CPSC 304 Project Cover Page

Milestone #: 2

Date: 20 July 2024

Group Number: 15

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Cooper Webb	64017890	y7m3s	cooperw473@gmail.com
Dennis Chiu	13229778	d9c2q	dennischiu14@gmail.com
Waris Bhatia	69385458	i2u2b	bhatiawaris@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Description:

Our project resides in the domain of tourism. Specifically, our project focuses on tourist spots present in Canada.

The database models all relevant information about a tourist attraction, such as available activities, booking information, photos, and location. This application is applicable for tourists that want to visit areas that best fit their preference criteria (achieved with activity category filtering), and add reviews and comments to help future tourists to make informed decisions.

Alterations:

Category has been removed as an entity and replaced as an attribute in the "Tourist Attractions" entity. The entity itself has been replaced with transportation. Which will list transportation vehicle ID's, name and price. These are flat rate services that tourists can use for many attractions. This is represented as a many-to-many relationship with the "Tourist Attractions" entity.

Date has been removed from booking as the Oracle date type for startTime and endTime will suffice ('YYYY-MM-DD HH24:MI').
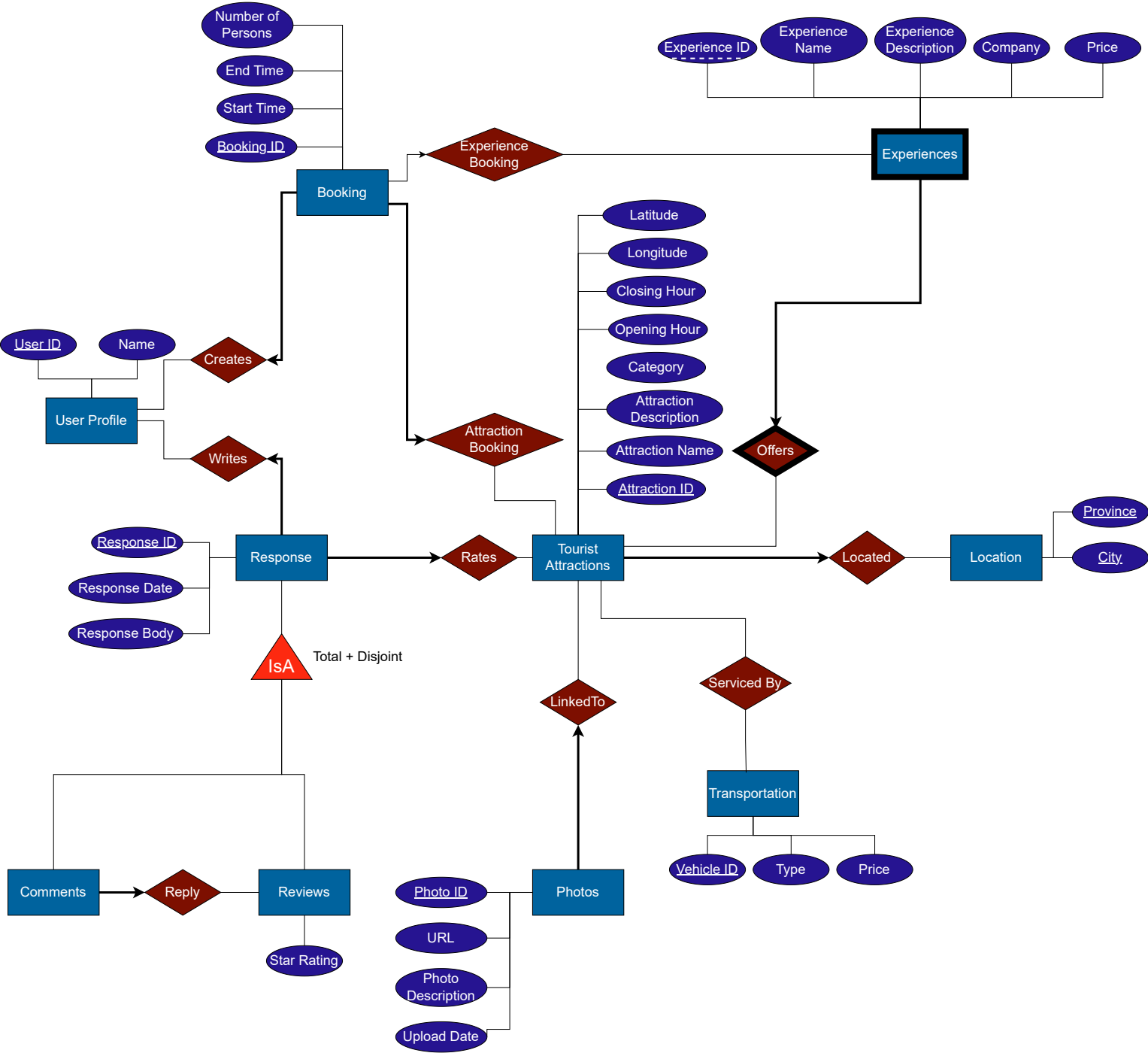
Location, User entities have been altered to Locations and UserProfile due to Oracle name reservations.

A few attributes have been altered with more specific names due to duplicate entries.

Altered attribute names:
- userName, experienceName, attractionName, experienceDesc, attractionDesc.

ER Diagram:

The ER Diagram contains the following entities and relationships:

Booking entity with attributes: Number of Persons, End Time, Start Time, Booking ID.

Experiences entity (strong/weak) with attributes: Experience ID, Experience Name, Experience Description, Company, Price.

Experience Booking relationship connects Booking and Experiences.

User Profile entity with attributes: User ID, Name.

Creates relationship connects User Profile and Booking.

Writes relationship connects User Profile and Response.

Tourist Attractions entity with attributes: Latitude, Longitude, Closing Hour, Opening Hour, Category, Attraction Description, Attraction Name, Attraction ID.

Attraction Booking relationship connects Booking and Tourist Attractions.

Offers relationship connects Experiences and Tourist Attractions.

Response entity with attributes: Response ID, Response Date, Response Body.

Rates relationship connects Response and Tourist Attractions.

IsA relationship (Total + Disjoint) from Response to Comments and Reviews.

Reply relationship connects Comments and Reviews.

Reviews entity with attribute: Star Rating.

Located relationship connects Tourist Attractions and Location.

Location entity with attributes: Province, City.

LinkedTo relationship connects Tourist Attractions and Photos.

Photos entity with attributes: Photo ID, URL, Photo Description, Upload Date.

Serviced By relationship connects Tourist Attractions and Transportation.

Transportation entity with attributes: Vehicle ID, Type, Price.

Database Schema

In the following schema, attributes that are <u>underlined</u> denote primary key attributes, and attributes that are **bolded** denote foreign key attributes.

- Locations(<u>province</u>**:** varchar, <u>city</u>: varchar)

- TouristAttractions(<u>attractionID</u>: number, **province**: varchar, **city:** varchar, attractionName: varchar, attractionDesc: varchar, category: varchar, openingHour: number, closingHour: number, latitude: number, longitude: number)

- Transportation(<u>vehicleID</u>: number, type: varchar, price: number)

- ServicedBy(**<u>vehicleID</u>**: number, **<u>attractionID</u>**: number)

- ExperienceOffered(<u>experienceID</u>: number, **attractionID**: number**,** experienceName: varchar, experienceDesc: varchar, company: varchar, price: number)

- UserProfile(**userID**: number**,** userName: varchar)

- Booking(<u>bookingID</u>: number, **experienceID**: number**, attractionID**: number**, userID**: number**,** startTime: date, endTime: date, numOfPersons: number)

- Responses(<u>responseID</u>: number, **userID**: number, **attractionID**: number, responseDate: date, body: varchar)

- Reviews(**<u>responseID</u>**: number, starRating: number)

- Comments(**<u>responseID</u>**: number, **reviewID**: number)

- Photos(<u>photoID</u>: number, **attractionID**: number, url: varchar, photoDesc: varchar, uploadDate: date)

Functional Dependencies (FD's)

Note: PK and CK refer to Primary Key and Candidate Key respectively.

UserProfile:
-   PK: userID → userName

Booking:
-   PK: bookingID → startTime, endTime, numOfPersons, experienceID, attractionID, userID
-   experienceID, userID, startTime → attractionID

Experience:
-   PK: experienceID, attractionID → experienceName, experienceDesc, company, price
-   CK: experienceName, latitude, longitude, experienceID → attractionID, experienceDesc, company, price

Responses:
-   responseID → responseDate, body, userID

Review:
-   reviewID → responseDate, body, userID, starRating

Comment:
-   commentID → responseDate, body, userID, reviewID

TouristAttractions:

-   PK: attractionID → attractionName, attractionDesc, category, openingHour, closingHour, latitude, longitude, province, city
-   CK: attractionName, latitude, longitude → attractionID, attractionDesc, category, openingHour, closingHour, province, city
-   long, lat → province, city

Locations: No non-trivial FD(s).

Transportation:
- vehicleID → type, price

Photos:
- PK: photoID → url, photoDesc, uploadDate, attractionID
- CK: url, uploadDate → photoID, photoDesc, attractionID

Normalization

All relations except **TouristAttractions and Booking** are already in BCNF. We will decompose both relations to satisfy the requirements of BCNF.

1. **Decompose TouristAttractions Relation:**

TouristAttractions(attractionID, **province**, **city,** attractionName, attractionDesc, category, openingHour, closingHour, latitude, longitude)

FD(s):
attractionID → attractionName, attractionDesc, category, openingHour, closingHour, latitude, longitude, province, city (BCNF)

attractionName, latitude, longitude → attractionID, attractionDesc, category, openingHour, closingHour, province, city (BCNF)

latitude, longitude-> province, city
(violates BCNF. latitude, longitude is not a superkey for TouristAttractions)

Decomposition:

TouristAttractions1(latitude, longitude, **province**, **city**)
(BCNF)
- (province, city) → Locations entity

TouristAttractions2(attractionID, attractionName, attractionDesc, category, openingHour, closingHour, **latitude, longitude**)
- (latitude, longitude,) → TouristAttractions1 entity
(BCNF)

2. Decompose Booking Relation:

FD(s):

bookingID → experienceID, attractionID, userID, startTime, endTime, numOfPersons
(BCNF)

experienceID, userID, startTime → attractionID
(violates BCNF. experienceID, userID, startTime are not a superkey for bookingID)

Decomposition:

Booking1(experienceID, **userID**, startTime, **attractionID**)
(BCNF)
- userID → UserProfile entity
- attractionID → TouristAttractionEntity

Booking2(bookingID, **experienceID**, **userID**, **startTime**, endTime, numOfPersons)
(BCNF)
- (experienceID, userID, startTime) → Booking1

We have arrived at the following list of tables, where attributes that are <u>underlined</u> denote primary key attributes, and attributes that are **bolded** denote foreign key attributes:

- Locations(<u>province</u>**:** varchar, <u>city</u>: varchar)
- TouristAttractions1(<u>latitude</u>: number, longitude: number **province**: varchar, **city:** varchar)
- TouristAttractions2(<u>attractionID</u>: number, attractionName: varchar, attractionDesc: varchar, category: varchar, openingHour: number, closingHour: number, **latitude**: number, **longitude**: number)
- Transportation(<u>vehicleID</u>: number, type: varchar, price: number)
- ServicedBy(**<u>vehicleID</u>**: number, **<u>attractionID</u>**: number)
- ExperienceOffered(<u>experienceID</u>: number, **attractionID**: number**,** experienceName: varchar, experienceDesc: varchar, company: varchar, price: number)
- UserProfile(**<u>userID</u>**: number**,** userName: varchar)
- Booking1(**<u>experienceID</u>**: number**,** **<u>userID</u>**: number**,** <u>startTime</u>: date, **attractionID**: number)
- Booking2(<u>bookingID</u>: number, **experienceID**: number**,** **userID**: number**,** **startTime**: date, **endTime**: date, numOfPersons: number)
- Responses(<u>responseID</u>: number, **userID**: number, **attractionID**: number, responseDate: date, body: varchar)
- Reviews(**<u>responseID</u>**: number, starRating: number)
- Comments(**<u>responseID</u>**: number, **reviewID**: number)
- Photos(<u>photoID</u>: number, **attractionID**: number, url: varchar, photoDesc: varchar, uploadDate: date)

SQL DDL Statements & Insertions

Below, the SQL statements required to create the tables and subsequently insert into them are listed.

```sql
CREATE TABLE Locations(
  province VARCHAR(2),
  city VARCHAR(20),
  PRIMARY KEY (province, city)
);


grant select on Locations to public;


CREATE TABLE UserProfile(
  userID number,
  userName varchar(20),
  PRIMARY KEY (userID)
);


grant select on UserProfile to public;


CREATE TABLE TouristAttractions1(
  latitude number(8, 5),
  longitude number(8, 5),
  province varchar(2),
  city varchar(20),
  PRIMARY KEY (latitude, longitude),
  FOREIGN KEY (province, city)
      REFERENCES Locations
      ON DELETE CASCADE
);


grant select on TouristAttractions1 to public;
```

```sql
CREATE TABLE TouristAttractions2(
  attractionID number,
  attractionName varchar(20),
  attractionDesc varchar(1000),
  category varchar(20),
  openingHour number(2, 0),
  closingHour number(2, 0),
  latitude number(8, 5),
  longitude number(8, 5),
  PRIMARY KEY (attractionID),
  FOREIGN KEY (latitude, longitude)
      REFERENCES TouristAttractions1
      ON DELETE CASCADE
);


grant select on TouristAttractions2 to public;


CREATE TABLE Transportation(
  vehicleID number,
  type varchar(20),
  price number(9, 2),
  PRIMARY KEY (vehicleID)
);


grant select on Transportation to public;


CREATE TABLE ServicedBy(
  vehicleID number,
  attractionID number,
  PRIMARY KEY (vehicleID, attractionID),
  FOREIGN KEY (vehicleID)
      REFERENCES Transportation
      ON DELETE CASCADE,
  FOREIGN KEY (attractionID)
      REFERENCES TouristAttractions2
      ON DELETE CASCADE
);


grant select on Transportation to public;
```

```sql
CREATE TABLE ExperienceOffered(
  experienceID number,
  attractionID number,
  experienceName varchar(20),
  experienceDesc varchar(1000),
  company varchar(20),
  price number(9, 2),
  PRIMARY KEY (experienceID),
  FOREIGN KEY (attractionID)
      REFERENCES TouristAttractions2
      ON DELETE CASCADE
);


grant select on ExperienceOffered to public;

CREATE TABLE Booking1(
  experienceID number,
  userID number,
  startTime date,
  attractionID number,
  PRIMARY KEY (experienceID, userID, startTime),
  FOREIGN KEY (attractionID)
      REFERENCES TouristAttractions2
      ON DELETE CASCADE,
  FOREIGN KEY (userID)
      REFERENCES UserProfile
      ON DELETE CASCADE
);


grant select on Booking1 to public;


CREATE TABLE Booking2(
  bookingID number,
  experienceID number,
  userID number,
  startTime date,
  endTime date,
  numOfPersons number,
  PRIMARY KEY (bookingID),
  FOREIGN KEY (experienceID, userID, startTime)
      REFERENCES Booking1
      ON DELETE CASCADE
);


grant select on Booking2 to public;
```

```sql
CREATE TABLE Responses (
  responseID int,
  userID int,
  attractionID int,
  responseDate DATE,
  body varchar(1000),
  PRIMARY KEY (responseID),
  FOREIGN KEY (userID)
      REFERENCES UserProfile
      ON DELETE CASCADE,
  FOREIGN KEY (attractionID)
      REFERENCES TouristAttractions2
      ON DELETE CASCADE
);

grant select on Responses to public;

CREATE TABLE Reviews (
  reviewID number,
  starRating number(1),
  PRIMARY KEY (reviewID),
  FOREIGN KEY (reviewID)
      REFERENCES Responses (responseID)
      ON DELETE CASCADE
);


grant select on Reviews to public;


CREATE TABLE Comments (
  commentID number,
  reviewID number,
  PRIMARY KEY (commentID),
  FOREIGN KEY (commentID)
      REFERENCES Responses (responseID)
      ON DELETE CASCADE,
  FOREIGN KEY (reviewID)
      REFERENCES reviews (reviewID)
      ON DELETE CASCADE
);


grant select on Comments to public;
```

```sql
CREATE TABLE Photos (
    photoID number,
    attractionID number,
    url varchar(2048),
    photoDescription varchar(1000),
    uploadDate date,
    PRIMARY KEY (photoID),
    FOREIGN KEY (attractionID)
        REFERENCES TouristAttractions2
        ON DELETE CASCADE
);

grant select on Photos to public;


insert into Locations
values('BC', 'Vancouver');
insert into Locations
values('BC', 'Victoria');

insert into Locations
values('AB', 'Calgary');
insert into Locations
values('ON', 'Ottawa');
insert into Locations
values('ON', 'Toronto');


insert into UserProfile
values(0, 'test0');
insert into UserProfile
values(1, 'test1');
insert into UserProfile
values(2, 'test2');
insert into UserProfile
values(3, 'test3');
insert into UserProfile
values(4, 'test4');


insert into TouristAttractions1
values(49.30427, -123.14421, 'BC', 'Vancouver');
insert into TouristAttractions1
values(48.42842, -123.36564, 'BC', 'Victoria');
insert into TouristAttractions1
values(51.04427, -114.06310, 'AB', 'Calgary');
insert into TouristAttractions1
values(45.42153, -75.69719, 'ON', 'Ottawa');
insert into TouristAttractions1
values(43.65107, -79.34702, 'ON', 'Toronto');
```

```sql
insert into TouristAttractions2
values(0, 'Stanley Park',
  q'[North America's third-largest park draws eight million visitors per year, many of whom
may skate or walk past you on the Seawall, a scenic, 5.5-mile path running along the water
on the park's perimeter. It's just one of many trails among the park's 1,000 acres, which
also house an aquarium, nature center and other recreational facilities.]',
  'Parks', null, null, 49.30427, -123.14421);

insert into TouristAttractions2
values(1, 'Butchart Gardens',
  q'[A group of floral display gardens in Brentwood Bay, British Columbia, Canada, near
Victoria on Vancouver Island. The gardens receive over a million visitors each year.]',
  'Parks', 9, 17, 48.42842, -123.36564);


insert into TouristAttractions2
values(2, 'Calgary Tower',
  q'[A 190.8-meter free standing observation tower in downtown Calgary, Alberta, Canada.]',
  'Landmarks', 08, 20, 51.04427, -114.06310);




insert into TouristAttractions2
values(3, 'Parliament Hill',
  q'[An area of Crown land on the southern banks of the Ottawa River in downtown Ottawa,
Ontario. Its Gothic revival suite of buildings is the home of the Parliament of Canada.]',
  'Landmarks', 10, 18, 45.421530, -75.697191);


insert into TouristAttractions2
values(4, 'CN Tower',
  q'[A 553.3 m-high concrete communications and observation tower located in Downtown
Toronto, Ontario.]',
  'Landmarks', null, null, 43.65107, -79.34702);

insert into Transportation
values(0, 'Translink', 3.00);

insert into Transportation
values(1, 'Mobi Bicycles', 10.00);

insert into Transportation
values(2, 'Yellow Cab Ottawa', 30.00);

insert into Transportation
values(3, 'Enterprise Calgary', 100.00);

insert into Transportation
values(4, 'Toronto Subway', 4.00);
```

```sql
insert into ServicedBy
values(0, 0);

insert into ServicedBy
values(0, 1);

insert into ServicedBy
values(3, 2);

insert into ServicedBy
values(3, 3);

insert into ServicedBy
values(4, 4);


insert into ExperienceOffered
values(0, 0, 'Seawall Walk', 'A scenic walk along the seawall in Stanley Park.',
'LocalTours', 30.00);

insert into ExperienceOffered
values (1, 1, 'Garden Tour', 'A guided tour of the Butchart Gardens.', 'Gardens', 20.00 );



insert into ExperienceOffered
values (2, 2, 'Tower Observation', 'An observation experience at the top of Calgary Tower.',
'Landmarks', 25.00 );

insert into ExperienceOffered
values (3, 3, 'Historical Tour', 'A guided historical tour of Parliament Hill.', 'History',
15.00 );

insert into ExperienceOffered
values (4, 4, 'Skyline View', 'An observation experience at the top of the CN Tower.',
'Landmarks', 35.00 );


INSERT INTO Booking1 VALUES (1, 0, TO_DATE('2024-07-20 12:30', 'YYYY-MM-DD HH24:MI'), 0);

INSERT INTO Booking1 VALUES (2, 1, TO_DATE('2024-07-21 13:30', 'YYYY-MM-DD HH24:MI'), 1);

INSERT INTO Booking1 VALUES (3, 2, TO_DATE('2024-07-22 15:40', 'YYYY-MM-DD HH24:MI'), 2);

INSERT INTO Booking1 VALUES (4, 3, TO_DATE('2024-07-23 17:20', 'YYYY-MM-DD HH24:MI'), 3);

INSERT INTO Booking1 VALUES (5, 4, TO_DATE('2024-07-24 8:30', 'YYYY-MM-DD HH24:MI'), 4);
```

```sql
INSERT INTO Booking2
VALUES (1, 1, 0, TO_DATE('2024-07-20 12:30', 'YYYY-MM-DD HH24:MI'), TO_DATE('2024-07-20
13:30', 'YYYY-MM-DD HH24:MI'), 2);

INSERT INTO Booking2
VALUES (2, 2, 1, TO_DATE('2024-07-21 13:30', 'YYYY-MM-DD HH24:MI'), TO_DATE('2024-07-21
15:40', 'YYYY-MM-DD HH24:MI'), 3);

INSERT INTO Booking2
VALUES (3, 3, 2, TO_DATE('2024-07-22 15:40', 'YYYY-MM-DD HH24:MI'), TO_DATE('2024-07-22
17:20', 'YYYY-MM-DD HH24:MI'), 4);

INSERT INTO Booking2
VALUES (4, 4, 3, TO_DATE('2024-07-23 17:20', 'YYYY-MM-DD HH24:MI'), TO_DATE('2024-07-23
18:00', 'YYYY-MM-DD HH24:MI'), 2);

INSERT INTO Booking2
VALUES (5, 5, 4, TO_DATE('2024-07-24 8:30', 'YYYY-MM-DD HH24:MI'), TO_DATE('2024-07-24
9:30', 'YYYY-MM-DD HH24:MI'), 1);


INSERT INTO Responses VALUES (1, 0, 0, TO_DATE('2024-07-20', 'YYYY-MM-DD'), 'Amazing
experience at Stanley Park!');

INSERT INTO Responses VALUES (2, 1, 1, TO_DATE('2024-07-21', 'YYYY-MM-DD'), 'The Butchart
Gardens were beautiful.');

INSERT INTO Responses VALUES (3, 2, 2, TO_DATE('2024-07-22', 'YYYY-MM-DD'), 'Great view from
the Calgary Tower.');

INSERT INTO Responses VALUES (4, 3, 3, TO_DATE('2024-07-23', 'YYYY-MM-DD'), 'Very
informative tour at Parliament Hill.');

INSERT INTO Responses VALUES (5, 4, 4, TO_DATE('2024-07-24', 'YYYY-MM-DD'), 'Stunning
skyline view from the CN Tower.');


INSERT INTO Reviews VALUES (1, 5);

INSERT INTO Reviews VALUES (2, 4);

INSERT INTO Reviews VALUES (3, 5);

INSERT INTO Reviews VALUES (4, 4);

INSERT INTO Reviews VALUES (5, 5);
```

```sql
INSERT INTO Comments VALUES (1, 1);

INSERT INTO Comments VALUES (2, 2);

INSERT INTO Comments VALUES (3, 3);

INSERT INTO Comments VALUES (4, 4);

INSERT INTO Comments VALUES (5, 5);


INSERT INTO Photos VALUES (1, 0, 'https://example.com/stanley_park.jpg', 'A beautiful view
of Stanley Park.', TO_DATE('2024-07-20', 'YYYY-MM-DD'));

INSERT INTO Photos VALUES (2, 1, 'https://example.com/butchart_gardens.jpg', 'Stunning
flowers at Butchart Gardens.', TO_DATE('2024-07-21', 'YYYY-MM-DD'));

INSERT INTO Photos VALUES (3, 2, 'https://example.com/calgary_tower.jpg', 'A view from
Calgary Tower.', TO_DATE('2024-07-22', 'YYYY-MM-DD'));

INSERT INTO Photos VALUES (4, 3, 'https://example.com/parliament_hill.jpg', 'Parliament Hill
in Ottawa.', TO_DATE('2024-07-23', 'YYYY-MM-DD'));

INSERT INTO Photos VALUES (5, 4, 'https://example.com/cn_tower.jpg', 'The CN Tower in
Toronto.', TO_DATE('2024-07-24', 'YYYY-MM-DD'));
```