

Exotic Travel Planner



Name:BelsabelWoldemichael
ID:20052
Instructor:Dr. Bhaskar
Vidhyacharan

Agenda

- Introduction
- Entities and Attributes
- Relationships
- Entity - Relationship Diagram
- Tables
- SQL Queries
- Conclusion

Introduction

Exotic Travel Planner enables users to :

- Tailor their trips based on their preferences, interests, and travel style.
- Easily create and customize detailed schedule to make the most of their travel adventures.
- Plan their days, including activities, accommodations, and transportation, all in one place.
- Set and track their budget for accommodations, activities, and more, ensuring you stay within your financial comfort zone.

Scenario

- **User Registration:** Bob visits the Exotic Travel Planner website, registers, and logs in.
- **Destination Search:** Bob decides on a relaxing vacation and selects Nordic island as his destination.
- **Destination Details:** Bob explores Nordic island page, finding details like: Descriptions of the islands and their unique features.

- **Itinerary Planning:** Excited about the destination, Bob plans his itinerary for the week.
- **Expense Tracking:** Bob wants to manage his budget, so he adds expenses to his itinerary.
- **User Reviews:** Before finalizing his plans, Bob reads user reviews to get insights from others who have visited Nordic island.
- **Finalizing Plans:** Bob adjusts his itinerary based on the reviews and feels confident about his plans.

Entities and Attributes

- **UserPreference:** PreferenceID(PK) , UserID, Category, PreferenceValue
- **User:** UserID(PK), FirstName, LastName, Email, ContactNumber, Gender, Address, RegistrationDate
- **Activity:** ActivityID(PK), DestinationID, Name, Description
- **Accommodation:** AccommodationID(PK), Name, Type, CostPerNight, Address, DestinationID

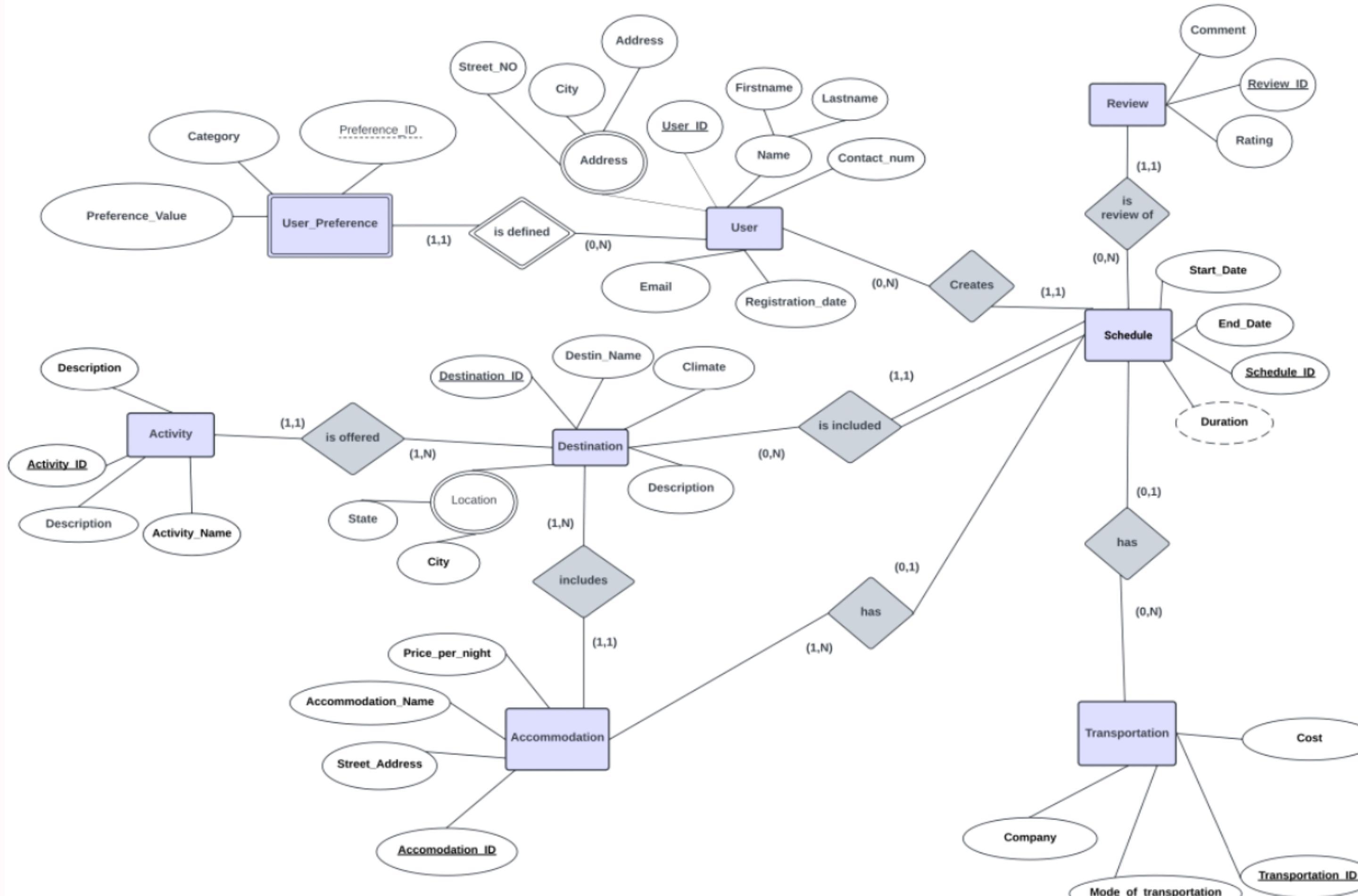
- **Review:** ReviewID(PK), ScheduleID, Rating, Comment
- **Transportation:** TransportID(PK), Mode, Company, DepartureLocation, Arrival Location, Cost
- **Schedule:** ScheduleID(PK), UserID, DestinationID, AccommodationID, TransportID, StartDate, EndDate, Duration, TotalCost
- **Destination:** DestinationID(PK), Name, Description, Location, Climate

Relationships

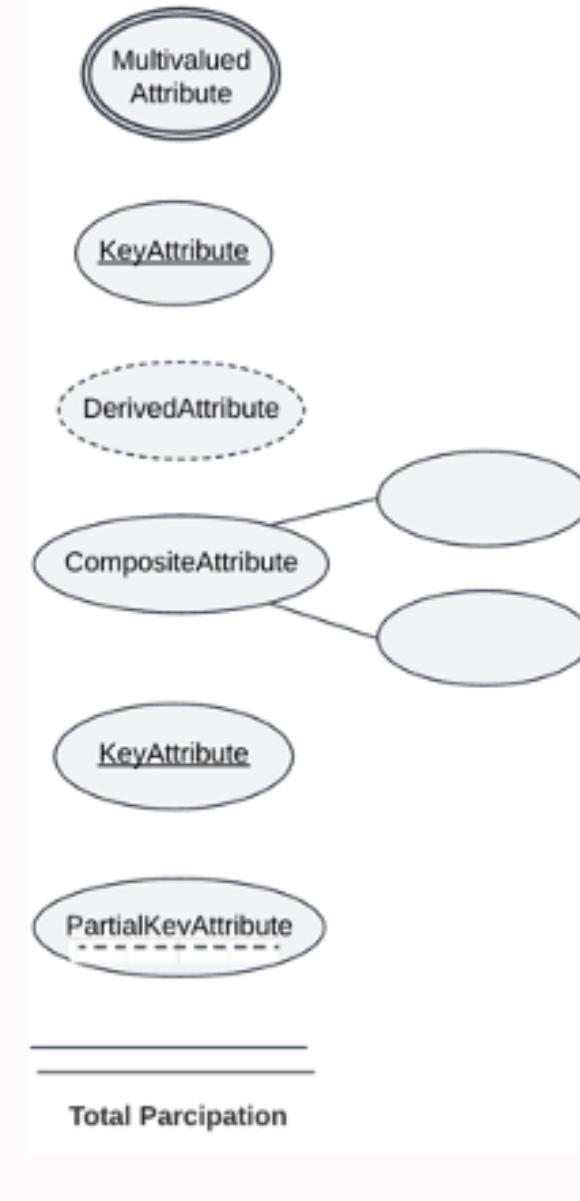
- **USER and USERPREFERENCE:** (1:M) One to Many
Connected through UserID
- **SCHEDULE and USER:** (M:1) Many-to-One
Connected through DestinationID
- **SCHEDULE and DESTINATION:** (1:M) One-to-Many
Connected through DestinationID
- **SCHEDULE and ACCOMMODATION:** (1:M) One-to-Many
Connected through AccommodationID

- **SCHEDULE** and **TRANSPORTATION**: (1:M) One to Many
Connected through TransportID.
- **ACTIVITY** and **DESTINATION**: (M:1) Many-to-One
Connected through DestinationID.
- **ACCOMMODATION** and **DESTINATION**: (M:1) Many-to-One
Connected through DestinationID.
- **REVIEW** and **SCHEDULE**: (M:1) Many-to-One
Connected through ScheduleID.

Entity-Relationship Diagram



LEGENDS



Tables

```
[MariaDB [20052wb]]> select * from Schedule;
```

ScheduleID	UserID	DestinationID	AccommodationID	TransportID	StartDate	EndDate	Duration	TotalCost
701	11	1	401	301	2023-12-15	2023-12-18	4	750.00
702	12	1	402	302	2023-12-20	2023-12-25	6	600.00
703	13	2	403	303	2024-01-05	2024-01-10	5	800.00
704	14	2	404	304	2024-02-01	2024-02-07	7	900.00
705	15	3	405	305	2024-03-15	2024-03-20	6	700.00
706	16	3	406	306	2024-04-10	2024-04-15	5	850.00
707	17	4	407	307	2024-05-05	2024-05-10	6	950.00
708	18	4	408	308	2024-06-01	2024-06-07	7	720.00
709	19	5	409	309	2024-07-15	2024-07-20	6	780.00
710	20	5	410	310	2024-08-10	2024-08-15	5	680.00

```
MariaDB [20052wb]> select * from ACCOMMODATION;
```

AccommodationID	Name	Type	CostPerNight	StreetAddress	DestinationID
401	Coastal Retreat	Resort	200.00	123 Coastal Blvd	1
402	Seaside Lodge	Lodge	120.00	456 Ocean View Dr	1
403	City View Hotel	Hotel	150.00	789 Downtown Ave	2
404	Riverside Inn	Bed and Breakfast	120.00	101 Riverwalk St	2
405	Beachside Villa	Vacation Rental	180.00	345 Beachfront Rd	3
406	Island Retreat	Resort	220.00	567 Island Paradise Ln	3
407	Luxury Resort	Resort	250.00	789 Luxury Blvd	4
408	Tropical Paradise Hotel	Hotel	170.00	432 Palm Tree Ave	4
409	Historic Inn	Bed and Breakfast	100.00	210 Historic Ln	5
410	Garden Retreat	Vacation Rental	130.00	543 Garden Path	5

```
10 rows in set (0.000 sec)
```

```
[MariaDB [20052wb]]> select * from UserPreference;
```

PreferenceID	UserID	Category	PreferenceValue
901	11	Destination	Beach
902	11	Accommodation Type	Resort
903	12	Destination	City
904	12	Accommodation Type	Hotel
905	13	Destination	Mountain
906	13	Accommodation Type	Lodge
907	14	Destination	Historical
908	14	Accommodation Type	Bed and Breakfast
909	15	Destination	City
910	15	Accommodation Type	Vacation Rental

```
[MariaDB [20052wb]]> Select * from Transportation;
```

TransportID	Mode	Company	DepartureLocation	ArrivalLocation	Cost
301	Flight	Delta Airlines	LAX, Los Angeles, CA	STS, Santa Rosa, CA	300.00
302	Train	Amtrak	Penn Station, NY, NY	Willits Station, Willits, CA	200.00
303	Bus	Greyhound	Houston Transit Center, TX	Albany Bus Terminal, NY	80.00
304	Flight	American Airlines	SFO, San Francisco, CA	Hudson Valley Airport, Hudson, NY	150.00
305	Flight	Delta Airlines	Navy Pier, Chicago, IL	GLS, Galveston, TX	250.00
306	Flight	American Airlines	MIA, Miami, FL	GLS, Galveston, TX	200.00
307	Bus	Greyhound	DFW, Dallas, TX	RSW, Fort Myers, FL	120.00
308	Train	Amtrak	ATL, Atlanta, GA	Naples Station, Naples, FL	180.00
309	Bus	Megabus	Denver Union Station, CO	SAV, Savannah, GA	160.00
310	Flight	Alaska Airlines	SEA, Seattle, WA	SAV, Savannah, GA	300.00

```
[MariaDB [20052wb]]> select * from DESTINATION;
```

DestinationID	Name	Description	Location	Climate
1	Headlands Park	Coastal charm	Mendocino, California	Mild climate
2	Olana Site	Hudson River city	Hudson, New York	Temperate climate
3	Strand District	Island beauty	Galveston, Texas	Subtropical climate
4	Sunshine Getaway	Luxury beach	Naples, Florida	Tropical climate
5	Southern Tour	Historic streets	Savannah, Georgia	Subtropical climate
6	Midwest Adventure	Art galleries and activities	Yellow Springs, Ohio	Continental climate
7	Red Rock State Park	Scenic state park	Sedona, Arizona	Mild climate
8	Greenfield Village	Outdoor museum	Dearborn, Michigan	Continental climate
9	Galena District	Charming shops	Galena, Illinois	Continental climate
10	Glass Beach	Colorful sea glass beach	Fort Bragg, California	Mild climate

```
[MariaDB [20052wb]]> Select * from ACTIVITY;
```

ActivityID	DestinationID	Name	Description
201	1	Coastal Hiking	Scenic Trail Explore
202	1	Photography Session	Capture Moments California
203	2	Historical Tour	Immerse Olana, NY
204	2	Scenic Drives	Breathtaking Views Hudson, NY
206	3	Beachfront Dining	Seafood Indulgence Strand
207	3	Waterfront Stroll	Leisurely Island Stroll
209	4	Water Sports	Exciting Naples Activities
210	4	Sunset Cruise	Romantic Naples Sunset
212	5	Historic Walking Tour	Explore Savannah Streets
213	5	Riverfront Entertainment	River Street Fun

```
10 rows in set (0.000 sec)
```

```
MariaDB [20052wb]> select * from DESTINATION;
```

DestinationID	Name	Description	Location	Climate
1	Headlands Park	Coastal charm	Mendocino, California	Mild climate
2	Olana Site	Hudson River city	Hudson, New York	Temperate climate
3	Strand District	Island beauty	Galveston, Texas	Subtropical climate
4	Sunshine Getaway	Luxury beach	Naples, Florida	Tropical climate
5	Southern Tour	Historic streets	Savannah, Georgia	Subtropical climate
6	Midwest Adventure	Art galleries and activities	Yellow Springs, Ohio	Continental climate
7	Red Rock State Park	Scenic state park	Sedona, Arizona	Mild climate
8	Greenfield Village	Outdoor museum	Dearborn, Michigan	Continental climate
9	Galena District	Charming shops	Galena, Illinois	Continental climate
10	Glass Beach	Colorful sea glass beach	Fort Bragg, California	Mild climate

```
MariaDB [20052wb]> select * from Review;
```

ReviewID	ScheduleID	Rating	Comment
801	701	5	Amazing!
802	702	4	Great accommodations, transportation can improve.
803	703	5	Excellent journey, well-planned.
804	704	3	Nice destination, better accommodation needed.
805	705	5	Fantastic trip! Enjoyed every moment.
806	706	4	Good overall, room for improvement in accommodation.
807	707	5	Breathtaking views at the destination!
808	708	4	Well-organized trip, slight transportation delay.
809	709	5	Memorable experience! Loved it all.
810	710	3	Average trip. Accommodation and activities were okay.

Basic Commands

1. Retrieve the details of destinations that have either a "Mild climate" or are located in California, excluding those with a "Tropical climate." Provide the DestinationID, Name, Description, Location, and Climate for each destination.

```
MariaDB [20052wb]> SELECT *
-> FROM DESTINATION
-> WHERE (Climate = 'Mild climate' OR Location = 'California') AND NOT Climate = 'Tropical climate';
+-----+-----+-----+-----+
| DestinationID | Name           | Description          | Location            | Climate           |
+-----+-----+-----+-----+
|      1 | Headlands Park    | Coastal charm        | Mendocino, California | Mild climate     |
|      7 | Red Rock State Park | Scenic state park    | Sedona, Arizona      | Mild climate     |
|     10 | Glass Beach        | Colorful sea glass beach | Fort Bragg, California | Mild climate     |
+-----+-----+-----+-----+
```

$$R2 = \sigma_{\text{Climate}='Mild climate' \text{ OR } \text{Location}='California'}(\text{DESTINATION})$$

$$R3 = \pi_{\text{DestinationID}, \text{Name}, \text{Description}, \text{Location}, \text{Climate}}(R2)$$

$$R4 = \sigma_{\text{NOT } (\text{Climate}='Tropical climate')} (R3)$$

2. Retrieve the user IDs, destination IDs, accommodation IDs, transport IDs, and durations of schedules for trips that have a duration between 5 and 7 days.

```
MariaDB [20052wb]> SELECT UserID, DestinationID, AccommodationID, TransportID, Duration  
-> FROM Schedule  
-> WHERE Duration BETWEEN 5 AND 7  
-> GROUP BY Duration;
```

UserID	DestinationID	AccommodationID	TransportID	Duration
13	2	403	303	5
12	1	402	302	6
14	2	404	304	7

$$\pi_{\text{UserID}, \text{DestinationID}, \text{AccommodationID}, \text{TransportID}, \text{Duration}}(\sigma_{\text{Duration} \in [5,7]}(\text{Schedule}))$$

3. Retrieve the Gender and User IDs of users who registered between December 1, 2023, and December 31, 2023.

```
MariaDB [20052wb]> SELECT Gender, UserID
-> FROM User
-> WHERE RegistrationDate >= '2023-12-01' AND RegistrationDate < '2024-01-01'
-> GROUP BY GENDER;
+-----+-----+
| Gender | UserID |
+-----+-----+
| F      |    12   |
| M      |    11   |
+-----+-----+
2 rows in set (0.000 sec)
```

$$R_1 \leftarrow \sigma_{\text{RegistrationDate} \geq '2023-12-01' \wedge \text{RegistrationDate} < '2024-01-01'}(\text{User})$$

$$R_2 \leftarrow \pi_{\text{Gender}, \text{UserID}}(R_1)$$

4. Retrieve the first names, user IDs, and email addresses of users whose email addresses contain 'gmail.com' and are not associated with the gender 'Female'.

```
MariaDB [20052wb]> SELECT FirstName, UserID, Email
    -> FROM User
    -> WHERE Email LIKE '%gmail.com%' AND Gender <> 'F';
+-----+-----+-----+
| FirstName | UserID | Email           |
+-----+-----+-----+
| John      |    11 | john@gmail.com |
| Bob       |    13 | bob@gmail.com  |
| Charlie   |    15 | charlie@gmail.com |
| David     |    17 | david@gmail.com |
| Frank     |    19 | frank@gmail.com |
| Ian       |    21 | ian@gmail.com  |
+-----+-----+-----+
6 rows in set (0.000 sec)
```

$$\pi_{FirstName, UserID, Email}(\sigma_{Email \text{ LIKE } \%gmail.com\% \text{ AND } Gender \neq 'F'}(User))$$

5. Retrieve the modes of transportation, their respective companies, total cost, and the number of trips for those modes and companies where the total cost of trips exceeds 500 from the transportation database.

```
MariaDB [20052wb]> SELECT Mode, Company,
-> SUM(Cost) AS TotalCost, COUNT(*) AS TripCount FROM Transportation
-> GROUP BY Mode, Company
-> HAVING TotalCost > 500;
+-----+-----+-----+
| Mode | Company | TotalCost | TripCount |
+-----+-----+-----+
| Flight | Delta Airlines | 550.00 | 2 |
+-----+-----+-----+
1 row in set (0.000 sec)
```

$R1 \leftarrow \rho_{\text{Mode, Company}, \text{SUM}(\text{Cost}) \text{ as } \text{TotalCost}, \text{COUNT}(\text{*}) \text{ as } \text{TripCount}}(R1)$

$R2 \leftarrow \sigma_{\text{Mode} = \text{Flight}} \rho_{\text{Company}, \text{SUM}(\text{Cost}) \text{ as } \text{TotalCost}, \text{COUNT}(\text{*}) \text{ as } \text{TripCount}}(R1)$

$R3 \leftarrow \pi_{\text{Mode, Company, TotalCost, TripCount}}(\sigma_{\text{TotalCost} > 500}(R2))$

Intermediate Commands

1. Retrieve activity information along with corresponding accommodation details for destinations that have both activities and accommodations. Perform a right join between the Activity and Accommodation tables based on the destination ID.

```
MariaDB [20052wb]> SELECT ACTIVITY.ActivityID, ACTIVITY.DestinationID, ACTIVITY.  
-> Name AS ActivityName, ACCOMMODATION.AccommodationID,  
-> ACCOMMODATION.Name AS AccommodationName,  
-> ACCOMMODATION.Type AS AccommodationType  
-> FROM ACTIVITY  
-> RIGHT JOIN ACCOMMODATION ON ACTIVITY.DestinationID = ACCOMMODATION.DestinationID;
```

ActivityID	DestinationID	ActivityName	AccommodationID	AccommodationName	AccommodationType
201	1	Coastal Hiking	401	Coastal Retreat	Resort
202	1	Photography Session	401	Coastal Retreat	Resort
201	1	Coastal Hiking	402	Seaside Lodge	Lodge
202	1	Photography Session	402	Seaside Lodge	Lodge
203	2	Historical Tour	403	City View Hotel	Hotel
204	2	Scenic Drives	403	City View Hotel	Hotel
203	2	Historical Tour	404	Riverside Inn	Bed and Breakfast
204	2	Scenic Drives	404	Riverside Inn	Bed and Breakfast
206	3	Beachfront Dining	405	Beachside Villa	Vacation Rental
207	3	Waterfront Stroll	405	Beachside Villa	Vacation Rental
206	3	Beachfront Dining	406	Island Retreat	Resort
207	3	Waterfront Stroll	406	Island Retreat	Resort
209	4	Water Sports	407	Luxury Resort	Resort
210	4	Sunset Cruise	407	Luxury Resort	Resort
209	4	Water Sports	408	Tropical Paradise Hotel	Hotel
210	4	Sunset Cruise	408	Tropical Paradise Hotel	Hotel
212	5	Historic Walking Tour	409	Historic Inn	Bed and Breakfast
213	5	Riverfront Entertainment	409	Historic Inn	Bed and Breakfast
212	5	Historic Walking Tour	410	Garden Retreat	Vacation Rental
213	5	Riverfront Entertainment	410	Garden Retreat	Vacation Rental

20 rows in set (0.000 sec)

2. Retrieve user information along with corresponding schedule details for users who have scheduled trips. Join the User and Schedule tables based on the user ID.

```
MariaDB [20052wb]> SELECT U.UserID, U.FirstName, U.LastName, S.ScheduleID, S.StartDate, S.EndDate
-> FROM User U
-> INNER JOIN Schedule S ON U.UserID = S.UserID;
+-----+-----+-----+-----+-----+
| UserID | FirstName | LastName | ScheduleID | StartDate | EndDate |
+-----+-----+-----+-----+-----+
| 11 | John | Doe | 701 | 2023-12-15 | 2023-12-18 |
| 12 | Jane | Smith | 702 | 2023-12-20 | 2023-12-25 |
| 13 | Bob | Johnson | 703 | 2024-01-05 | 2024-01-10 |
| 14 | Alice | Williams | 704 | 2024-02-01 | 2024-02-07 |
| 15 | Charlie | Brown | 705 | 2024-03-15 | 2024-03-20 |
| 16 | Eva | Martinez | 706 | 2024-04-10 | 2024-04-15 |
| 17 | David | Clark | 707 | 2024-05-05 | 2024-05-10 |
| 18 | Grace | Lee | 708 | 2024-06-01 | 2024-06-07 |
| 19 | Frank | Garcia | 709 | 2024-07-15 | 2024-07-20 |
| 20 | Holly | Lopez | 710 | 2024-08-10 | 2024-08-15 |
+-----+-----+-----+-----+
10 rows in set (0.000 sec)
```

$$R_1 \leftarrow \text{User} \bowtie_{\text{User.UserID} = \text{Schedule.UserID}} \text{Schedule}$$

$$R_2 \leftarrow \pi_{\text{UserID}, \text{FirstName}, \text{LastName}, \text{ScheduleID}, \text{StartDate}, \text{EndDate}}(R_1)$$

3. Retrieve user preference information along with corresponding user details for users who have provided preferences. Perform a left join between the UserPreference and User tables based on the user ID.

```
MariaDB [20052wb]> SELECT UP.PreferenceID, UP.UserID, UP.Category, UP.PreferenceValue,  
-> U.FirstName, U.LastName, U.Email, U.ContactNumber  
-> FROM UserPreference AS UP  
-> LEFT JOIN User AS U ON UP.UserID = U.UserID;
```

PreferenceID	UserID	Category	PreferenceValue	FirstName	LastName	Email	ContactNumber
901	11	Destination	Beach	John	Doe	john@gmail.com	+1 555-123-4567
902	11	Accommodation Type	Resort	John	Doe	john@gmail.com	+1 555-123-4567
903	12	Destination	City	Jane	Smith	jane@gmail.com	+1 555-234-5678
904	12	Accommodation Type	Hotel	Jane	Smith	jane@gmail.com	+1 555-234-5678
905	13	Destination	Mountain	Bob	Johnson	bob@gmail.com	+1 555-345-6789
906	13	Accommodation Type	Lodge	Bob	Johnson	bob@gmail.com	+1 555-345-6789
907	14	Destination	Historical	Alice	Williams	alice@gmail.com	+1 555-456-7890
908	14	Accommodation Type	Bed and Breakfast	Alice	Williams	alice@gmail.com	+1 555-456-7890
909	15	Destination	City	Charlie	Brown	charlie@gmail.com	+1 555-567-8901
910	15	Accommodation Type	Vacation Rental	Charlie	Brown	charlie@gmail.com	+1 555-567-8901

4. Retrieve destination and activity details by performing an equi-join between the DESTINATION and ACTIVITY tables on the common attribute DestinationID.

```
MariaDB [20052wb]> SELECT D.DestinationID, D.Name AS DestinationName, A.ActivityID, A.Name AS ActivityName
-> FROM DESTINATION D
-> JOIN ACTIVITY A ON D.DestinationID = A.DestinationID;
+-----+-----+-----+-----+
| DestinationID | DestinationName | ActivityID | ActivityName |
+-----+-----+-----+-----+
| 1 | Headlands Park | 201 | Coastal Hiking |
| 1 | Headlands Park | 202 | Photography Session |
| 2 | Olana Site | 203 | Historical Tour |
| 2 | Olana Site | 204 | Scenic Drives |
| 3 | Strand District | 206 | Beachfront Dining |
| 3 | Strand District | 207 | Waterfront Stroll |
| 4 | Sunshine Getaway | 209 | Water Sports |
| 4 | Sunshine Getaway | 210 | Sunset Cruise |
| 5 | Southern Tour | 212 | Historic Walking Tour |
| 5 | Southern Tour | 213 | Riverfront Entertainment |
+-----+-----+-----+-----+
10 rows in set (0.000 sec)
```

5. Retrieve schedule details (schedule ID, start date, end date, duration, total cost) along with destination details (destination name, location, climate) for scheduled trips. Perform a natural join between the Schedule and DESTINATION tables.

```
MariaDB [20052wb]> SELECT S.ScheduleID, S.StartDate, S.EndDate, S.Duration, S.TotalCost,  
-> D.Name AS DestinationName, D.Location,D.Climate  
-> FROM Schedule S  
-> NATURAL JOIN DESTINATION D;
```

ScheduleID	StartDate	EndDate	Duration	TotalCost	DestinationName	Location	Climate
701	2023-12-15	2023-12-18	4	750.00	Headlands Park	Mendocino, California	Mild climate
702	2023-12-20	2023-12-25	6	600.00	Headlands Park	Mendocino, California	Mild climate
703	2024-01-05	2024-01-10	5	800.00	Olana Site	Hudson, New York	Temperate climate
704	2024-02-01	2024-02-07	7	900.00	Olana Site	Hudson, New York	Temperate climate
705	2024-03-15	2024-03-20	6	700.00	Strand District	Galveston, Texas	Subtropical climate
706	2024-04-10	2024-04-15	5	850.00	Strand District	Galveston, Texas	Subtropical climate
707	2024-05-05	2024-05-10	6	950.00	Sunshine Getaway	Naples, Florida	Tropical climate
708	2024-06-01	2024-06-07	7	720.00	Sunshine Getaway	Naples, Florida	Tropical climate
709	2024-07-15	2024-07-20	6	780.00	Southern Tour	Savannah, Georgia	Subtropical climate
710	2024-08-10	2024-08-15	5	680.00	Southern Tour	Savannah, Georgia	Subtropical climate

Advanced Commands

1. Retrieve schedule details including user information, transportation details, destination details, and review information for schedules that meet certain criteria. Specifically, retrieve schedule ID, user ID, user name, destination name, transportation mode, transportation company, start date, end date, duration, total cost for schedules that start on or after '2023-12-15', have a total cost of at least 200, and either have no associated review or have a review with a rating greater than or equal to 2. The results should be ordered by start date in descending order.

```
MariaDB [20052wb]> SELECT S.ScheduleID, U.UserID, CONCAT(U.FirstName, ' ', U.LastName) AS UserName,
-> D.Name AS DestinationName, T.Mode AS TransportationMode, T.Company AS TransportationCompany, S.StartDate, S.EndDate, S.Duration, S.TotalCost
-> FROM Schedule AS S
-> INNER JOIN User AS U ON S.UserID = U.UserID
-> INNER JOIN Transportation AS T ON S.TransportID = T.TransportID
-> INNER JOIN DESTINATION AS D ON S.DestinationID = D.DestinationID
-> LEFT JOIN Review AS R ON R.ScheduleID = S.ScheduleID
-> WHERE
->     S.StartDate >= '2023-12-15'
->     AND S.TotalCost >= 200
->     AND (R.Rating IS NULL OR R.Rating >= 2)
-> ORDER BY
->     S.StartDate DESC;
```

ScheduleID	UserID	UserName	DestinationName	TransportationMode	TransportationCompany	StartDate	EndDate	Duration	TotalCost
710	20	Holly Lopez	Southern Tour	Flight	Alaska Airlines	2024-08-10	2024-08-15	5	680.00
709	19	Frank Garcia	Southern Tour	Bus	Megabus	2024-07-15	2024-07-20	6	780.00
708	18	Grace Lee	Sunshine Getaway	Train	Amtrak	2024-06-01	2024-06-07	7	720.00
707	17	David Clark	Sunshine Getaway	Bus	Greyhound	2024-05-05	2024-05-10	6	950.00
706	16	Eva Martinez	Strand District	Flight	American Airlines	2024-04-10	2024-04-15	5	850.00
705	15	Charlie Brown	Strand District	Flight	Delta Airlines	2024-03-15	2024-03-20	6	700.00
704	14	Alice Williams	Olana Site	Flight	American Airlines	2024-02-01	2024-02-07	7	900.00
703	13	Bob Johnson	Olana Site	Bus	Greyhound	2024-01-05	2024-01-10	5	800.00
702	12	Jane Smith	Headlands Park	Train	Amtrak	2023-12-20	2023-12-25	6	600.00
701	11	John Doe	Headlands Park	Flight	Delta Airlines	2023-12-15	2023-12-18	4	750.00

10 rows in set (0.001 sec)

2. Retrieve schedule details along with associated transportation details by performing a join between the Schedule, Transportation, and a subquery that links Schedule with Transportation on the common attribute TransportID

```
MariaDB [20052wb]> SELECT S.ScheduleID, S.UserID, S.DestinationID, S.AccommodationID,
->      T.TransportID, T.Mode AS TransportMode
-> FROM (SELECT sc.ScheduleID, sc.UserID, sc.DestinationID, sc.AccommodationID, tr.TransportID
->       FROM Schedule sc
->      JOIN Transportation tr ON sc.TransportID = tr.TransportID
-> ) AS S
-> JOIN Transportation T ON S.TransportID = T.TransportID;
```

ScheduleID	UserID	DestinationID	AccommodationID	TransportID	TransportMode
701	11	1	401	301	Flight
702	12	1	402	302	Train
703	13	2	403	303	Bus
704	14	2	404	304	Flight
705	15	3	405	305	Flight
706	16	3	406	306	Flight
707	17	4	407	307	Bus
708	18	4	408	308	Train
709	19	5	409	309	Bus
710	20	5	410	310	Flight

$$R_1 \leftarrow \text{Schedule} \bowtie_{\text{Transport.TransportID} = \text{Schedule.TransportID}} \text{Transport}$$
$$R_2 \leftarrow \pi_{\text{ScheduleID}, \text{UserID}, \text{DestinationID}, \text{AccommodationID}, \text{TransportID}}(R_1)$$
$$R_3 \leftarrow \text{Transportation} \bowtie_{\text{S.TransportID} = \text{T.TransportID}} R_2$$
$$R_4 \leftarrow$$
$$\pi_{\text{ScheduleID}, \text{UserID}, \text{DestinationID}, \text{AccommodationID}, \text{T.TransportID}, \text{Mode as TransportMode}}(R_3)$$

3. Retrieve user information, destination information, transportation details, schedule details, and review information for schedules that meet specific criteria. Specifically, retrieve user ID, first name, last name, destination name, transportation mode, start date, end date, duration, and total cost for schedules with a duration greater than 4, a total cost greater than 500, a review rating of 5, and no reviews with a rating less than 3. The results should be ordered by schedule ID.

```
ariaDB [20052wb]> SELECT U.UserID, U.FirstName, U.LastName,
->     D.Name AS DestinationName, T.Mode AS TransportationMode,
->     S.StartDate, S.EndDate, S.Duration, S.TotalCost
-> FROM Schedule AS S
-> LEFT JOIN User AS U ON S.UserID = U.UserID
-> JOIN DESTINATION AS D ON S.DestinationID = D.DestinationID
-> JOIN Transportation AS T ON S.TransportID = T.TransportID
-> WHERE S.Duration > 4 AND S.TotalCost > 500
->     AND EXISTS ( SELECT * FROM Review AS R
->                 WHERE R.ScheduleID = S.ScheduleID AND R.Rating = 5 )
->     AND NOT EXISTS ( SELECT * FROM Review AS R
->                     WHERE R.ScheduleID = S.ScheduleID AND R.Rating < 3 )
-> ORDER BY S.ScheduleID;
```

UserID	FirstName	LastName	DestinationName	TransportationMode	StartDate	EndDate	Duration	TotalCost
13	Bob	Johnson	Olana Site	Bus	2024-01-05	2024-01-10	5	800.00
15	Charlie	Brown	Strand District	Flight	2024-03-15	2024-03-20	6	700.00
17	David	Clark	Sunshine Getaway	Bus	2024-05-05	2024-05-10	6	950.00
19	Frank	Garcia	Southern Tour	Bus	2024-07-15	2024-07-20	6	780.00

rows in set (0.001 sec)

4. Retrieve user information, destination name, transportation mode, start date, end date, and duration for schedules with a total cost greater than the average total cost for schedules with the same destination. Additionally, ensure that there exists at least one review for the schedule with a rating of 4 or higher. Order the results by the start date in descending order.

```
MariaDB [20052wb]> SELECT U.UserID, D.Name AS DestinationName, T.Mode AS TransportationMode,
->     S.StartDate, S.EndDate, S.Duration
-> FROM Schedule AS S
-> JOIN User AS U ON S.UserID = U.UserID
-> JOIN Transportation AS T ON S.TransportID = T.TransportID
-> JOIN DESTINATION AS D ON S.DestinationID = D.DestinationID
-> WHERE S.TotalCost > (SELECT AVG(TotalCost) FROM Schedule WHERE DestinationID = S.DestinationID)
->     AND EXISTS (SELECT * FROM Review AS R WHERE R.ScheduleID = S.ScheduleID AND R.Rating >= 4)
-> ORDER BY S.StartDate DESC;
```

UserID	DestinationName	TransportationMode	StartDate	EndDate	Duration
19	Southern Tour	Bus	2024-07-15	2024-07-20	6
17	Sunshine Getaway	Bus	2024-05-05	2024-05-10	6
16	Strand District	Flight	2024-04-10	2024-04-15	5
11	Headlands Park	Flight	2023-12-15	2023-12-18	4

4 rows in set (0.001 sec)

5. Retrieve schedule information including schedule ID, user name, destination name, transportation mode, transportation company, start date, end date, duration, and total cost for schedules that have at least one review with a rating of 5 and no reviews with a rating less than 3.

```
MariaDB [20052wb]> SELECT S.ScheduleID, CONCAT(U.FirstName, ' ', U.LastName) AS UserName,  
-> D.Name AS DestinationName, T.Mode AS TransportationMode,  
-> T.Company AS TransportationCompany, S.StartDate, S.EndDate,  
-> S.Duration, S.TotalCost FROM Schedule AS S  
-> JOIN User AS U ON S.UserID = U.UserID  
-> JOIN Transportation AS T ON S.TransportID = T.TransportID  
-> JOIN DESTINATION AS D ON S.DestinationID = D.DestinationID  
-> WHERE EXISTS (SELECT * FROM Review AS R WHERE R.ScheduleID = S.ScheduleID AND R.Rating = 5)  
-> AND NOT EXISTS (SELECT * FROM Review AS R WHERE R.ScheduleID = S.ScheduleID AND R.Rating < 3);
```

ScheduleID	UserName	DestinationName	TransportationMode	TransportationCompany	StartDate	EndDate	Duration	TotalCost
701	John Doe	Headlands Park	Flight	Delta Airlines	2023-12-15	2023-12-18	4	750.00
703	Bob Johnson	Olana Site	Bus	Greyhound	2024-01-05	2024-01-10	5	800.00
705	Charlie Brown	Strand District	Flight	Delta Airlines	2024-03-15	2024-03-20	6	700.00
707	David Clark	Sunshine Getaway	Bus	Greyhound	2024-05-05	2024-05-10	6	950.00
709	Frank Garcia	Southern Tour	Bus	Megabus	2024-07-15	2024-07-20	6	780.00

5 rows in set (0.001 sec)

Conclusion

- The ER model provided a visual representation of the relationships among entities.
- Tables provided a structured and organized way to store data in a relational database, allowing for efficient data management and retrieval.
- Tables enabled the execution of complex queries and support a wide range of data retrieval operations, providing flexibility for diverse application requirements.

Reference

Relational database design based on the entity-relationship model. Data & Knowledge Engineering (2003, February 12).
<https://www.sciencedirect.com/science/article/pii/0169023X9190033T>

What is an entity relationship diagram (ERD)? . Lucidchart. (n.d.).
<https://www.lucidchart.com/pages/entity-relationship-diagrams#:~:text=Make%20an%20ERD-,What%20is%20an%20ER%20diagram%3F,each%20other%20within%20a%20system.>