University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: ____2____

Date: October 15th, 2024

Group Number: 18

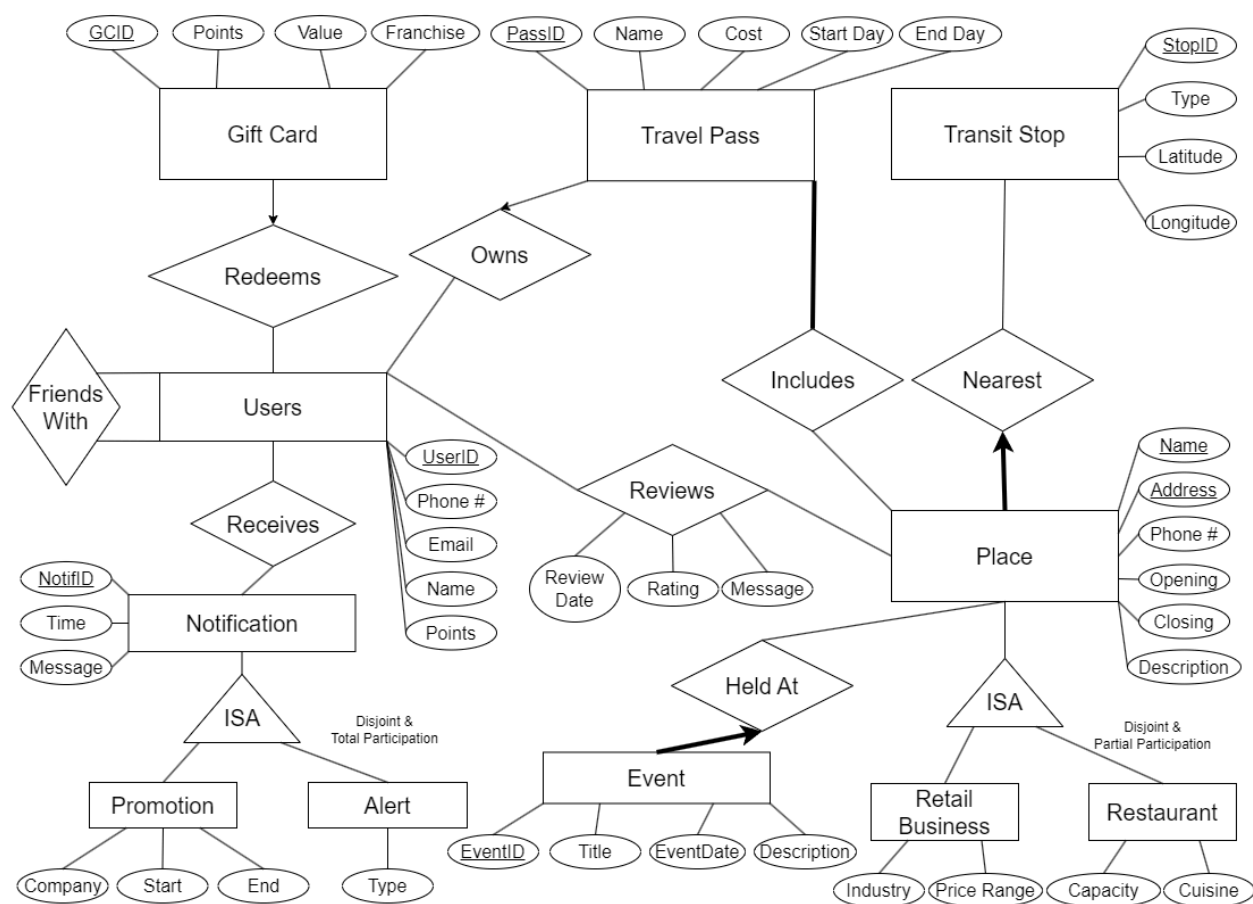| Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Matthew Fung	99002214	d0l9u	matthewfung9001@gmail.com
Parsa Seyed Zehtab	84226935	p7m3b	parsaz@shaw.ca
Ethan Wong	14532469	y7n9r	ethanwongca@gmail.com

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 application focuses on community engagement, local exploration, and tourism. This project will help users discover Vancouver's hidden gems of lesser-known foods, events, and recreation activities encouraging users to explore the whole extent of Vancouver. By providing references like costs, reviews, and transit options, this application will be a city exploration tool that highlights community experiences.

3. **ER DIAGRAM**

Changes to the ER Diagram:
The ISA constraints have been added as they were previously missing. Notifications total participation, while Places are partial as there can be many more different types of places not listed in our structure, but both ISAs are disjoint.

As recommended, we have updated Place to use address and name as the key. We previously chose not to use the address for the key as we thought some businesses may operate out of the same location, so we decided to include Name which seems to be a reasonable decision to us. Further, we have also updated Transit Stop's key to be a stopID, as suggested. We agree with this because the coordinates may not clearly define a stop (for example, a bus loop by a skytrain station).

We also added a Points attribute to GiftCard to create an extra non-trivial functional dependency. This is the amount of points a user needs to redeem a gift card. Same reason for the added Name attribute on User.

Lastly, we updated User to be Users because User is a keyword in Oracle. Same for Date in Reviews and Event to ReviewDate and EventDate.

4. **Initial Schema**

Users(UserID: INT, Name: VARCHAR(100), Phone: VARCHAR(15), Email: VARCHAR(100), Points: INT)
PK: UserID
Constraints: Email (Unique, Not Null)

FriendsWith(UserID1: INT, UserID2: INT)
PK: UserID1, UserID2
FKs: UserID1 REFERENCES Users(UserID), UserID2 REFERENCES Users(UserID)
Constraints: UserID1 <> UserID2

GiftCard(GCID: INT, Value: DECIMAL(10, 2), Points: INT Franchise: VARCHAR(100), UserID: INT)
PK: GCID
FK: UserID REFERENCES Users(UserID)
Constraints: Value >= 0

TravelPass(PassID: INT, Name: VARCHAR(100), Cost: DECIMAL(10, 2), StartDate: DATE, EndDate: DATE, UserID: INT)
PK: PassID
FK: UserID REFERENCES Users(UserID)
Constraints: (EndDate > StartDate), CHECK: Cost >= 0,

Includes(PassID: INT, Name: VARCHAR(100), Address: VARCHAR(200))
PK: (PassID, Name, Address)
FKs: PassID REFERENCES TravelPass(PassID), (Name, Address) REFERENCES Place(Name, Address)

TransitStop(StopID: INT, Type: VARCHAR(50), Latitude: DECIMAL(9, 6), Longitude: DECIMAL(9, 6))
PK: StopID
Constraints: Latitude between -90,90 & Longitude between -180,180

Place(Name: VARCHAR(100), Address: VARCHAR(200), Phone: VARCHAR(15), OpeningTime: VARCHAR2(8), ClosingTime: VARCHAR2(8), Description: VARCHAR(255), StopID: INT)
PK: (Name, Address)
FK: StopID REFERENCES TransitStop(StopID)
Constraint: (ClosingTime > OpeningTime)
Constraint: StopID (Not null)

RetailBusiness(Name: VARCHAR(100), Address: VARCHAR(200), Industry: VARCHAR(100), PriceRange: VARCHAR(10))
PK: (Name, Address)
FK: (Name, Address) REFERENCES Place(Name, Address)

Restaurant(Name: VARCHAR(100), Address: VARCHAR(200), Cuisine: VARCHAR(100), Capacity: INT)
PK: (Name, Address)
FK: Name, Address REFERENCES Place(Name, Address)
Constraint: Capacity > 0

Reviews(UserID: INT, Name: VARCHAR(100), Address: VARCHAR(200), ReviewDate: DATE, Rating: INT, Message: VARCHAR(255))
PK: (UserID, Name, Address)
FKs: UserID REFERENCES Users(UserID), (Name,Address) REFERENCES Place(Name, Address)
CHECK: Rating between 1,5

Event(EventID: INT, Name: VARCHAR(100), Address: VARCHAR(200), Title: VARCHAR(100), EventDate: DATE, Rating: INT, Description: VARCHAR(255))
PK: EventID
FK: (Name, Address) REFERENCES Place(Name, Address)
Constraint: Rating between 1,5
Constraint: Name, Address (Not null)

Notification(NotifID: INT, Time: TIMESTAMP, Message: VARCHAR(255))
PK: NotifID
Constraint: Time (Not null)

Promotion(NotifID: INT, Company: VARCHAR(100), StartDate: DATE, EndDate: DATE)
PK: NotifID

FK: NotifID REFERENCES Notification(NotifID)
Constraint: (EndDate > StartDate)
Alert(NotifID: INT, Type: VARCHAR(100))
PK: NotifID
FK: NotifID REFERENCES Notification(NotifID)

Receives(UserID: INT, NotifID: INT)
PK: (UserID, NotifID),
FKs: UserID REFERENCES Users(UserID), NotifID REFERENCES Notification(NotifID)

Note: There were no possible candidate keys (which is why most of the entities have [name]ID attributes).

5. **Functional Dependencies**

Users
  ● UserID → Name, Phone, Email, Points (PK)
  ● Email → Name
FriendsWith
  ● (UserID1, UserID2) → (UserID1, UserID2) (PK)
GiftCard
  ● GCID → Value, Franchise, UserID (PK), Points
  ● Value → Points
TravelPass
  ● PassID → Name, Cost, StartDate, EndDate, UserID (PK)
Includes
  ● (PassID, Name, Address) → (PassID, Name, Address) (PK)
TransitStop
  ● StopID → Type, Latitude, Longitude (PK)
Place
  ● Name, Address → Phone, OpeningTime, ClosingTime, Description, StopID (PK)
RetailBusiness
  ● Name, Address → Industry, PriceRange (PK)
Restaurant
  ● Name, Address → Cuisine, Capacity (PK)
Reviews
  ● (UserID, Name, Address) → ReviewDate, Rating, Message (PK)
Event
  ● EventID → Name, Address, Title, EventDate, Rating, Description (PK)
Notification
  ● NotifID → Time, Message (PK)
Promotion
  ● NotifID → Company, StartDate, EndDate (PK)

Alert
  ● NotifID → Type (PK)
Receives
  ● (UserID, NotifID) → (UserID, NotifID) (PK)

6. **Normalization**
Given that only User and Gift Card are the only ones with dependencies unrelated to the primary key, those are the only ones we will need to check for normalization. We will convert to BCNF.
For Users, Email → name violates BCNF, since email is not a superkey.
Decomposing, we get:
Users(UserID: INT, Phone: VARCHAR(15), Email: VARCHAR(100), Points: INT)
PK: UserID
Constraints: Email (Unique, Not Null)
 and
EmailName(Email: VARCHAR(100), Name: VARCHAR(100))
PK: Email

Similarly for GiftCard, Value→Points violates BCNF. Decomposing:
GiftCard(GCID: INT, Value: DECIMAL(10, 2), Franchise: VARCHAR(100), UserID: INT)
PK: GCID FK: UserID REFERENCES Users(UserID)
Constraints: Value >=0
 and
GCPoints(Value: DECIMAL(10, 2), Points: INT)
PK: Value

The rest are as written in part 4:

FriendsWith(UserID1: INT, UserID2: INT)
PK: UserID1, UserID2
FKs: UserID1 REFERENCES Users(UserID), UserID2 REFERENCES Users(UserID)
CHECK: (UserID1 <> UserID2)

TravelPass(PassID: INT, Name: VARCHAR(100), Cost: DECIMAL(10, 2), StartDate: DATE, EndDate: DATE, UserID: INT)
PK: PassID
FK: UserID REFERENCES Users(UserID)
Constraints: (EndDate > StartDate), CHECK: Cost >= 0,

Includes(PassID: INT, Name: VARCHAR(100), Address: VARCHAR(200))
PK: (PassID, Name, Address)

FKs: PassID REFERENCES TravelPass(PassID), (Name, Address) REFERENCES Place(Name, Address)

TransitStop(StopID: INT, Type: VARCHAR(50), Latitude: DECIMAL(9, 6), Longitude: DECIMAL(9, 6))
PK: StopID
Constraints: Latitude between -90,90 & Longitude between -180,180

Place(Name: VARCHAR(100), Address: VARCHAR(200), Phone: VARCHAR(15), OpeningTime: VARCHAR2(8), ClosingTime: VARCHAR2(8), Description: VARCHAR(255), StopID: INT)
PK: (Name, Address)
FK: StopID REFERENCES TransitStop(StopID)
Constraint: (ClosingTime > OpeningTime)

RetailBusiness(Name: VARCHAR(100), Address: VARCHAR(200), Industry: VARCHAR(100), PriceRange: VARCHAR(10))
PK: (Name, Address)
FK: (Name, Address) REFERENCES Place(Name, Address)

Restaurant(Name: VARCHAR(100), Address: VARCHAR(200), Cuisine: VARCHAR(100), Capacity: INT)
PK: (Name, Address)
FK: Name, Address REFERENCES Place(Name, Address)
Constraint: Capacity > 0

Reviews(UserID: INT, Name: VARCHAR(100), Address: VARCHAR(200), ReviewDate: DATE, Rating: INT, Message: VARCHAR(255))
PK: (UserID, Name, Address)
FKs: UserID REFERENCES Users(UserID), (Name,Address) REFERENCES Place(Name, Address)
CHECK: Rating between 1,5

Event(EventID: INT, Name: VARCHAR(100), Address: VARCHAR(200), Title: VARCHAR(100), EventDate: DATE, Rating: INT, Description: VARCHAR(255))
PK: EventID
FK: (Name, Address) REFERENCES Place(Name, Address)
Constraint: Rating between 1,5

Notification(NotifID: INT, Time: TIMESTAMP, Message: VARCHAR(255))
PK: NotifID
Constraint: Time (Not null)

Promotion(NotifID: INT, Company: VARCHAR(100), StartDate: DATE, EndDate: DATE)
PK: NotifID
FK: NotifID REFERENCES Notification(NotifID)
Constraint: (EndDate > StartDate)

Alert(NotifID: INT, Type: VARCHAR(100))
PK: NotifID
FK: NotifID REFERENCES Notification(NotifID)

Receives(UserID: INT, NotifID: INT)
PK: (UserID, NotifID),
FKs: UserID REFERENCES Users(UserID), NotifID REFERENCES Notification(NotifID)

7. **SQL DDL Statements**

```
CREATE TABLE TransitStop (
        StopID INT PRIMARY KEY,
        Type VARCHAR(50),
        Latitude DECIMAL(9, 6),
        Longitude DECIMAL(9, 6),
        CHECK (Latitude BETWEEN -90 AND 90),
        CHECK (Longitude BETWEEN -180 AND 180)
);

CREATE TABLE Place (
        Name VARCHAR(100),
        Address VARCHAR(200),
        Phone VARCHAR(15),
        OpeningTime VARCHAR2(8),
        ClosingTime VARCHAR2(8),
        Description VARCHAR(255),
        StopID INT,
        PRIMARY KEY (Name, Address),
        FOREIGN KEY (StopID) REFERENCES TransitStop(StopID) ON DELETE
CASCADE,
        CHECK (ClosingTime > OpeningTime)
);

CREATE TABLE Users (
        UserID INT PRIMARY KEY,
        Phone VARCHAR(15) UNIQUE,
        Email VARCHAR(100) UNIQUE NOT NULL,
        Points INT
```

```
);

CREATE TABLE EmailName (
      Email VARCHAR(100) PRIMARY KEY,
      Name VARCHAR(100)
);

CREATE TABLE GiftCard (
      GCID INT PRIMARY KEY,
      Value DECIMAL(10, 2) CHECK (Value >= 0),
      Franchise VARCHAR(100),
      UserID INT,
      FOREIGN KEY (UserID) REFERENCES Users(UserID)
);

CREATE TABLE GCPoints (
      Value DECIMAL(10, 2) PRIMARY KEY,
      Points INT
);

CREATE TABLE FriendsWith (
      UserID1 INT,
      UserID2 INT,
      PRIMARY KEY (UserID1, UserID2),
      FOREIGN KEY (UserID1) REFERENCES Users(UserID) ON DELETE
CASCADE,
      FOREIGN KEY (UserID2) REFERENCES Users(UserID) ON DELETE
CASCADE,
      CHECK (UserID1 <> UserID2)
);

CREATE TABLE TravelPass (
      PassID INT PRIMARY KEY,
      Name VARCHAR(100),
      Cost DECIMAL(10, 2) CHECK (Cost >= 0),
      StartDate DATE,
      EndDate DATE,
      UserID INT,
      FOREIGN KEY (UserID) REFERENCES Users(UserID),
      CHECK (EndDate > StartDate)
);

CREATE TABLE Includes (
      PassID INT,
```

```
        Name VARCHAR(100),
        Address VARCHAR(200),
        PRIMARY KEY (PassID, Name, Address),
        FOREIGN KEY (PassID) REFERENCES TravelPass(PassID) ON DELETE
CASCADE,
        FOREIGN KEY (Name, Address) REFERENCES Place(Name, Address) ON
DELETE CASCADE
);

CREATE TABLE RetailBusiness (
        Name VARCHAR(100),
        Address VARCHAR(200),
        Industry VARCHAR(100),
        PriceRange VARCHAR(10),
        PRIMARY KEY (Name, Address),
        FOREIGN KEY (Name, Address) REFERENCES Place(Name, Address) ON
DELETE CASCADE
);

CREATE TABLE Restaurant (
        Name VARCHAR(100),
        Address VARCHAR(200),
        Cuisine VARCHAR(100),
        Capacity INT CHECK (Capacity > 0),
        PRIMARY KEY (Name, Address),
        FOREIGN KEY (Name, Address) REFERENCES Place(Name, Address) ON
DELETE CASCADE
);

CREATE TABLE Reviews (
        UserID INT,
        Name VARCHAR(100),
        Address VARCHAR(200),
        ReviewDate DATE,
        Rating INT CHECK (Rating BETWEEN 1 AND 5),
        Message VARCHAR(255),
        PRIMARY KEY (UserID, Name, Address),
        FOREIGN KEY (UserID) REFERENCES Users(UserID),
        FOREIGN KEY (Name, Address) REFERENCES Place(Name, Address)
);


CREATE TABLE Event (
        EventID INT PRIMARY KEY,
```

```
        Name VARCHAR(100),
        Address VARCHAR(200),
        Title VARCHAR(100),
        EventDate DATE,
        Rating INT CHECK (Rating BETWEEN 1 AND 5),
        Description VARCHAR(255),
        FOREIGN KEY (Name, Address) REFERENCES Place(Name, Address) ON
DELETE CASCADE
);

CREATE TABLE Notification (
    NotifID INT PRIMARY KEY,
    Address VARCHAR(200),
        Time TIMESTAMP NOT NULL,
        Message VARCHAR(255)
);

CREATE TABLE Promotion (
        NotifID INT PRIMARY KEY,
        Company VARCHAR(100),
        StartDate DATE,
        EndDate DATE,
        FOREIGN KEY (NotifID) REFERENCES Notification(NotifID) ON DELETE
CASCADE,
        CHECK (EndDate > StartDate)
);

CREATE TABLE Alert (
        NotifID INT PRIMARY KEY,
        Type VARCHAR(100),
        FOREIGN KEY (NotifID) REFERENCES Notification(NotifID) ON DELETE
CASCADE
);

CREATE TABLE Receives (
        UserID INT,
        NotifID INT,
        PRIMARY KEY (UserID, NotifID),
        FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE
CASCADE,
        FOREIGN KEY (NotifID) REFERENCES Notification(NotifID) ON DELETE
CASCADE
);
```

8. **Insert Statements**

TransitStop
- INSERT INTO TransitStop (StopID, Type, Latitude, Longitude)
  VALUES (1, 'Bus', 40.712776, -74.005974);

- INSERT INTO TransitStop (StopID, Type, Latitude, Longitude)
  VALUES (2, 'Bus', 34.052235, -118.243683);

- INSERT INTO TransitStop (StopID, Type, Latitude, Longitude)
  VALUES (3, 'SkyTrain', 51.507351, -0.127758);

- INSERT INTO TransitStop (StopID, Type, Latitude, Longitude)
  VALUES (4, 'SkyTrain', 48.856613, 2.352222);

- INSERT INTO TransitStop (StopID, Type, Latitude, Longitude)
  VALUES (5, 'SeaBus', -33.868820, 151.209296);

Place
- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
  VALUES ('Stanley Park', 'Vancouver, BC V6G 1Z4', '(604) 123-4567', '08:00:00', '20:00:00', 'A large park with scenic views and trails.', 1);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
  VALUES ('Granville Island Public Market', '1669 Johnston St, Vancouver, BC V6H 3R9', '(604) 987-6543', '09:00:00', '18:00:00', 'A popular market offering fresh food, arts, and crafts.', 2);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
  VALUES ('Vancouver Aquarium', '845 Avison Way, Vancouver, BC V6G 3E2', '(604) 876-1234', '10:00:00', '17:00:00', 'Marine life exhibitions.', 3);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
  VALUES ('Science World', '1455 Quebec St, Vancouver, BC V6A 3Z7', '(604) 888-8888', '10:00:00', '18:00:00', 'Interactive science exhibits for all ages.', 4);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
  VALUES ('Capilano Suspension Bridge', '3735 Capilano Rd, North Vancouver, BC V7R 4J1', '(604) 654-3210', '09:00:00', '19:00:00', 'A long suspension bridge offering forest views.', 5);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Hills of Kerrisdale', '2125 W 41st Ave, Vancouver, BC V6M 1Z6', '(604) 266-9177', '10:00:00', '18:00:00', 'A boutique clothing store in the Kerrisdale shopping district.', 5);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Aritzia', '1100 Robson St, Vancouver, BC V6E 1B2', '(604) 684-3251', '10:00:00', '21:00:00', 'A popular womens fashion boutique known for trendy styles.', 4);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('MEC Vancouver', '111 2nd Ave E, Vancouver, BC V5T 1B4', '(604) 872-7858', '10:00:00', '20:00:00', 'A co-op outdoor gear store offering clothing and equipment for hiking, camping, and more.', 3);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Lululemon', '970 Robson St, Vancouver, BC V6Z 2E7', '(604) 681-3118', '10:00:00', '21:00:00', 'A popular athletic wear company.', 4);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Vancouver Pen Shop', '512 W Hastings St, Vancouver, BC V6B 1L5', '(604) 681-1612', '10:00:00', '17:00:00', 'A specialty shop offering luxury pens and writing instruments.', 2);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Vijs', '3106 Cambie St, Vancouver, BC V5Z 2W2', '(604) 736-6664', '17:00:00', '22:00:00', 'Famous for its modern Indian cuisine and no-reservation policy.', 1);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Miku', '200 Granville St #70, Vancouver, BC V6C 1S4', '(604) 568-3900', '11:30:00', '22:00:00', 'High-end Japanese restaurant specializing in aburi sushi.', 2);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)

VALUES ('Hawksworth Restaurant', '801 W Georgia St, Vancouver, BC V6C 1P7', '(604) 673-7000', '11:30:00', '23:00:00', 'Elegant fine dining with a focus on contemporary Canadian cuisine.', 3);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Medina Cafe', '780 Richards St, Vancouver, BC V6B 3A4', '(604) 879-3114', '08:00:00', '15:00:00', 'Popular brunch spot known for waffles and Mediterranean-inspired dishes.', 4);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Blue Water Cafe', '1095 Hamilton St, Vancouver, BC V6B 5T4', '(604) 688-8078', '17:00:00', '23:00:00', 'A top seafood restaurant located in the heart of Yaletown.', 5);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Grouse Mountain', '6400 Nancy Greene Way, North Vancouver, BC V7R 4K9', '(604) 555-1234', '09:00:00', '20:00:00', 'A popular mountain resort offering skiing, hiking, and scenic views.', 3);

- INSERT INTO Place (Name, Address, Phone, OpeningTime, ClosingTime, Description, StopID)
VALUES ('Museum of Vancouver', '1100 Chestnut St, Vancouver, BC V6J 3J9', '(604) 555-5678', '10:00:00', '17:00:00', 'A museum showcasing the history of Vancouver and its culture.', 5);

User
- INSERT INTO USERS (UserID, Phone, Email, Points)
VALUES (1, '1234567890', 'alice@example.com', 100);

- INSERT INTO USERS (UserID, Phone, Email, Points)
VALUES (2, '0987654321', 'bob@example.com', 150);

- INSERT INTO USERS (UserID, Phone, Email, Points)
VALUES (3, '1112223333', 'charlie@example.com', 200);

- INSERT INTO USERS (UserID, Phone, Email, Points)
VALUES (4, '4445556666', 'dave@example.com', 250);

- INSERT INTO USERS (UserID, Phone, Email, Points)
VALUES (5, '7778889999', 'eve@example.com', 300);

University of British Columbia, Vancouver
Department of Computer Science

EmailName
- INSERT INTO EmailName (Email, Name)
  VALUES ('alice@example.com', 'Alice Johnson');

- INSERT INTO EmailName (Email, Name)
  VALUES ('bob@example.com', 'Bob Smith');

- INSERT INTO EmailName (Email, Name)
  VALUES ('charlie@example.com', 'Charlie Adams');

- INSERT INTO EmailName (Email, Name)
  VALUES ('dave@example.com', 'Dave Williams');

- INSERT INTO EmailName (Email, Name)
  VALUES ('eve@example.com', 'Eve Thompson');

FriendsWith
- INSERT INTO FriendsWith (UserID1, UserID2)
  VALUES (1, 2);

- INSERT INTO FriendsWith (UserID1, UserID2)
  VALUES (1, 3);

- INSERT INTO FriendsWith (UserID1, UserID2)
  VALUES (1, 4);

- INSERT INTO FriendsWith (UserID1, UserID2)
  VALUES (1, 5);

- INSERT INTO FriendsWith (UserID1, UserID2)
  VALUES (2, 3);

GiftCard
- INSERT INTO GiftCard (GCID, Value, Franchise, UserID)
  VALUES (101, 50.00, 'Amazon', 1);

- INSERT INTO GiftCard (GCID, Value, Franchise, UserID)
  VALUES (102, 100.00, 'Starbucks', 2);

- INSERT INTO GiftCard (GCID, Value, Franchise, UserID)
  VALUES (103, 25.00, 'Walmart', 3);

- INSERT INTO GiftCard (GCID, Value, Franchise, UserID)
  VALUES (104, 75.00, 'Best Buy', 4);

- INSERT INTO GiftCard (GCID, Value, Franchise, UserID)
  VALUES (105, 150.00, 'Target', 5);

GCPoints

- INSERT INTO GCPoints (Value, Points)
  VALUES (50.00, 100);

- INSERT INTO GCPoints (Value, Points)
  VALUES (100.00, 200);

- INSERT INTO GCPoints (Value, Points)
  VALUES (25.00, 50);

- INSERT INTO GCPoints (Value, Points)
  VALUES (75.00, 150);

- INSERT INTO GCPoints (Value, Points)
  VALUES (150.00, 300);

TravelPass

- INSERT INTO TravelPass (PassID, Name, Cost, StartDate, EndDate, UserID)
  VALUES (1, 'City Explorer', 120.00, TO_DATE('2024-10-01', 'YYYY-MM-DD'),
  TO_DATE('2024-10-07', 'YYYY-MM-DD'), 1);

- INSERT INTO TravelPass (PassID, Name, Cost, StartDate, EndDate, UserID)
  VALUES (2, 'Outdoor Fun', 250.00, TO_DATE('2024-11-05', 'YYYY-MM-DD'),
  TO_DATE('2024-11-12', 'YYYY-MM-DD'), 2);

- INSERT INTO TravelPass (PassID, Name, Cost, StartDate, EndDate, UserID)
  VALUES (3, 'Mountain Adventure', 300.00, TO_DATE('2024-12-01',
  'YYYY-MM-DD'), TO_DATE('2024-12-10', 'YYYY-MM-DD'), 3);

- INSERT INTO TravelPass (PassID, Name, Cost, StartDate, EndDate, UserID)
  VALUES (4, 'History Enthusiast', 180.00, TO_DATE('2024-09-20',
  'YYYY-MM-DD'), TO_DATE('2024-09-28', 'YYYY-MM-DD'), 4);

- INSERT INTO TravelPass (PassID, Name, Cost, StartDate, EndDate, UserID)
  VALUES (5, 'Cultural Experience', 150.00, TO_DATE('2024-10-15',
  'YYYY-MM-DD'), TO_DATE('2024-10-22', 'YYYY-MM-DD'), 5);

Includes

- INSERT INTO Includes (PassID, Name, Address)
  VALUES (4, 'Vancouver Aquarium', '845 Avison Way, Vancouver, BC V6G 3E2');

- INSERT INTO Includes (PassID, Name, Address)

VALUES (3, 'Grouse Mountain', '6400 Nancy Greene Way, North Vancouver, BC V7R 4K9');

- INSERT INTO Includes (PassID, Name, Address)
  VALUES (1, 'Science World', '1455 Quebec St, Vancouver, BC V6A 3Z7');

- INSERT INTO Includes (PassID, Name, Address)
  VALUES (2, 'Capilano Suspension Bridge', '3735 Capilano Rd, North Vancouver, BC V7R 4J1');

- INSERT INTO Includes (PassID, Name, Address)
  VALUES (5, 'Museum of Vancouver', '1100 Chestnut St, Vancouver, BC V6J 3J9');

RetailBusiness
- INSERT INTO RetailBusiness (Name, Address, Industry, PriceRange)
  VALUES ('Hills of Kerrisdale', '2125 W 41st Ave, Vancouver, BC V6M 1Z6', 'Clothing Boutique', '$$$');

- INSERT INTO RetailBusiness (Name, Address, Industry, PriceRange)
  VALUES ('Aritzia', '1100 Robson St, Vancouver, BC V6E 1B2', 'Fashion Retail', '$$$');

- INSERT INTO RetailBusiness (Name, Address, Industry, PriceRange)
  VALUES ('MEC Vancouver', '111 2nd Ave E, Vancouver, BC V5T 1B4', 'Outdoor Gear', '$$');

- INSERT INTO RetailBusiness (Name, Address, Industry, PriceRange)
  VALUES ('Lululemon', '970 Robson St, Vancouver, BC V6Z 2E7', 'Athletic Wear', '$$$');

- INSERT INTO RetailBusiness (Name, Address, Industry, PriceRange)
  VALUES ('Vancouver Pen Shop', '512 W Hastings St, Vancouver, BC V6B 1L5', 'Specialty Retail', '$$$');

Restaurant
- INSERT INTO Restaurant (Name, Address, Cuisine, Capacity)
  VALUES ('Vijs', '3106 Cambie St, Vancouver, BC V5Z 2W2', 'Indian', 100);

- INSERT INTO Restaurant (Name, Address, Cuisine, Capacity)
  VALUES ('Miku', '200 Granville St #70, Vancouver, BC V6C 1S4', 'Japanese', 150);

- INSERT INTO Restaurant (Name, Address, Cuisine, Capacity)

VALUES ('Hawksworth Restaurant', '801 W Georgia St, Vancouver, BC V6C 1P7', 'Canadian', 120);

- INSERT INTO Restaurant (Name, Address, Cuisine, Capacity)
  VALUES ('Medina Cafe', '780 Richards St, Vancouver, BC V6B 3A4', 'Mediterranean', 60);

- INSERT INTO Restaurant (Name, Address, Cuisine, Capacity)
  VALUES ('Blue Water Cafe', '1095 Hamilton St, Vancouver, BC V6B 5T4', 'Seafood', 180);

Reviews
- INSERT INTO Reviews (UserID, Name, Address, ReviewDate, Rating, Message)
  VALUES (1, 'Stanley Park', 'Vancouver, BC V6G 1Z4', TO_DATE('2024-10-01', 'YYYY-MM-DD'), 5, 'Beautiful park with amazing views and trails.');

- INSERT INTO Reviews (UserID, Name, Address, ReviewDate, Rating, Message)
  VALUES (2, 'Miku', '200 Granville St #70, Vancouver, BC V6C 1S4', TO_DATE('2024-09-28', 'YYYY-MM-DD'), 4, 'Delicious sushi, though a bit pricey.');

- INSERT INTO Reviews (UserID, Name, Address, ReviewDate, Rating, Message)
  VALUES (3, 'Vijs', '3106 Cambie St, Vancouver, BC V5Z 2W2', TO_DATE('2024-09-30', 'YYYY-MM-DD'), 5, 'The best Indian food I have ever tasted!');

- INSERT INTO Reviews (UserID, Name, Address, ReviewDate, Rating, Message)
  VALUES (4, 'Blue Water Cafe', '1095 Hamilton St, Vancouver, BC V6B 5T4', TO_DATE('2024-09-25', 'YYYY-MM-DD'), 4, 'Amazing seafood and great atmosphere.');

- INSERT INTO Reviews (UserID, Name, Address, ReviewDate, Rating, Message)
  VALUES (5, 'Medina Cafe', '780 Richards St, Vancouver, BC V6B 3A4', TO_DATE('2024-10-02', 'YYYY-MM-DD'), 5, 'Best brunch in Vancouver, their waffles are outstanding!');

Event
- INSERT INTO Event (EventID, Name, Address, Title, EventDate, Rating, Description)
  VALUES (1, 'Stanley Park', 'Vancouver, BC V6G 1Z4', 'Light Show', TO_DATE('2024-12-01', 'YYYY-MM-DD'), 5, 'A magical light show at the park.');

- INSERT INTO Event (EventID, Name, Address, Title, EventDate, Rating, Description)
  VALUES(2, 'Granville Island Public Market', '1669 Johnston St, Vancouver, BC V6H 3R9', 'Christmas Market', TO_DATE('2024-12-15', 'YYYY-MM-DD'), 4, 'Holiday-themed market.');

- INSERT INTO Event (EventID, Name, Address, Title, EventDate, Rating, Description)
  VALUES(3, 'Vancouver Aquarium', '845 Avison Way, Vancouver, BC V6G 3E2', 'Shark Week', TO_DATE('2024-11-10', 'YYYY-MM-DD'), 5, 'Special exhibits on sharks.');

- INSERT INTO Event (EventID, Name, Address, Title, EventDate, Rating, Description)
  VALUES(4, 'Science World', '1455 Quebec St, Vancouver, BC V6A 3Z7', 'New Tech Expo', TO_DATE('2024-11-01', 'YYYY-MM-DD'), 4, 'Showcasing latest technology.');

- INSERT INTO Event (EventID, Name, Address, Title, EventDate, Rating, Description)
  VALUES(5, 'Capilano Suspension Bridge', '3735 Capilano Rd, North Vancouver, BC V7R 4J1', 'Bridge Illumination', TO_DATE('2024-12-20', 'YYYY-MM-DD'), 5, 'The bridge is lit with thousands of lights.');

Notification
- INSERT INTO Notification (NotifID, Time, Message)
  VALUES (1, TO_TIMESTAMP('2024-11-01 10:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'New event: Light Show at Stanley Park');

- INSERT INTO Notification (NotifID, Time, Message)
  VALUES (2, TO_TIMESTAMP('2024-11-15 08:30:00', 'YYYY-MM-DD HH24:MI:SS'), 'Promotion: 20% off on Christmas Market tickets');

- INSERT INTO Notification (NotifID, Time, Message)
  VALUES (3, TO_TIMESTAMP('2024-11-05 09:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Shark Week is back at Vancouver Aquarium');

- INSERT INTO Notification (NotifID, Time, Message)

VALUES (4, TO_TIMESTAMP('2024-11-20 12:00:00', 'YYYY-MM-DD HH24:MI:SS'), 'Check out the New Tech Expo at Science World');

- INSERT INTO Notification (NotifID, Time, Message)
VALUES (5, TO_TIMESTAMP('2024-11-25 07:45:00', 'YYYY-MM-DD HH24:MI:SS'), 'Capilano Bridge Illumination tickets are available.');

Promotion
- INSERT INTO Promotion (NotifID, Company, StartDate, EndDate)
VALUES(1, 'Stanley Park Society', TO_DATE('2024-11-01', 'YYYY-MM-DD'), TO_DATE('2024-12-01', 'YYYY-MM-DD'));

- INSERT INTO Promotion (NotifID, Company, StartDate, EndDate)
VALUES (2, 'Granville Island', TO_DATE('2024-11-15', 'YYYY-MM-DD'), TO_DATE('2024-12-20', 'YYYY-MM-DD'));

- INSERT INTO Promotion (NotifID, Company, StartDate, EndDate)
VALUES(3, 'Vancouver Aquarium', TO_DATE('2024-11-10', 'YYYY-MM-DD'), TO_DATE('2024-12-10', 'YYYY-MM-DD'));

- INSERT INTO Promotion (NotifID, Company, StartDate, EndDate)
VALUES(4, 'Science World', TO_DATE('2024-11-01', 'YYYY-MM-DD'), TO_DATE('2024-11-30', 'YYYY-MM-DD'));

- INSERT INTO Promotion (NotifID, Company, StartDate, EndDate)
VALUES(5, 'Capilano Suspension Bridge', TO_DATE('2024-12-01', 'YYYY-MM-DD'), TO_DATE('2024-12-31', 'YYYY-MM-DD'));

Alert
- INSERT INTO Alert (NotifID, Type)
VALUES(1, 'Reminder');
- INSERT INTO Alert (NotifID, Type)
VALUES(2, 'Promotion');
- INSERT INTO Alert (NotifID, Type)
VALUES(3, 'Event Announcement');
- INSERT INTO Alert (NotifID, Type)
VALUES(4, 'Event Update');
- INSERT INTO Alert (NotifID, Type)

VALUES(5, 'Special Offer');

Receives
- INSERT INTO Receives (UserID, NotifID) VALUES(1, 1);
- INSERT INTO Receives (UserID, NotifID) VALUES(2, 2);
- INSERT INTO Receives (UserID, NotifID) VALUES(3, 3);
- INSERT INTO Receives (UserID, NotifID) VALUES(4, 4);
- INSERT INTO Receives (UserID, NotifID) VALUES(5, 5);