Resolved', 'Issue with cancellation process'),
  (7, 7, '2025-04-07 10:30:00', 'Feedback', 'Agent', 'Service
Feedback', 'Resolved', 'Very satisfied with service'),
  (8, 8, '2025-04-08 10:45:00', 'Inquiry', 'Agent', 'Discounts',
'Pending', 'Asking about promo codes'),
  (9, 9, '2025-04-09 11:00:00', 'Feedback', 'Agent', 'Agent Behavior',
'Resolved', 'Polite and efficient'),
```

```sql
  (10, 10, '2025-04-10 11:15:00', 'Inquiry', 'Agent', 'Extra Info',
'Resolved', 'Requested additional package details');

INSERT INTO Client_Guide_logs (user_id, guide_id, date_time, type_,
to_whom, subject_, status_, content) VALUES
  (1, 1, '2025-04-01 12:00:00', 'Inquiry', 'Guide', 'Tour Schedule',
'Resolved', 'Asked about tour schedule'),
  (2, 2, '2025-04-02 12:15:00', 'Feedback', 'Guide', 'Guide Rating',
'Resolved', 'Complimented the guide'),
  (3, 3, '2025-04-03 12:30:00', 'Complaint', 'Guide', 'Late Start',
'Pending', 'Guide started late'),
  (4, 4, '2025-04-04 12:45:00', 'Inquiry', 'Guide', 'Itinerary Detail',
'Resolved', 'Requested detailed itinerary'),
  (5, 5, '2025-04-05 13:00:00', 'Feedback', 'Guide', 'Knowledgeable',
'Resolved', 'Very knowledgeable guide'),
  (6, 6, '2025-04-06 13:15:00', 'Complaint', 'Guide', 'Rudeness',
'Pending', 'Guide was not courteous'),
  (7, 7, '2025-04-07 13:30:00', 'Inquiry', 'Guide', 'Local Info',
'Resolved', 'Asked about local spots'),
  (8, 8, '2025-04-08 13:45:00', 'Feedback', 'Guide', 'Enthusiasm',
'Resolved', 'Very enthusiastic guide'),
  (9, 9, '2025-04-09 14:00:00', 'Complaint', 'Guide', 'Pace Too Fast',
'Pending', 'Tour was too rushed'),
  (10, 10, '2025-04-10 14:15:00', 'Inquiry', 'Guide', 'Accommodation
Tips', 'Resolved', 'Requested accommodation advice');

INSERT INTO Agent_Guide_logs (agent_id, guide_id, date_time, type_,
to_whom, subject_, status_, content) VALUES
  (1, 1, '2025-04-01 15:00:00', 'Collaboration', 'Guide',
'Coordination', 'Resolved', 'Coordinated itinerary details'),
  (2, 2, '2025-04-02 15:15:00', 'Meeting', 'Guide', 'Schedule Sync',
'Resolved', 'Scheduled meeting time'),
  (3, 3, '2025-04-03 15:30:00', 'Update', 'Guide', 'Client Feedback',
'Resolved', 'Shared client feedback'),
  (4, 4, '2025-04-04 15:45:00', 'Coordination', 'Guide', 'Tour
Details', 'Resolved', 'Discussed tour details'),
  (5, 5, '2025-04-05 16:00:00', 'Meeting', 'Guide', 'Booking Update',
'Pending', 'Booked a new package'),
  (6, 6, '2025-04-06 16:15:00', 'Update', 'Guide', 'Itinerary Change',
'Resolved', 'Changed tour itinerary'),
  (7, 7, '2025-04-07 16:30:00', 'Coordination', 'Guide', 'Client
Query', 'Resolved', 'Addressed client query'),
  (8, 8, '2025-04-08 16:45:00', 'Meeting', 'Guide', 'Discount
Discussion', 'Pending', 'Discussed client discount'),
  (9, 9, '2025-04-09 17:00:00', 'Update', 'Guide', 'Feedback Share',
'Resolved', 'Shared client feedback'),
  (10, 10, '2025-04-10 17:15:00', 'Coordination', 'Guide', 'Next Tour',
'Resolved', 'Planned next tour schedule');
```

Querys

1. List all available travel packages with price, duration, and best season

```
SELECT package_name, price, duration, best_season
FROM Packages
WHERE availability_ = ‘Yes';
```

2. Get all bookings done by a particular user (with their name and package booked)

```
SELECT u.user_id, pb.booking_id, p.package_name, pb.booking_date,
pb.booking_status
FROM Package_Booking pb
JOIN LoggedInUser u ON u.user_id = pb.user_id
JOIN Packages p ON p.package_id = pb.package_id
WHERE u.user_id = 1; -- replace with actual user_id
```

3. Get the guides who know 'French'

```
SELECT g.guide_id, g.specialization, g.yrs_of_experience
FROM Guide g
JOIN Languages_Known lk ON g.guide_id = lk.guide_id
WHERE lk.language_ = 'French';
```

4. Show feedback for a particular package

```
SELECT pf.rating, pf.review, pf.date_
FROM Pack_Feedback pf
WHERE pf.package_id = 1; -- replace with actual package_id
```

5. Find top 5 most expensive packages

```
SELECT package_name, price
FROM Packages
ORDER BY price DESC
```

```
LIMIT 5;
```

6. Total amount paid by a user for a booking

```
SELECT SUM(pd.paid) AS total_paid
FROM Payment_done pd
WHERE pd.user_id = 1 AND pd.booking_id = 1;
```

7. Agents and the total commission they earned

```
SELECT agent_id, SUM(amount) AS total_commission
FROM Commission_earned_agent
GROUP BY agent_id
ORDER BY total_commission DESC;
```

8. List all accommodations used in a specific package

```
SELECT a.name_, a.type_, a.room_type, a.room_price
FROM Accom_details ad
JOIN Accommodation a ON ad.accom_id = a.accom_id
WHERE ad.package_id = 6;
```

9. List of users who gave feedback to a specific guide

```
SELECT u.username, gf.rating, gf.review
FROM Guide_feedback gf
JOIN User_ u ON gf.user_id = u.user_id
WHERE gf.guide_id = 1;
```

10. Languages known by each guide

```
SELECT g.guide_id, g.specialization, array_agg(l.language_) AS languages
FROM Guide g
JOIN Languages_Known l ON g.guide_id = l.guide_id
GROUP BY g.guide_id, g.specialization;
```

More Advanced Quries

11. Top 3 Highest Rated Packages (by average user rating)

```
SELECT package_id, AVG(rating) AS avg_rating
FROM Pack_Feedback
GROUP BY package_id
ORDER BY avg_rating DESC
LIMIT 3;
```

12. Total Revenue Generated by Each Package

```
SELECT p.package_id, pk.package_name, SUM(pay.paid) AS total_revenue
FROM Payment_done pay
JOIN Packages pk ON pk.package_id = pay.package_id
JOIN Package_Booking p ON p.booking_id = pay.booking_id
GROUP BY p.package_id, pk.package_name
ORDER BY total_revenue DESC;
```

13. Guides With Average Rating > 4.5 and More Than 5 Reviews

```
SELECT guide_id, AVG(rating) AS avg_rating, COUNT(*) AS total_reviews
FROM Guide_feedback
GROUP BY guide_id
HAVING AVG(rating) > 4.5 AND COUNT(*) > 5;
```

14. Most Popular Transportation Mode

```
SELECT t.type_ AS transport_type, COUNT(*) AS used_count
FROM Transportation t
JOIN Trans_details td ON t.trans_id = td.trans_id
GROUP BY t.type_
ORDER BY used_count DESC
LIMIT 1;
```

1**15. List Packages That Include Both Accommodation AND Transportation**

```
SELECT p.package_id, p.package_name
FROM Packages p
WHERE p.package_id IN (SELECT package_id FROM Accom_details)
AND p.package_id IN (SELECT package_id FROM Trans_details);
```

16. Monthly Booking Summary (Current Year)

```
SELECT
   EXTRACT(MONTH FROM booking_date) AS month,
   COUNT(*) AS total_bookings,
   SUM(pk.price) AS total_revenue
FROM Package_Booking pb
JOIN Packages pk ON pb.package_id = pk.package_id
WHERE EXTRACT(YEAR FROM booking_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
GROUP BY EXTRACT(MONTH FROM booking_date) ORDER BY month;
```

17. Users Who Have Completed Full Payment (No Outstanding Amount)

```
SELECT DISTINCT user_id
FROM Payment_done
WHERE outstanding = 10;
```

18. Users Who Have Cancelled More Than 2 Times

```
SELECT user_id, COUNT(*) AS cancel_count
FROM Cancelled
GROUP BY user_id
HAVING COUNT(*) > 2;
```

19. Running Total of Payment Made by a User Across Bookings (Window Function)

```
SELECT
user_id,
booking_id,
paid,
SUM(paid) OVER (PARTITION BY user_id ORDER BY booking_id) AS
running_total_paid
FROM Payment_done;
```

20. Most Frequently Chosen Interest Category

```
SELECT interest_category, COUNT(*) AS preference_count
FROM Preference_details
GROUP BY interest_category
ORDER BY preference_count DESC
LIMIT 1;
```

21. CTE: Packages With Above Average Price

```
WITH avg_price AS (
SELECT AVG(price) AS avg_val FROM Packages
)
SELECT package_id, package_name, price
FROM Packages, avg_price
WHERE price > avg_val;
```

22. Packages Never Booked

```
SELECT package_id, package_name
```

```
FROM Packages
WHERE package_id IN (
SELECT DISTINCT package_id FROM Package_Booking
);
```