DataBase Project Report

Course Details:

Course Number: CSE 3110

Course Name: Database Systems Laboratory

Project Name: TravelMate- A Tourguide Finder Database Project

Submitted to:

Information	Details
Name	Md. Shahidul Salim
Position	**Lecturer**
Department	**Computer Science & Engineering**
Institution	**Khulna University of Engineering & Technology**

Information	Details
Name	**Nazia Jahan Khan Chowdhury**
Position	Assistant Professor
Department	**Computer Science & Engineering**
Institution	**Khulna University of Engineering & Technology**

Submitted by:

Mofazzal Hosen
Roll: 2007074

Group: B1 **Batch:2k20**

Dept of Computer Science & Engineering,
Khulna University of Engineering & Technology

Project Files Link: Github Link
URL:
https://github.com/Mofazzal874/CSE3110_DataBase_project

Introduction

Project Overview:

The main project is about finding tourguide in a specific places , tourist spot or in any area where guides are rare.

- The main aim of this project is it gives information about the Functioning of Databases in A Travel Booking system.
- All the Functionalities of MultiUser are credited in this project.

Project Objectives:

1. **Design a Comprehensive Relational Database Schema**:
   - Develop a relational database schema that efficiently organizes and stores travel-related data, including user information, tourist places, tour guides, hotels, agencies, and packages.
2. **SQL Queries for Data Management**:
   - Create SQL queries to facilitate the retrieval, updating, and management of data within the database.
   - Enable functionalities such as adding new records, updating existing information, and retrieving detailed data based on specific criteria.
3. **Triggers for Automated Actions**:
   - Design and implement triggers to automate certain actions based on database events, such as updating user spending totals or enforcing business rules.
4. **Nested Subqueries for Complex Data Retrieval**:
   - Implement nested subqueries to perform complex data retrieval operations, such as retrieving information based on multiple conditions or aggregating data from multiple tables.
5. **Implement Loops for Iterative Processing**:
   - Utilize loops within SQL or PL/SQL code to perform iterative processing tasks, such as processing multiple records or performing batch updates.

Rules Governing the TravelMate Project:

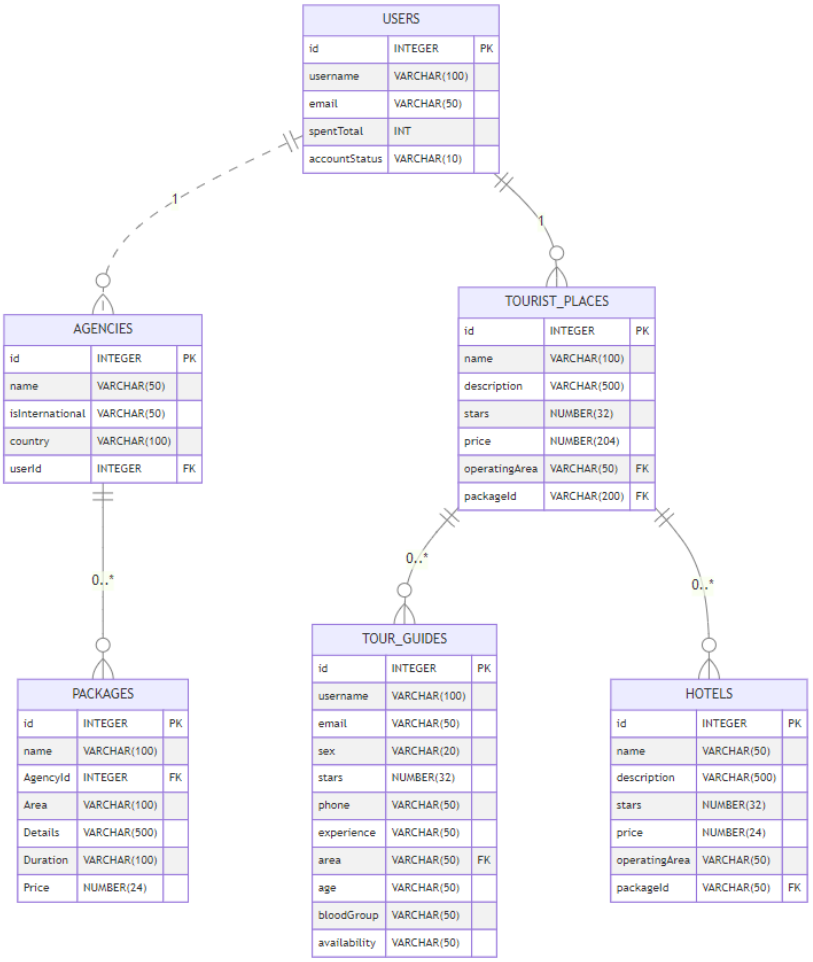
1. **Unique User Identifiers**:
   - Each user registered in the TravelMate system must have a unique identifier, such as a user ID.
2. **Mandatory User Information**:
   - Essential user information fields, such as username and email, must be provided during registration and cannot be left empty.
3. **Minimum Spent Total Requirement**:
   - Users must have a minimum total amount spent (e.g., 0) in order to be considered active within the TravelMate system.

Entities and Attributes for TravelMate:

- **TouristPlaces:**
  - Place_ID (Primary Key),
  - Name,
  - Description,
  - Stars,
  - Price,
  - OperatingArea

- **Users:**
  - User_ID (Primary Key),
  - Username,
  - Email,
  - SpentTotal,
  - AccountStatus
- **TourGuides:**
  - Guide_ID (Primary Key),
  - Username,
  - Email,
  - Sex,
  - Stars,
  - Phone,
  - Experience,
  - Area,
  - Age,
  - BloodGroup,
  - Availability
- **Hotels:**
  - Hotel_ID (Primary Key),
  - Name,
  - Description,
  - Stars,
  - Price,
  - OperatingArea
- **Agencies:**
  - Agency_ID (Primary Key),
  - Name,
  - IsInternational,
  - Country,
  - User_ID (Foreign Key referencing Users)
- **Packages:**
  - Package_ID (Primary Key),
  - Name,
  - Agency_ID (Foreign Key referencing Agencies),
  - Area,
  - Details,
  - Duration,
  - Price

ER Diagram:



Database Table Creation :

```
-- Users table
CREATE TABLE Users (
    id INTEGER PRIMARY KEY,
    username VARCHAR2(100) NOT NULL,
    email VARCHAR2(50) NOT NULL,
    spentTotal INT,
    accountStatus VARCHAR2(10) CHECK (accountStatus IN ('active', 'inactive', 'pending')
);

-- TouristPlaces table
CREATE TABLE TouristPlaces (
    id INTEGER PRIMARY KEY,
```

```
    name VARCHAR2(100),
    description VARCHAR2(500),
    stars NUMBER(32),
    price NUMBER(20, 4),
    operatingArea VARCHAR2(50),
    packageId VARCHAR2(200),
    FOREIGN KEY (packageId) REFERENCES Packages(id)
);

-- TourGuides table
CREATE TABLE TourGuides (
    id INTEGER PRIMARY KEY,
    username VARCHAR2(100) NOT NULL,
    email VARCHAR2(50),
    sex VARCHAR2(20),
    stars NUMBER(32),
    phone VARCHAR2(50),
    experience VARCHAR2(50),
    area VARCHAR2(50),
    FOREIGN KEY (area) REFERENCES TouristPlaces(id),
    age VARCHAR2(50),
    bloodGroup VARCHAR2(50),
    availability VARCHAR2(50)
);

-- Hotels table
CREATE TABLE Hotels (
    id INTEGER PRIMARY KEY,
    name VARCHAR2(50),
    description VARCHAR2(500),
    stars NUMBER(32),
    price NUMBER(24),
    operatingArea VARCHAR2(50),
    packageId VARCHAR2(50),
    FOREIGN KEY (packageId) REFERENCES Packages(id)
);

-- Agencies table
CREATE TABLE Agencies (
    id INTEGER PRIMARY KEY,
    name VARCHAR2(50),
    isInternational VARCHAR2(50) CHECK (isInternational IN ('YES', 'NO')),
    country VARCHAR2(100),
    userId INTEGER,
    FOREIGN KEY (userId) REFERENCES Users(id)
);

-- Packages table
CREATE TABLE Packages (
    id INTEGER PRIMARY KEY,
    name VARCHAR2(100),
    AgencyId INTEGER,
    FOREIGN KEY (AgencyId) REFERENCES Agencies(id),
    Area VARCHAR2(100),
    Details VARCHAR2(500),
```

```
    Duration VARCHAR2(100),
    Price NUMBER(24)
);
```

Database Table Data Insertion :

For `Users` table:

```
INSERT INTO Users (id, username, email, spentTotal, accountStatus)
VALUES (1, 'mofazzal874', 'mofa@gmail.com', 200, 'active');

INSERT INTO Users (id, username, email, spentTotal, accountStatus)
VALUES (2, 'john_doe', 'john@example.com', 150, 'inactive');

INSERT INTO Users (id, username, email, spentTotal, accountStatus)
VALUES (3, 'jane_smith', 'jane@example.com', 300, 'active');

INSERT INTO Users (id, username, email, spentTotal, accountStatus)
VALUES (4, 'alice123', 'alice@example.com', 100, 'pending');

INSERT INTO Users (id, username, email, spentTotal, accountStatus)
VALUES (5, 'bob456', 'bob@example.com', 50, 'active');
```

2.For `Agencies` table:

```
INSERT INTO Agencies (id, name, isInternational, country, userId)
VALUES (1, 'TravelWorld', 'YES', 'United States', 1);

INSERT INTO Agencies (id, name, isInternational, country, userId)
VALUES (2, 'Adventure Seekers', 'NO', 'Australia', 3);

INSERT INTO Agencies (id, name, isInternational, country, userId)
VALUES (3, 'EuroTrip Tours', 'YES', 'France', 2);

INSERT INTO Agencies (id, name, isInternational, country, userId)
VALUES (4, 'Inca Treks', 'NO', 'Peru', 4);

INSERT INTO Agencies (id, name, isInternational, country, userId)
VALUES (5, 'Royal India Tours', 'YES', 'India', 5);
```

3.For `Packages` table:

```
INSERT INTO Packages (id, name, AgencyId, Area, Details, Duration, Price)
VALUES (1, 'Paris Adventure', 1, 'Paris', 'Explore the best of Paris', '5 days', 1000.
00);
```

```
INSERT INTO Packages (id, name, AgencyId, Area, Details, Duration, Price)
VALUES (2, 'Grand Canyon Experience', 2, 'Arizona', 'Experience the wonders of the Gra
nd Canyon', '4 days', 800.00);

INSERT INTO Packages (id, name, AgencyId, Area, Details, Duration, Price)
VALUES (3, 'Great Barrier Reef Getaway', 3, 'Queensland', 'Discover the beauty of the
Great Barrier Reef', '7 days', 1500.00);

INSERT INTO Packages (id, name, AgencyId, Area, Details, Duration, Price)
VALUES (4, 'Machu Picchu Expedition', 4, 'Cusco', 'Trek to the ancient ruins of Machu
Picchu', '6 days', 1200.00);

INSERT INTO Packages (id, name, AgencyId, Area, Details, Duration, Price)
VALUES (5, 'Taj Mahal Tour', 5, 'Agra', 'Visit the iconic Taj Mahal and other attracti
ons', '3 days', 600.00);
```

4.For `TouristPlaces` table:

```
INSERT INTO TouristPlaces (id, name, description, stars, price, operatingArea, package
Id)
VALUES (1, 'Eiffel Tower', 'Iconic iron tower with a viewing deck', 4.5, 25.00, 'Pari
s', 1);

INSERT INTO TouristPlaces (id, name, description, stars, price, operatingArea, package
Id)
VALUES (2, 'Grand Canyon', 'Vast canyon with colorful rock formations', 4.8, 30.00, 'A
rizona', 2);

INSERT INTO TouristPlaces (id, name, description, stars, price, operatingArea, package
Id)
VALUES (3, 'Great Barrier Reef', 'World\'s largest coral reef system', 4.7, 50.00, 'Qu
eensland', 3);

INSERT INTO TouristPlaces (id, name, description, stars, price, operatingArea, package
Id)
VALUES (4, 'Machu Picchu', 'Incan citadel set high in the Andes Mountains', 4.9, 40.0
0, 'Cusco', 4);

INSERT INTO TouristPlaces (id, name, description, stars, price, operatingArea, package
Id)
VALUES (5, 'Taj Mahal', 'Ivory-white marble mausoleum on the Yamuna river', 4.6, 35.0
0, 'Agra', 5);
```

5.For `TourGuides` table:

```
INSERT INTO TourGuides (id, username, email, sex, stars, phone, experience, areaId, ag
e, bloodGroup, availability)
VALUES (1, 'guide1', 'guide1@example.com', 'Male', 4.7, '+123456789', '5 years', 1, '3
0', 'O+', 'Available');
```

```
INSERT INTO TourGuides (id, username, email, sex, stars, phone, experience, areaId, ag
e, bloodGroup, availability)
VALUES (2, 'guide2', 'guide2@example.com', 'Female', 4.9, '+987654321', '8 years', 2,
'35', 'A-', 'Available');

INSERT INTO TourGuides (id, username, email, sex, stars, phone, experience, areaId, ag
e, bloodGroup, availability)
VALUES (3, 'guide3', 'guide3@example.com', 'Male', 4.5, '+111222333', '3 years', 3, '2
8', 'AB+', 'Not Available');

INSERT INTO TourGuides (id, username, email, sex, stars, phone, experience, areaId, ag
e, bloodGroup, availability)
VALUES (4, 'guide4', 'guide4@example.com', 'Female', 4.8, '+444555666', '6 years', 4,
'32', 'B+', 'Available');

INSERT INTO TourGuides (id, username, email, sex, stars, phone, experience, areaId, ag
e, bloodGroup, availability)
VALUES (5, 'guide5', 'guide5@example.com', 'Male', 4.6, '+777888999', '4 years', 5, '2
7', 'O-', 'Available');
```

6. For `Hotels` table:

```
INSERT INTO Hotels (id, name, description, stars, price, operatingArea, packageId)
VALUES (1, 'Hotel Paris', 'Luxury hotel in the heart of Paris', 4.7, 150.00, 'Paris',
1);

INSERT INTO Hotels (id, name, description, stars, price, operatingArea, packageId)
VALUES (2, 'Grand Canyon Lodge', 'Rustic lodge with canyon views', 4.5, 100.00, 'Arizo
na', 2);

INSERT INTO Hotels (id, name, description, stars, price, operatingArea, packageId)
VALUES (3, 'Great Barrier Reef Resort', 'Beachfront resort on the Great Barrier Reef',
4.8, 200.00, 'Queensland', 3);

INSERT INTO Hotels (id, name, description, stars, price, operatingArea, packageId)
VALUES (4, 'Machu Picchu Hotel', 'Charming hotel near Machu Picchu', 4.6, 120.00, 'Cus
co', 4);

INSERT INTO Hotels (id, name, description, stars, price, operatingArea, packageId)
VALUES (5, 'Taj Mahal Palace', 'Historic luxury hotel overlooking the Taj Mahal', 4.9,
180.00, 'Agra', 5);
```

Database Tables in ORACLE XE APPlication Express:

Users Table

ID	USERNAME	EMAIL	SPENTTOTAL	ACCOUNTSTATUS
1	mofazzal874	mofa@gmail.com	200	active
2	john_doe	john@example.com	150	inactive
3	jane_smith	jane@example.com	300	active
4	alice123	alice@example.com	100	pending
5	bob456	bob@example.com	50	active

Agencies

ID	NAME	ISINTERNATIONAL	COUNTRY	USERID
1	TravelWorld	YES	United States	1
2	Adventure Seekers	NO	Australia	3
3	EuroTrip Tours	YES	France	2
4	Inca Treks	NO	Peru	4
5	Royal India Tours	YES	India	5

Packages

ID	NAME	AGENCYID	AREA	DETAILS	DURATION	PRICE
1	Paris Adventure	1	Paris	Explore the best of Paris	5 days	1000
2	Grand Canyon Experience	2	Arizona	Experience the wonders of the Grand Canyon	4 days	800
3	Great Barrier Reef Getaway	3	Queensland	Discover the beauty of the Great Barrier Reef	7 days	1500
4	Machu Picchu Expedition	4	Cusco	Trek to the ancient ruins of Machu Picchu	6 days	1200
5	Taj Mahal Tour	5	Agra	Visit the iconic Taj Mahal and other attractions	3 days	600

TourGuides

ID	USERNAME	EMAIL	SEX	STARS	PHONE	EXPERIENCE	AREAID	AGE	BLOODGROUP	AVAILABILITY
1	guide1	guide1@example.com	Male	5	+123456789	5 years	1	30	O+	Available
2	guide2	guide2@example.com	Female	5	+987654321	8 years	2	35	A-	Available
4	guide4	guide4@example.com	Female	5	+444555666	6 years	4	32	B+	Available
5	guide5	guide5@example.com	Male	5	+777888999	4 years	5	27	O-	Available

Hotels

ID	NAME	DESCRIPTION	STARS	PRICE	OPERATINGAREA	PACKAGEID
1	Hotel Paris	Luxury hotel in the heart of Paris	5	150	Paris	1
2	Grand Canyon Lodge	Rustic lodge with canyon views	5	100	Arizona	2
3	Great Barrier Reef Resort	Beachfront resort on the Great Barrier Reef	5	200	Queensland	3
4	Machu Picchu Hotel	Charming hotel near Machu Picchu	5	120	Cusco	4
5	Taj Mahal Palace	Historic luxury hotel overlooking the Taj Mahal	5	180	Agra	5

TouristPlaces

ID	NAME	DESCRIPTION	STARS	PRICE	OPERATINGAREA	PACKAGEID
1	Eiffel Tower	Iconic iron tower with a viewing deck	5	25	Paris	1
2	Grand Canyon	Vast canyon with colorful rock formations	5	30	Arizona	2
4	Machu Picchu	Incan citadel set high in the Andes Mountains	5	40	Cusco	4
5	Taj Mahal	Ivory-white marble mausoleum on the Yamuna river	5	35	Agra	5

Basic SQL Operations

1.Retriving Some information based on a value

```
select username, email from users where accountStatus = 'active';
```

USERNAME	EMAIL
mofazzal874	mofa@gmail.com
jane_smith	jane@example.com
bob456	bob@example.com

2.Retrieve the name, description, and stars of tourist places where operatingArea = 'Paris':

```
select name, description, stars from touristplaces where operatingArea = 'Paris';
```

NAME	DESCRIPTION	STARS
Eiffel Tower	Iconic iron tower with a viewing deck	5

Nested Sub Queries

3.Find all the detailed information of the agency who has a package named "Paris Adventure":

```
SELECT * FROM Agencies
WHERE id = (
    SELECT AgencyId FROM Packages
    WHERE name = 'Paris Adventure'
);
```

ID	NAME	ISINTERNATIONAL	COUNTRY	USERID
1	TravelWorld	YES	United States	1

4.Retrieve the name and email of the user who booked a package with a price greater than 1000:

```
SELECT username, email FROM Users
WHERE id IN (
    SELECT userId FROM Agencies
    WHERE id IN (
        SELECT AgencyId FROM Packages
        WHERE Price > 1000
    )
);
```

USERNAME	EMAIL
john_doe	john@example.com
alice123	alice@example.com

5.Retrieve the username and phone number of tour guides working in tourist places with more than 4 stars:

```
SELECT username, phone FROM TourGuides
WHERE areaId IN (
    SELECT id FROM TouristPlaces
    WHERE stars > 4
);
```

USERNAME	PHONE
guide1	+123456789
guide2	+987654321
guide4	+444555666
guide5	+777888999

6.Retrieve the all of users who booked a package with a price greater than the average package price:

```
SELECT * FROM Users
WHERE id IN (
    SELECT userId FROM Agencies
    WHERE id IN (
        SELECT AgencyId FROM Packages
        WHERE Price > (
            SELECT AVG(Price) FROM Packages
        )
    )
);
```

ID	USERNAME	EMAIL	SPENTTOTAL	ACCOUNTSTATUS
2	john_doe	john@example.com	150	inactive
4	alice123	alice@example.com	100	pending

7.Show the all info of tourist places offered by agencies based in countries where the average hotel stars are greater than 4.5:

```
SELECT * FROM TouristPlaces
WHERE packageId IN (
    SELECT id FROM Packages
    WHERE AgencyId IN (
        SELECT id FROM Agencies
        WHERE country IN (
            SELECT country FROM Hotels
            GROUP BY country
            HAVING AVG(stars) > 4.5
        )
    )
);
```

ID	NAME	DESCRIPTION	STARS	PRICE	OPERATINGAREA	PACKAGEID
1	Eiffel Tower	Iconic iron tower with a viewing deck	5	25	Paris	1
2	Grand Canyon	Vast canyon with colorful rock formations	5	30	Arizona	2
4	Machu Picchu	Incan citadel set high in the Andes Mountains	5	40	Cusco	4
5	Taj Mahal	Ivory-white marble mausoleum on the Yamuna river	5	35	Agra	5

8.Show the name and country of agencies that offer packages to tourist places with an average star rating greater than 4.7:

```
SELECT name, country FROM Agencies
WHERE id IN (
    SELECT AgencyId FROM Packages
    WHERE Area IN (
        SELECT Area FROM TouristPlaces
        GROUP BY Area
        HAVING AVG(stars) > 4.7
    )
);
```

NAME	COUNTRY
TravelWorld	United States
Adventure Seekers	Australia
EuroTrip Tours	France
Inca Treks	Peru
Royal India Tours	India

5 rows returned in 0.01 seconds Download

```
SELECT * FROM Users WHERE username = 'mofazzal874';
```

ID	USERNAME	EMAIL	SPENTTOTAL	ACCOUNTSTATUS
1	mofazzal874	mofa@gmail.com	200	active

Aggregate Function

1. Find the average price of packages offered by each agency:

```
SELECT AgencyId, AVG(Price) AS average_price
FROM Packages
GROUP BY AgencyId;
```

AGENCYID	AVERAGE_PRICE
1	1000
2	800
4	1200
5	600
3	1500

5 rows returned in 0.01 seconds Download

2.Count the number of users with each account status:

```
SELECT accountStatus, COUNT(*) AS user_count
FROM Users
GROUP BY accountStatus;
```

ACCOUNTSTATUS	USER_COUNT
pending	1
active	3
inactive	1

PL/SQL

1. PL/SQL variable declaration and print value for retrieving information about a specific user:

```
SET SERVEROUTPUT ON
DECLARE
    v_username Users.username%TYPE;
    v_email Users.email%TYPE;
    v_spent_total Users.spentTotal%TYPE;
    v_account_status Users.accountStatus%TYPE;
BEGIN
    SELECT username, email, spentTotal, accountStatus
    INTO v_username, v_email, v_spent_total, v_account_status
    FROM Users
    WHERE id = 1; -- Specify the user ID here
    DBMS_OUTPUT.PUT_LINE('Username: ' || v_username);
    DBMS_OUTPUT.PUT_LINE('Email: ' || v_email);
    DBMS_OUTPUT.PUT_LINE('Spent Total: ' || v_spent_total);
    DBMS_OUTPUT.PUT_LINE('Account Status: ' || v_account_status);
END;
/
```

```
Username: mofazzal874
Email: mofa@gmail.com
Spent Total: 200
Account Status: active

PL/SQL procedure successfully completed.
```

2.Insert and set default values for adding a new agency:

```
SET SERVEROUTPUT ON
DECLARE
    v_id Agencies.id%TYPE := 16; -- Specify the agency ID here
    v_name Agencies.name%TYPE := 'Global Adventures';
    v_is_international Agencies.isInternational%TYPE := 'YES';
    v_country Agencies.country%TYPE := 'United States';
    v_user_id Agencies.userId%TYPE := 11; -- Specify the user ID here
BEGIN
    INSERT INTO Agencies (id, name, isInternational, country, userId)
    VALUES (v_id, v_name, v_is_international, v_country, v_user_id);
    DBMS_OUTPUT.PUT_LINE('New agency inserted successfully.');
END;
/
```

```
New agency inserted successfully.

PL/SQL procedure successfully completed.
```

3.Cursor and row count for fetching and displaying information about all packages:

```
SET SERVEROUTPUT ON
DECLARE
    v_id Packages.id%TYPE;
    v_name Packages.name%TYPE;
    v_agency_id Packages.AgencyId%TYPE;
    v_area Packages.Area%TYPE;
    v_details Packages.Details%TYPE;
    v_duration Packages.Duration%TYPE;
    v_price Packages.Price%TYPE;
    v_row_count INTEGER := 0;
BEGIN
    FOR package_rec IN (SELECT * FROM Packages) LOOP
        v_id := package_rec.id;
        v_name := package_rec.name;
        v_agency_id := package_rec.AgencyId;
        v_area := package_rec.Area;
        v_details := package_rec.Details;
        v_duration := package_rec.Duration;
        v_price := package_rec.Price;

        DBMS_OUTPUT.PUT_LINE('ID: ' || v_id || ', Name: ' || v_name || ', Agency I
D: ' || v_agency_id || ', Area: ' || v_area || ', Details: ' || v_details || ', Dur
ation: ' || v_duration || ', Price: ' || v_price);
        v_row_count := v_row_count + 1;
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('Total rows fetched: ' || v_row_count);
END;
/
```

```
ID: 1, Name: Paris Adventure, Agency ID: 1, Area: Paris, Details: Explore the
best of Paris, Duration: 5 days, Price: 1000
ID: 2, Name: Grand Canyon Experience, Agency ID: 2, Area: Arizona, Details:
Experience the wonders of the Grand Canyon, Duration: 4 days, Price: 800
ID: 3, Name: Great Barrier Reef Getaway, Agency ID: 3, Area: Queensland,
Details: Discover the beauty of the Great Barrier Reef, Duration: 7 days, Price:
1500
ID: 4, Name: Machu Picchu Expedition, Agency ID: 4, Area: Cusco, Details: Trek
to the ancient ruins of Machu Picchu, Duration: 6 days, Price: 1200
ID: 5, Name: Taj Mahal Tour, Agency ID: 5, Area: Agra, Details: Visit the iconic
Taj Mahal and other attractions, Duration: 3 days, Price: 600
ID: 6, Name: London Excursion, Agency ID: 6, Area: London, Details: Explore the
vibrant city of London, Duration: 4 days, Price: 900
ID: 7, Name: Tokyo Discovery, Agency ID: 7, Area: Tokyo, Details: Discover the
bustling streets of Tokyo, Duration: 5 days, Price: 1200
ID: 8, Name: Rio Carnival Experience, Agency ID: 8, Area: Rio de Janeiro,
Details: Experience the vibrant Rio Carnival, Duration: 7 days, Price: 1800
ID: 9, Name: African Safari Adventure, Agency ID: 9, Area: Serengeti, Details:
Embark on a thrilling African safari, Duration: 6 days, Price: 1500
ID: 10, Name: Outback Wilderness Trek, Agency ID: 10, Area: Australian Outback,
Details: Trek through the rugged Australian Outback, Duration: 8 days, Price:
2000
ID: 11, Name: Great Wall Expedition, Agency ID: 11, Area: Beijing, Details:
Explore the Great Wall of China, Duration: 3 days, Price: 700
ID: 12, Name: Italian Riviera Retreat, Agency ID: 12, Area: Cinque Terre,
Details: Relax on the Italian Riviera, Duration: 6 days, Price: 1200
ID: 13, Name: Outback Adventure, Agency ID: 13, Area: Australian Outback,
Details: Experience the Australian wilderness, Duration: 5 days, Price: 1500
ID: 14, Name: Andean Discovery, Agency ID: 14, Area: Peru, Details: Discover the
Andes Mountains, Duration: 7 days, Price: 1700
ID: 15, Name: Canadian Rockies Tour, Agency ID: 15, Area: Alberta, Details:
Explore the stunning Canadian Rockies, Duration: 4 days, Price: 1100
Total rows fetched: 15
```

Trigger

1. Create a trigger to update the total spent amount of a user when a new package is purchased:

```
CREATE OR REPLACE TRIGGER update_spent_total
AFTER INSERT ON Packages
FOR EACH ROW
BEGIN
    UPDATE Users
    SET spentTotal = spentTotal + :new.Price
    WHERE id = (SELECT userId FROM Agencies WHERE id = :new.AgencyId);
END;
/


UPDATE Packages
SET name = 'Luxury Vacation', AgencyId = 1, Area = 'Tropical Paradise', Details =
'All-inclusive luxury vacation package', Duration = '7 days', Price = 2500
WHERE id = 1;
```

```
SQL> UPDATE Packages
  2  SET name = 'Luxury Vacation', AgencyId = 1, Area = 'Tropical Paradise', Details = 'All-inclusive luxury vacation package', Duration = '7 days', Price =
 2500
  3  WHERE id = 1;

1 row updated.
```

3.Create a trigger to automatically increment the number of users when a new agency is added:

```
CREATE OR REPLACE TRIGGER increment_user_count
AFTER INSERT ON Agencies
FOR EACH ROW
BEGIN
    UPDATE Agencies
    SET userId = userId + 1
    WHERE id = :new.id;
END;
/

UPDATE Agencies
SET userId = userId + 1
WHERE id = (SELECT MAX(id) FROM Agencies);
```

```
SQL> UPDATE Agencies
  2  SET userId = userId + 1
  3  WHERE id = (SELECT MAX(id) FROM Agencies);

1 row updated.
```

Conclusion

Challenges:

1. **Data Complexity**: Managing diverse data types and relationships between entities such as users, tourist places, tour guides, hotels, agencies, and packages can be complex.
2. **Data Integrity**: Ensuring data accuracy and consistency, especially with transactions and updates across multiple tables.
3. **Performance Optimization**: Optimizing database performance to handle large volumes of data and complex queries efficiently.
4. **Security**: Implementing robust security measures to protect sensitive user information and prevent unauthorized access.
5. **Scalability**: Designing the database to accommodate future growth and expansion of the travel management system.

Significance:

1. **Streamlined Travel Management**: The database facilitates efficient management of travel-related information, providing users with a seamless experience when planning and booking trips.
2. **Data-driven Decision Making**: By centralizing travel data, the system enables agencies to analyze customer preferences, booking trends, and performance metrics to make informed business decisions.
3. **Enhanced User Experience**: Users can easily access and retrieve information about tourist places, hotels, and packages, helping them make informed decisions and plan their trips effectively.
4. **Improved Operational Efficiency**: Automation of processes such as booking, payment processing, and customer management streamlines operations and reduces manual effort.

5. **Competitive Advantage**: A well-designed and functional database system can give travel agencies a competitive edge by offering personalized services, optimizing pricing strategies, and delivering superior customer experiences.

Use Cases:

1. **Booking Management**: Users can search for and book tourist places, hotels, and tour packages based on their preferences and budget.
2. **Customer Relationship Management (CRM)**: Agencies can track customer interactions, manage bookings, and provide personalized recommendations to enhance customer satisfaction and loyalty.
3. **Inventory Management**: Hotels and tour operators can manage room availability, pricing, and package inventory to optimize sales and revenue.
4. **Financial Management**: The system can generate financial reports, track transactions, and manage payments for bookings, ensuring transparency and accuracy in financial operations.
5. **Marketing and Promotions**: Agencies can leverage customer data to target marketing campaigns, offer discounts, and promotions to attract new customers and retain existing ones.

Overall, the TravelMate database project addresses the challenges of managing travel-related information, providing significant benefits to both users and travel agencies through streamlined operations, enhanced customer experiences, and data-driven decision-making capabilities.