



## CPSC 304 Project Cover Page

Milestone #: **2**

Date: **March 1, 2023**

Group #: **45**

Name	Student Number	CS ID	Email
Flora Zhou	17977596	h5t4s	flozhou@student.ubc.ca
Jasvir Sandhu	26638189	b8f8u	jsand01@students.cs.ubc.ca
Payam Forouzandeh	51597292	l5n2p	payamfz@student.ubc.ca

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

# Summary

Our database project models a public transit system and tries to capture necessary information about transit lines, vehicles, commuters and employees. It allows for administrative supervision, rider account and payments management, and real-time schedule information for client-side applications.

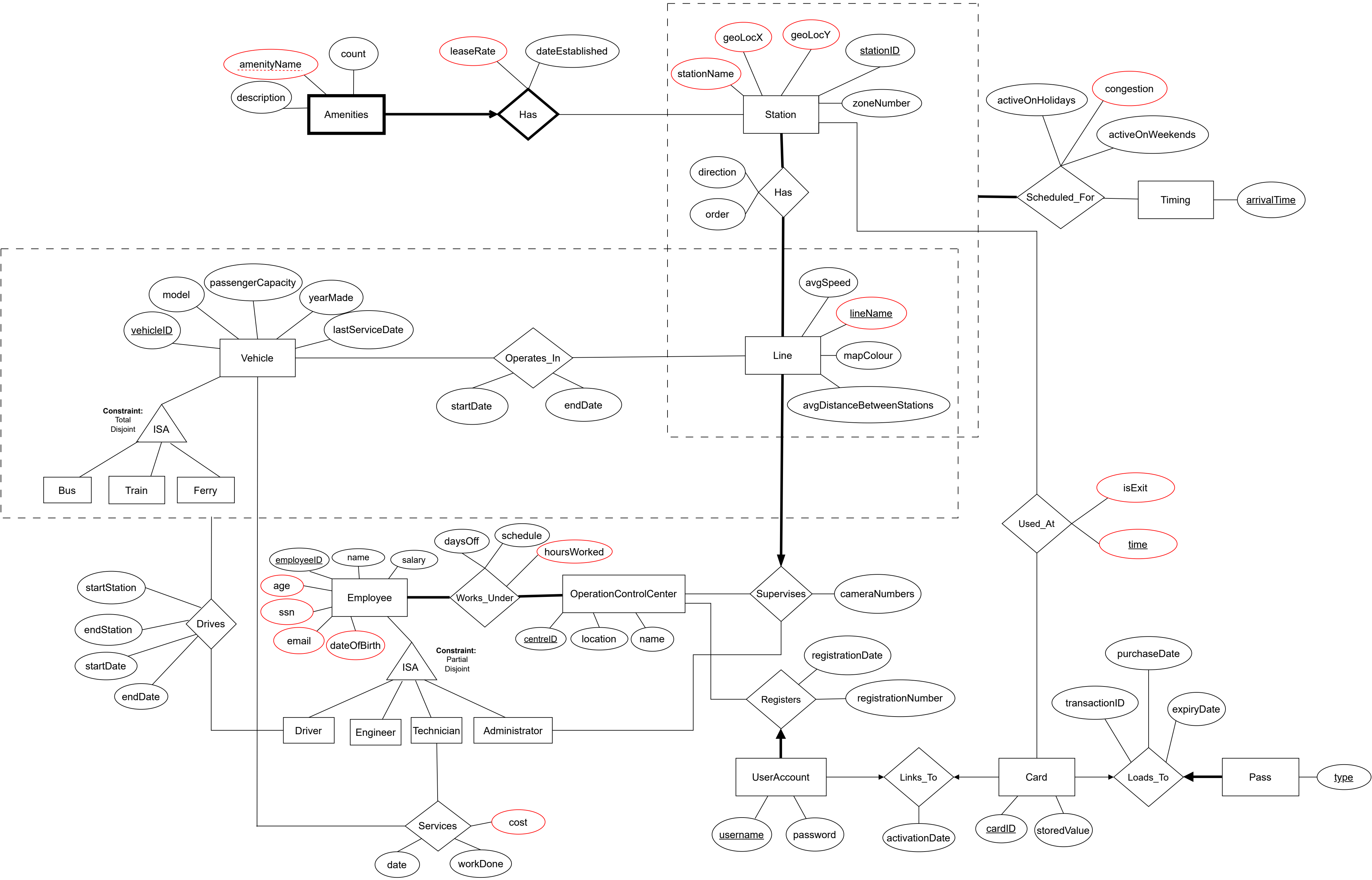
## ER Diagram

Changes done to the diagram:

1. Removed departureTime from Timing because it was redundant
2. renamed 'name' attributes of Line, Station, and Amenity to lineName, stationName, and amenityName respectively.
3. Replaced 'address' attribute of Station to geoLocX and geoLocY to have more interesting functional dependencies.
4. renamed 'lease' attribute of amenity to 'leaseRate' for better clarity.
5. moved 'averageWaitTime' of Station to the 'Scheduled\_For' aggregate relationship since the crowdedness of a station depends on which line it is for and at what time. Renamed it to 'congestion' to be more descriptive.
6. added 'isExit' attribute to the Used\_At relation to distinguish between entrance or exit.
7. Added 'hoursWorked' attribute to the Works\_Under relation to allow for meaningful BCNF decomposition.
8. Added 'ssn', 'dateofBirth', 'age', 'email' to Employee to allow for meaningful BCNF decomposition.
9. Added 'cost' to Services to allow for meaningful BCNF decomposition.

We chose to keep the subtypes in our ISA hierarchy schema for Vehicle because it is total and disjoint, requiring the vehicle to be one of the subtypes. We chose to keep both the superclass and subtypes in our ISA hierarchy schema for Employee because it is partial and disjoint, an employee may not be one of the listed subtypes but can not fall under more than 1 of the listed subtypes (i.e, an employee cannot be both an engineer and technician).

ER Diagram



Note: Throughout all the following sections, each number is corresponding to a specific relation.

## Database Schema

1. Station(stationID:integer, geoLocX:float, geoLocY:float, stationName:vchar, zoneNumber:integer)
  - Primary Key: stationID
  - Candidate Key: (geoLocX, geoLocY)
2. Line(lineName:vchar, avgSpeed:integer, mapColour:char, avgDistanceBetweenStations:integer)
  - Primary Key: lineName
3. Timing(arrivalTime:time)
  - Primary Key: arrivalTime
4. Station\_Has\_Amenities(amenityName:vchar, stationID:integer, count:integer, description:vchar, leaseRate:float, dateEstablished:date)
  - Primary Key: amenityName, stationID
  - Foreign Key: stationID REFERENCES Station
5. Line\_Has\_Station(stationID:integer, lineName:vchar, direction:char, order:integer)
  - Primary Key: stationID, lineName
  - Foreign Key: stationID REFERENCES Station, lineName REFERENCES Line
6. StationLine\_Scheduled\_For\_Timing(stationID:integer, lineName:vchar, arrivalTime:time, activeOnHolidays:boolean, activeOnWeekends:boolean, congestion:integer)
  - Primary Key: stationID, lineName, arrivalTime
  - Foreign Key: (stationID, lineName) REFERENCES Station\_Has\_Line, arrivalTime REFERENCES Timing
7. Card\_Used\_At\_Station(cardID:integer, stationID:integer, time:datetime, isExit:boolean)
  - Primary Key: stationID, cardID, time
  - Foreign Key: stationID REFERENCES Station, cardID REFERENCES Card
8. OperationControlCenter(centreID: char, location: char, name: char)
  - Primary Key: centreID
  - Candidate Key: name
  - Not NULL: location, name
9. Supervises\_Line(lineName: char, centreID: char, cameraNumbers: integer)
  - Primary Key: lineName
  - Not NULL: centreID
  - Foreign Key: lineName REFERENCES Line, centreID REFERENCES Operation Control Centre

10. Works\_Under(**employeeID**: char, **centreID**: char, daysOff: integer, schedule: varchar, hoursWorked: integer)
- Primary Key: employeeID, centreID
  - Foreign Key: centreID REFERENCES OperationControlCentre, employeeID REFERENCES Employee
11. UserAccount(**username**: varchar, password: varchar)
- Primary Key: username
  - Not NULL: password
12. UserAccount\_Registers(**username**: varchar, **centreID**: char, registrationDate: datetime, registrationNumber: integer)
- Primary Key: username
  - Candidate Key: registrationNumber
  - Not NULL: centreID, registrationDate
  - Foreign Key: username REFERENCES UserAccount, centreID REFERENCES OperationControlCentre
13. Card(**cardID**: char, storedValue: decimal)
- Primary Key: cardID
  - Not NULL: storedValue
14. Card\_Links\_To(**cardID**: char, **username**: varchar, activationDate: timedate)
- Primary Key: cardID
  - Is Unique: username
  - Foreign Key: cardID REFERENCES Card, username REFERENCES UserAccount
15. Pass\_Loads\_To(**type**: char, transactionID: char, purchaseDate: datetime, expiryDate: date, **cardID**: char)
- Primary Key: type
  - Foreign Key: cardID REFERENCES Card
  - Is unique: transactionID, cardID
16. Bus(**vehicleID**: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
- Primary Key: vehicleID
  - Not Null: model, passengerCapacity
17. Train(**vehicleID**: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
- Primary Key: vehicleID
  - Not Null: model, passengerCapacity
18. Ferry(**vehicleID**: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
- Primary Key: vehicleID

- Not Null: model, passengerCapacity
19. Employee(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
- Primary Key: employeeID
  - Candidate Keys: ssn, email
  - Not Null: name
20. Driver(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
- Primary Key: employeeID
  - Candidate Keys: ssn, email
  - Foreign Key: employeeID REFERENCES Employee
  - Not Null: name
21. Engineer(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
- Primary Key: employeeID
  - Candidate Keys: ssn, email
  - Foreign Key: employeeID REFERENCES Employee
  - Not Null: name
22. Technician(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
- Primary Key: employeeID
  - Candidate Keys: ssn, email
  - Foreign Key: employeeID REFERENCES Employee
  - Not Null: name
23. Administrator(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
- Primary Key: employeeID
  - Candidate Keys: ssn, email
  - Foreign Key: employeeID REFERENCES Employee
  - Not Null: name
24. Operates\_In(vehicleID: char, lineName: char, startDate: date, endDate: date)
- Primary Key: vehicleID, lineName
  - Foreign Key: vehicleID REFERENCES Vehicle, lineName REFERENCES Line
  - Not Null: startDate
25. Services(employeeID: char, vehicleID: char, cost: integer, workDone: char, date: date)
- Primary Key: employeeID, vehicleID
  - Foreign Key: employeeID REFERENCES Technician, vehicleID REFERENCES Vehicle

- Not Null: workDone
- 26. Drives(**employeeID**: char, **vehicleID**: char, **lineName**: char, startStation: char, endStation: char, startDate: char, endDate)
- Primary Key: employeeID, vehicleID, lineName
- Foreign Key: employeeID REFERENCES Driver, (vehicleID, lineName) REFERENCES Operates\_In

## Functional Dependencies

1. Station
  - stationID -> geoLocX, geoLocY, stationName, zoneNumber
  - geoLocX, geoLocY -> stationName, zoneNumber
2. Line
  - lineName -> avgSpeed, mapColour, avgDistanceBetweenStations
3. Timing
  - N/A
4. Station\_Has\_Amenities
  - lineName, stationID -> count, description, leaseRate, dateEstablished
5. Line\_Has\_Station
  - stationID, lineName -> direction, order
6. StationLine\_Scheduled\_For\_Timing
  - stationID, lineName, arrivalTime -> activeOnHolidays, activeOnWeekends, congestion
7. Card\_Used\_At\_Station
  - cardID, stationID, time -> isExit
8. OperationControlCenter
  - centreID -> location, name
9. Supervises\_Line
  - lineName, centreID → cameraNumbers
10. Works\_Under
  - centreID, employeeID → hoursWorked
  - employeeID → daysOff, schedule
11. UserAccount
  - username → password
12. UserAccount\_Registers
  - username → registrationDate, registrationNumber, centreID
13. Card
  - cardID → storedValue

14. Card\_Links\_To

- username, cardID → activationDate

15. Pass\_Loads\_To

- cardID, transactionID → type, purchaseDate, expiryDate

16. Bus

- vehicleID → model, passengerCapacity, yearMade, lastServiceDate
- model, yearMade → passengerCapacity

17. Train

- vehicleID → model, passengerCapacity, yearMade, lastServiceDate
- model, yearMade → passengerCapacity

18. Ferry

- vehicleID → model, passengerCapacity, yearMade, lastServiceDate
- model, yearMade → passengerCapacity

19. Employee

- employeeID → name, ssn, age, dateofBirth, salary, email
- ssn → name, employeeID, age, dateOfBirth
- email → name, employeeID
- dateOfBirth → age

20. Driver

- employeeID → name, ssn, age, dateofBirth, salary, email
- ssn → name, employeeID, age, dateOfBirth
- email → name, employeeID
- dateOfBirth → age

21. Engineer

- employeeID → name, ssn, age, dateofBirth, salary, email
- ssn → name, employeeID, age, dateOfBirth
- email → name, employeeID
- dateOfBirth → age

22. Technician

- employeeID → name, ssn, age, dateofBirth, salary, email
- ssn → name, employeeID, age, dateOfBirth
- email → name, employeeID
- dateOfBirth → age

23. Administrator

- employeeID → name, ssn, age, dateofBirth, salary, email
- ssn → name, employeeID, age, dateOfBirth



- email → name, employeeID
  - dateOfBirth → age
24. Operates\_In
- vehicleID, lineName → startDate, endDate
25. Services
- employeeID, vehicleID → date, workDone, cost
  - vehicleID, date, workDone → employeeID
  - workDone → cost
26. Drives
- employeeID, vehicleID, lineName → startStation, endStation, startDate, endDate

## Normalization

### 1. Station

- geoLocX, geoLocY → stationName, zoneNumber violates BCNF. Therefore, we need to decompose it as follows:
  - a. Station(stationID, **geoLocX**, **geoLocY**)
    - geoLocX and geoLocY cannot be null.
  - b. Station\_Location(geoLocX, geoLocY, stationName, zoneNumber)

### 2. Works\_Under

- employeeID → daysOff, schedule violates BCNF. Therefore, we need to decompose it as follows:
  - a. Employee\_Works\_Under(employeeID, daysOff, schedule)
  - b. OperationControlCenter\_Works\_Under(employeeID, centreID)

### 3. Bus

- "[model, yearMade]+ = model, yearMade, passengerCapacity
- model, yearMade → passengerCapacity (not in BCNF)
  - Decompose Bus(vehicleID: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
    - R1: Vehicle\_Capacity(model, yearMade, passengerCapacity) (in BCNF)
    - R2: Bus(vehicleID, model, yearMade, lastServiceDate) (in BCNF)

### 4. Train

- "[model, yearMade]+ = model, yearMade, passengerCapacity
- model, yearMade → passengerCapacity (not in BCNF)

- Decompose Train(vehicleID: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
  - R1: Vehicle\_Capacity(model, yearMade, passengerCapacity) (in BCNF)
  - R2: Train(vehicleID, model, yearMade, lastServiceDate) (in BCNF)

#### 5. Ferry

- "[model, yearMade]+ = model, yearMade, passengerCapacity
- model, yearMade → passengerCapacity (not in BCNF)
  - Decompose Ferry(vehicleID: char, model: char, passengerCapacity: integer, yearMade: integer, lastServiceDate: date)
    - R1: Vehicle\_Capacity(model, yearMade, passengerCapacity) (in BCNF)
    - R2: Ferry(vehicleID, model, yearMade, lastServiceDate) (in BCNF)

#### 6. Employee

- "[employeeID]+ = name, ssn, age, dateofBirth, salary, email
- [ssn]+ = name, employeeID, age, dateOfBirth, ssn, salary, email
- [email]+ = email, name, employeeID, ssn, age, dateofBirth, salary
- [dateOfBirth]+ = dateOfBirth, age
- Ck: employeeID, ssn, email
- dateOfBirth → age (not in BCNF)
  - Decompose Employee(employeeID: char, name: char, ssn: integer, salary: integer, email: char, age: integer, dateOfBirth: char)
    - R1: Employee\_Age(dateOfBirth, age) (in BCNF)
    - R2: Employee(employeeID, name, ssn, salary, email, dateOfBirth) (in BCNF)

#### 7. Driver

(same BCNF process as Employee)

- Employee\_Age(dateOfBirth, Age)
- Driver(employeeID, name, ssn, salary, email, dateOfBirth)

#### 8. Engineer

(same BCNF process as Employee)

- Employee\_Age(dateOfBirth, Age)
- Engineer(employeeID, name, ssn, salary, email, dateOfBirth)

### 9. Technician

(same BCNF process as Employee)

- Employee\_Age(dateOfBirth, Age)
- Technician(employeeID, name, ssn, salary, email, dateOfBirth)

### 10. Administrator

(same BCNF process as Employee)

- Employee\_Age(dateOfBirth, Age)
- Administrator(employeeID, name, ssn, salary, email, dateOfBirth)

### 11. Services

[workDone]+ = workDone, cost

- workDone → cost, schedule violates BCNF. Therefore, we need to decompose it as follows:
  - ServiceCost(workDone, cost)
  - Services(employeeID, vehicleID, workDone, date)

# SQL DDL statements

1a. CREATE TABLE Station (

stationID INTEGER PRIMARY KEY,  
geoLocX FLOAT NOT NULL,  
geoLocY FLOAT NOT NULL,  
FOREIGN KEY (geoLocX, geoLocY) REFERENCES Station\_Location ON DELETE CASCADE

)

1b. CREATE TABLE Station\_Location (

geoLocX FLOAT,  
geoLocY FLOAT,  
stationName VARCHAR(40),  
zoneNumber INTEGER,  
PRIMARY KEY (geoLocX, geoLocY)

)

2. CREATE TABLE Line (

lineName VARCHAR(20) PRIMARY KEY,  
avgSpeed INTEGER,  
mapColour CHAR(6),  
avgDistanceBetweenStations INTEGER

)

3. CREATE TABLE Timing (

arrivalTime TIME PRIMARY KEY

)

4. CREATE TABLE Station\_Has\_Amenities (

amenityName VARCHAR(40),  
stationID INTEGER,  
count INTEGER DEFAULT 1,  
description VARCHAR(50),  
leaseRate FLOAT,  
dateEstablished DATE,  
PRIMARY KEY (amenityName, stationID),  
FOREIGN KEY (stationID) REFERENCES Station ON DELETE CASCADE

)

5. CREATE TABLE Line\_Has\_Station (

stationID INTEGER,  
lineName CHAR(20),  
direction CHAR(1),

```

        order                INTEGER,
        PRIMARY KEY (stationID, lineName),
        FOREIGN KEY (stationID) REFERENCES Station ON DELETE CASCADE,
        FOREIGN KEY (lineName) REFERENCES Line ON DELETE CASCADE
    )

6. StationLine_Scheduled_For_Timing (
    stationID                INTEGER,
    lineName                  CHAR(20),
    arrivalTime               TIME,
    activeOnHolidays          BOOLEAN DEFAULT true,
    activeOnWeekends          BOOLEAN DEFAULT true,
    congestion                 INTEGER DEFAULT 0,
    PRIMARY KEY (stationID, lineName, arrivalTime)
    FOREIGN KEY (stationID, lineName) REFERENCES Station ON DELETE CASCADE
    FOREIGN KEY (arrivalTime) REFERENCES Timing ON DELETE SET NULL
)

7. Card_Used_At_Station(
    cardID                    INTEGER,
    stationID                  INTEGER,
    time                       DATETIME,
    isExit                     BOOLEAN DEFAULT false,
    PRIMARY KEY (cardID, stationID, time)
    FOREIGN KEY (cardID) REFERENCES Card,
    FOREIGN KEY (stationID) REFERENCES Station
)

8. CREATE TABLE OperationControlCenter (
    centreID                  CHAR(10),
    location                   CHAR(40) NOT NULL,
    name                       CHAR(40) NOT NULL,
    PRIMARY KEY (centreID),
    UNIQUE (name)
)

9. CREATE TABLE Supervises_Line (
    name                       CHAR(40),
    centreID                   CHAR(10) NOT NULL,
    cameraNumbers              INTEGER,
    PRIMARY KEY (lineName),
    FOREIGN KEY (lineName) REFERENCES Line

```

```
        ON DELETE CASCADE ON UPDATE CASCADE,  
        FOREIGN KEY (centreID) REFERENCES OperationControlCentre  
        ON DELETE CASCADE ON UPDATE CASCADE  
    )
```

```
10a. CREATE TABLE Employee_Works_Under (  
    employeeID      CHAR(10),  
    daysOff         INTEGER,  
    schedule        STRING,  
    PRIMARY KEY (employeeID),  
    FOREIGN KEY (employeeID) REFERENCES Employee  
    ON UPDATE CASCADE  
)
```

```
10b. CREATE TABLE OperationControlCenter_Works_Under (  
    employeeID      CHAR(10),  
    centreID        CHAR(10),  
    hoursWorked     INTEGER,  
    PRIMARY KEY (employeeID, centreID),  
    FOREIGN KEY (centreID) REFERENCES OperationControlCentre  
    ON UPDATE CASCADE,  
    FOREIGN KEY (employeeID) REFERENCES Employee  
    ON UPDATE CASCADE  
)
```

```
11. CREATE TABLE UserAccount (  
    username        VARCHAR(50),  
    password        VARCHAR(100) NOT NULL,  
    PRIMARY KEY (username)  
)
```

```
12. CREATE TABLE UserAccount_Registers (  
    username        VARCHAR(50),  
    centreID        CHAR(10) NOT NULL,  
    registrationDate DATETIME,  
    registrationNumber INTEGER,  
    PRIMARY KEY (username),  
    FOREIGN KEY (username) REFERENCES UserAccount,
```

```
        FOREIGN KEY (centreID) REFERENCES OperationControlCentre  
)
```

```
13. CREATE TABLE Card (  
    cardID          CHAR(10),  
    storedValue     DECIMAL DEFAULT 0,  
    PRIMARY KEY (cardID)  
)
```

```
14. CREATE TABLE Card_Links_To (  
    cardID          CHAR(10),  
    username        VARCHAR(50),  
    activationDate   TIMEDATE,  
    PRIMARY KEY (cardID),  
    UNIQUE (username),  
    FOREIGN KEY (cardID) REFERENCES Card  
    ON DELETE CASCADE,  
    FOREIGN KEY (username) REFERENCES UserAccount  
    ON UPDATE CASCADE  
)
```

```
15. CREATE TABLE Pass_Loads_To (  
    type            CHAR(1),  
    transactionID    CHAR(10),  
    purchaseDate     DATETIME  
    expiryDate       DATE  
    cardID           CHAR(10),  
    PRIMARY KEY (type),  
    UNIQUE (transactionID, cardID),  
    FOREIGN KEY (cardID) REFERENCES Card  
    ON DELETE CASCADE  
)
```

```
16. CREATE TABLE Vehicle_Capacity (  
    model VARCHAR(50),  
    yearMade INT,  
    passengerCapacity INT,  
    PRIMARY KEY (model, yearMade)
```

)

```
17. CREATE TABLE Bus (  
    vehicleID VARCHAR(20),  
    model VARCHAR(50), NOT NULL  
    yearMade INT, NOT NULL  
    lastServiceDate DATE,  
    PRIMARY KEY (vehicleID)  
    FOREIGN KEY (model, yearMade) REFERENCES Vehicle_Capacity ON UPDATE CASCADE
```

)

```
18. CREATE TABLE Train (  
    vehicleID VARCHAR(20),  
    model VARCHAR(50), NOT NULL  
    yearMade INT, NOT NULL  
    lastServiceDate DATE,  
    PRIMARY KEY (vehicleID)  
    FOREIGN KEY (model, yearMade) REFERENCES Vehicle_Capacity ON UPDATE CASCADE
```

)

```
19. CREATE TABLE Ferry (  
    vehicleID VARCHAR(20),  
    model VARCHAR(50), NOT NULL  
    yearMade INT, NOT NULL  
    lastServiceDate DATE  
    PRIMARY KEY (vehicleID)  
    FOREIGN KEY (model, yearMade) REFERENCES Vehicle_Capacity ON UPDATE CASCADE
```

)

```
20. CREATE TABLE Employee (  
    employeeID VARCHAR(20),  
    name VARCHAR(50), NAME  
    ssn INT(9),  
    salary INT(20),  
    email VARCHAR(50),  
    dateOfBirth VARCHAR(20),  
    PRIMARY KEY (employeeID)  
    UNIQUE (ssn, email)
```



```

        FOREIGN KEY (dateOfBirth) REFERENCES Employee_Age
    )
21. CREATE TABLE Employee_Age (
    dateOfBirth VARCHAR(20),
    age INT(3),
    PRIMARY KEY (dateOfBirth)
)

22. CREATE TABLE Driver (
    employeeID VARCHAR(20),
    name VARCHAR(50), NOT NULL
    ssn INT(9),
    salary INT(20),
    email VARCHAR(50),
    dateOfBirth VARCHAR(20),
    PRIMARY KEY (employeeID)
    UNIQUE (ssn, email)
    FOREIGN KEY (employeeID) REFERENCES Employee ON DELETE CASCADE
    FOREIGN KEY (dateOfBirth) REFERENCES Employee_Age
)

23. CREATE TABLE Engineer (
    employeeID VARCHAR(20),
    name VARCHAR(50), NOT NULL
    ssn INT(9),
    salary INT(20),
    email VARCHAR(50),
    dateOfBirth VARCHAR(20),
    PRIMARY KEY (employeeID)
    UNIQUE (ssn, email)
    FOREIGN KEY (employeeID) REFERENCES Employee ON DELETE CASCADE
    FOREIGN KEY (dateOfBirth) REFERENCES Employee_Age
)

24. CREATE TABLE Technician (
    employeeID VARCHAR(20),
    name VARCHAR(50), NOT NULL
    ssn INT(9),

```

```
    salary INT(20),
    email VARCHAR(50),
    dateOfBirth VARCHAR(20),
    PRIMARY KEY (employeeID)
    UNIQUE (ssn, email)
    FOREIGN KEY (employeeID) REFERENCES Employee ON DELETE CASCADE
    FOREIGN KEY (dateOfBirth) REFERENCES Employee_Age
)
```

```
25. CREATE TABLE Administrator (
    employeeID VARCHAR(20),
    name VARCHAR(50), NOT NULL
    ssn INT(9),
    salary INT(20),
    email VARCHAR(50),
    dateOfBirth VARCHAR(20),
    PRIMARY KEY (employeeID)
    UNIQUE (ssn, email)
    FOREIGN KEY (employeeID) REFERENCES Employee ON DELETE CASCADE
    FOREIGN KEY (dateOfBirth) REFERENCES Employee_Age
)
```

```
26. CREATE TABLE Operates_In (
    vehicleID VARCHAR(20),
    name VARCHAR(50),
    startDate DATE, NOT NULL
    endDate DATE,
    PRIMARY KEY (vehicleID, name)
    FOREIGN KEY (vehicleID) REFERENCES Vehicle ON DELETE CASCADE
    FOREIGN KEY (name) REFERENCES Line ON DELETE CASCADE
)
```

```
27. CREATE TABLE Services (
    employeeID VARCHAR(20),
    vehicleID VARCHAR(20),
    workDone VARCHAR(255), NOT NULL
    date DATE
    PRIMARY KEY (employeeID, vehicleID)
```

```
FOREIGN KEY (employeeID) REFERENCES Technician ON DELETE CASCADE
FOREIGN KEY (vehicleID) REFERENCES Vehicle
FOREIGN KEY (workDone) REFERENCES Service_Cost ON UPDATE CASCADE
)
```

```
28. CREATE TABLE Service_Cost (
    workDone VARCHAR(255),
    cost INT
    PRIMARY KEY (workDone)
)
```

```
29. CREATE TABLE Drives (
    employeeID VARCHAR(20),
    vehicleID VARCHAR(20),
    name VARCHAR(50),
    startStation VARCHAR(100), NOT NULL
    endStation VARCHAR(100), NOT NULL
    startDate DATE, NOT NULL
    end date DATE,
    PRIMARY KEY (employeeID, vehicleID, name)
    FOREIGN KEY (employeeID) REFERENCES Driver ON DELETE CASCADE
    FOREIGN KEY (vehicleID) REFERENCES Vehicle ON DELETE CASCADE
    FOREIGN KEY (name) REFERENCES Line ON UPDATE CASCADE
)
```

# Insert statements

## 1a. Station

- INSERT INTO Station VALUES (123, 49.258193, -123.214802)
- INSERT INTO Station VALUES (456, 49.285641, -123.111956)
- INSERT INTO Station VALUES (789, 49.285791, -123.119479)
- INSERT INTO Station VALUES (140, 49.323894, -123.111831)
- INSERT INTO Station VALUES (240, 49.262605, -123.069234)
- INSERT INTO Station VALUES (105, 49.285566, -123.120318)

## 1b. Station\_Location

- INSERT INTO Station\_Location  
VALUES (49.323894, -123.111831, 'Marine Dr @ Capilano Rd', 2)
- INSERT INTO Station\_Location  
VALUES (49.258193, -123.214802, '16thAve @ Blanca St', 1)
- INSERT INTO Station\_Location  
VALUES (49.285641, -123.111956, 'Waterfront Stn', 1)
- INSERT INTO Station\_Location  
VALUES (49.262605, -123.069234, 'Commercial-Broadway Stn', 2)
- INSERT INTO Station\_Location  
VALUES (49.285566, -123.120318, 'Burrard Stn', 1)

## 2. Line

- INSERT INTO Line VALUES ('Bus44', 50, '0335FC', 300)
- INSERT INTO Line VALUES ('SeaBus', 50, '4A433D', 2000)
- INSERT INTO Line VALUES ('CanadaLine', 120, '0099E0', 1000)
- INSERT INTO Line VALUES ('Bus14', 30, '0335FC', 100)
- INSERT INTO Line VALUES ('MillenniumLine', 120, 'FFDD00', 1000)

## 3. Timing

- INSERT INTO Timing VALUES ('07:00:00')
- INSERT INTO Timing VALUES ('07:10:00')
- INSERT INTO Timing VALUES ('07:20:00')
- INSERT INTO Timing VALUES ('23:10:00')
- INSERT INTO Timing VALUES ('12:12:00')
- INSERT INTO Timing VALUES ('12:45:00')

#### 4. Station\_Has\_Amenities

- INSERT INTO Station\_Has\_Amenities  
VALUES ('Starbucks', 456, 1, 'Coffeeshop', 2000.00, '2010-03-15')
- INSERT INTO Station\_Has\_Amenities  
VALUES ('Subway', 456, 1, 'Fast Food', 2200.00, '2009-10-01')
- INSERT INTO Station\_Has\_Amenities  
VALUES ('Ticket Booth', 456, 2, 'Purchase transit tickets', 0.00, '2006-01-01')
- INSERT INTO Station\_Has\_Amenities  
VALUES ('Vending Machine', 105, 4, 'Fast Food', 2200.00, '2009-10-01')
- INSERT INTO Station\_Has\_Amenities  
VALUES ('Ticket Booth', 105, 2, 'Purchase transit tickets', 0.00, '2002-01-01')

#### 5. Line\_Has\_Station

- INSERT INTO Line\_Has\_Station VALUES (456, 'CanadaLine', 'N', 10)
- INSERT INTO Line\_Has\_Station VALUES (240, 'MillenniumLine', 'E', 4)
- INSERT INTO Line\_Has\_Station VALUES (123, 'Bus14', 'E', 3)
- INSERT INTO Line\_Has\_Station VALUES (456, 'SeaBus', 'N', 1)
- INSERT INTO Line\_Has\_Station VALUES (140, 'Bus14', 'E', 48)

#### 6. StationLine\_Scheduled\_For\_Timing

- INSERT INTO StationLine\_Scheduled\_For\_Timing  
VALUES (456, 'CanadaLine', '07:00:00', true, true, 10)
- INSERT INTO StationLine\_Scheduled\_For\_Timing  
VALUES (456, 'CanadaLine', '23:10:00', true, true, 10)
- INSERT INTO StationLine\_Scheduled\_For\_Timing  
VALUES (123, 'Bus14', '12:12:00', true, true, NULL)
- INSERT INTO StationLine\_Scheduled\_For\_Timing  
VALUES (456, 'SeaBus', '23:10:00', false, false, 5)
- INSERT INTO StationLine\_Scheduled\_For\_Timing  
VALUES (240, 'MillenniumLine', '07:20:00', true, true, 8)

#### 7. Card\_Used\_At\_Station

- INSERT INTO Card\_Used\_At\_Station  
VALUES ('76bc7cbbd7', 123, '2022-10-22 07:00:00', false)
- INSERT INTO Card\_Used\_At\_Station  
VALUES ('76bc7cbbd7', 456, '2022-10-22 07:20:00', true)
- INSERT INTO Card\_Used\_At\_Station  
VALUES ('63a3ddb037', 789, '2019-10-10 23:10:00', false)

- INSERT INTO Card\_Used\_At\_Station  
VALUES ('8f7f669852', 140, '2022-10-23 07:10:00', false)
- INSERT INTO Card\_Used\_At\_Station  
VALUES ('76bc7cbbd7', 123, '2021- 08-01 23:10:00', false)

#### 8. OperationControlCenter

- INSERT INTO OperationControlCenter VALUES ('A8H6G2H837', 'Vancouver',  
'Broadway-City-Hall-Control-Center')
- INSERT INTO OperationControlCenter VALUES ('HW7NK2YD8K', 'Vancouver',  
'Mount-Pleasant-Center-of-Control')
- INSERT INTO OperationControlCenter VALUES ('H388HDLNAZ', 'Vancouver',  
'UBC-Lands-Control-Center')
- INSERT INTO OperationControlCenter VALUES ('7H2OJDBZGA', 'Vancouver',  
'River-District-Control-Center')
- INSERT INTO OperationControlCenter VALUES ('HE7BSKRUFH', 'Vancouver',  
'Dunbar-Center-of-Control')

#### 9. Supervises\_Line

- INSERT INTO Supervises\_Line VALUES ('CanadaLine', '7HEKBYA73H', 5)
- INSERT INTO Supervises\_Line VALUES ('MillenniumLine', 'EH7JDBH754', 4)
- INSERT INTO Supervises\_Line VALUES ('Bus14', 'E6HF3VWJD7', 3)
- INSERT INTO Supervises\_Line VALUES ('SeaBus', 'NHD63HABXJ', 1)
- INSERT INTO Supervises\_Line VALUES ('Bus15', 'E9FH3BSKAL', 12)

#### 10a. Employee\_Works\_Under

- INSERT INTO Employee\_Works\_Under VALUES ('1C2DB18462', 25, 'April 2023: 1, 3, 5, 7,  
10, 12')
- INSERT INTO Employee\_Works\_Under VALUES ('7H3IAHDBC8', 40, 'April 2023: 3, 10, 12,  
16, 24, 28')
- INSERT INTO Employee\_Works\_Under VALUES ('G4NSH7HGSG', 30, 'April 2023: 1, 4, 7, 8,  
10, 13, 16, 18, 22, 24, 27, 30')
- INSERT INTO Employee\_Works\_Under VALUES ('7J0WJ3JAHB', 50, 'April 2023: 3, 4, 7,  
28')
- INSERT INTO Employee\_Works\_Under VALUES ('1HQBCHEYAK', 31, 'April 2023: 2, 4, 7, 8,  
9, 12, 14, 17, 18, 21, 26, 27')

#### 10b. OperationControlCenter\_Works\_Under

- INSERT INTO OperationControlCenter\_Works\_Under VALUES ('1C2DB18462',  
'A8H6G2H837', 20)

- INSERT INTO OperationControlCenter\_Works\_Under VALUES ('7H3IAHDBC8', 'HW7NK2YD8K', 40)
- INSERT INTO OperationControlCenter\_Works\_Under VALUES ('G4NSH7HGSG', 'H388HDLNAZ', 57)
- INSERT INTO OperationControlCenter\_Works\_Under VALUES ('7J0WJ3JAHB', 'HE7BSKRUHF', 428)
- INSERT INTO OperationControlCenter\_Works\_Under VALUES ('1HQBCHEYAK', '7H2OJDBZGA', 252)

#### 11. UserAccount

- INSERT INTO UserAccount VALUES ('raymondng', 'bestprofever')
- INSERT INTO UserAccount VALUES ('ansonchung', 'bestTAever')
- INSERT INTO UserAccount VALUES ('flozhou', 'bbtenthusiast')
- INSERT INTO UserAccount VALUES ('jsand01', 'bcnfllegend')
- INSERT INTO UserAccount VALUES ('payamfz', 'anythingEnthusiast1')

#### 12. UserAccount\_Registers

- INSERT INTO UserAccount\_Registers VALUES ('flozhou', 'A8H6G2H837', '2023-02-01 14:34:27', 228158002)
- INSERT INTO UserAccount\_Registers VALUES ('jsand01', 'HW7NK2YD8K', '2023-02-15 03:23:52', 228158146)
- INSERT INTO UserAccount\_Registers VALUES ('jsand01', 'HW7NK2YD8K', '2023-02-16 12:12:43', 2281582232)
- INSERT INTO UserAccount\_Registers VALUES ('payamfz', 'H388HDLNAZ', '2023-02-17 21:42:21', 2281582343)
- INSERT INTO UserAccount\_Registers VALUES ('payamfz', '7H2OJDBZGA', '2023-02-19 11:22:56', 2281582122)

#### 13. Card

- INSERT INTO Card VALUES ('76bc7cbbd7', 10.45)
- INSERT INTO Card VALUES ('d62bafcb3e', 0.00)
- INSERT INTO Card VALUES ('63a3ddb037', 0.75)
- INSERT INTO Card VALUES ('8f7f669852', 20.00)
- INSERT INTO Card VALUES ('d5c9ee0b22', 13.60)

#### 14. Card\_Links\_To

- INSERT INTO Card\_Links\_To VALUES ('AF7905D714', 'serafinapetru', '2017-04-19 21:23:55')

- INSERT INTO Card\_Links\_To VALUES ('2C0EFC1076', 'felixkabujiya', '2018-02-19 09:32:23')
- INSERT INTO Card\_Links\_To VALUES ('10C154982E', 'emmaaisha', '2019-02-19 11:12:55')
- INSERT INTO Card\_Links\_To VALUES ('31333C2090', 'claudioshad', '2020-01-22 13:14:45')
- INSERT INTO Card\_Links\_To VALUES ('519552E6F3', 'danielomar', '2022-05-20 04:42:28')

#### 15. Pass\_Loads\_To

- INSERT INTO Pass\_Loads\_To VALUES ('M', '6HA86HBNCJ', '2023-03-01 04:42:28', '2023-04-01', AF7905D714)
- INSERT INTO Pass\_Loads\_To VALUES ('D', 'AFC052E87F', '2023-03-01 08:12:22', '2023-03-02', 2C0EFC1076)
- INSERT INTO Pass\_Loads\_To VALUES ('M', '7CEE183B35', '2023-03-01 14:12:45', '2023-04-01', 10C154982E)
- INSERT INTO Pass\_Loads\_To VALUES ('M', 'A6FA8EEFE2', '2023-03-01 22:52:12', '2023-04-01', 31333C2090)
- INSERT INTO Pass\_Loads\_To VALUES ('M', '24AD7F80B6', '2023-03-01 06:12:35', '2023-04-01', 519552E6F3)

#### 16. Vehicle\_Capacity

- INSERT INTO Vehicle\_Capacity VALUES ('Articulated', 2017, 75)
- INSERT INTO Vehicle\_Capacity VALUES ('Kahloke', 1973, 200)
- INSERT INTO Vehicle\_Capacity VALUES ('Minibus', 2015, 30)
- INSERT INTO Vehicle\_Capacity VALUES ('West Coast Express', 1995, 300)
- INSERT INTO Vehicle\_Capacity VALUES ('Malaspina Sky', 2008, 462)

#### 17. Bus

- INSERT INTO Bus VALUES ('2c994m', 'Minibus, 2012, 2011-06-07)
- INSERT INTO Bus VALUES ('adef70', 'Trolley', 2017, 2020-09-04)
- INSERT INTO Bus VALUES ('f0b2a1', 'Double-Decker', 2013, 2016-01-21)
- INSERT INTO Bus VALUES ('6d1d10', 'Diesel-Electric', 2015, 2020-09-04)
- INSERT INTO Bus VALUES ('c47259', 'Battery-Electric, 2020, 2022-04-17)

#### 18. Train

- INSERT INTO Train VALUES (daaa1f, 'Hyundai Rotem', 2011, 2021-07-19)
- INSERT INTO Train VALUES (e886a6, 'Bombardier Mark II', 2009, 2021-12-15)
- INSERT INTO Train VALUES (a60cf4, 'Bombardier Innovia Metro 300', 2016, 2020-02-25)
- INSERT INTO Train VALUES (236572, 'UTDC ICTS Mark I', 2005, 2016-03-17)
- INSERT INTO Train VALUES (fa1534, 'West Coast Express', 1995, 2022-06-29)



#### 19. Ferry

- INSERT INTO Ferry VALUES ( 'be2626', 'Kahloke', 1973, 2016-08-17)
- INSERT INTO Ferry VALUES ('9a8a61', 'Malaspina Sky', 2008, 2019-12-09)
- INSERT INTO Ferry VALUES ('efa24a', 'Kwuna', 1975, 2020-09-10)
- INSERT INTO Ferry VALUES ('c6f3f0', 'Coastal Celebration', 2008, 2021-09-22)
- INSERT INTO Ferry VALUES ('563cc2', 'Baynes Sound Connector', 2015, 2022-01-12)

#### 20. Employee

- INSERT INTO Employee VALUES (4832946783, 'Sandeep Nata', 264368554, 43000, nata@gmail.com, 1986-02-20)
- INSERT INTO Employee VALUES (6764274856, 'Florry Nasir', 807983843, 73500, nasir@gmail.com, 1991-09-27)
- INSERT INTO Employee VALUES (6561815739, 'Evgenios Yoana', 241361031, 95000, yoana@gmail.com, 1984-04-11)
- INSERT INTO Employee VALUES (8988005130, 'Patrizia Viktoryia', 173082338, 42000, viktoryia@gmail.com, 2000-04-20)
- INSERT INTO Employee VALUES (3434791957, 'Tabitha Timeus', 356673301, 67000, timeus@gmail.com, 1961-05-20)

#### 21. Employee\_Age

- INSERT INTO Employee\_Age VALUES (1969-06-26, 53)
- INSERT INTO Employee\_Age VALUES (1970-11-01, 55)
- INSERT INTO Employee\_Age VALUES (1986-02-20, 37)
- INSERT INTO Employee\_Age VALUES (1997-05-18, 25)
- INSERT INTO Employee\_Age VALUES (2000-01-18, 23)

#### 22. Driver

- INSERT INTO Driver VALUES (5810205722, 'Karlo Lesego', 725717803, 74000, 'lesego@gmail.com', 1998-03-13)
- INSERT INTO Driver VALUES (5303928084, 'Aurore Sigimund', 327221907, 82000, 'sigimund@gmail.com', 1997-03-06)
- INSERT INTO Driver VALUES (7529032867, 'Wilky Ioseph', 358249419, 52000, 'ioseph@gmail.com', 1997-11-30)
- INSERT INTO Driver VALUES (3850072567, 'Tresha Shanta', 600920298, 63000, 'shanta@gmail.com', 2000-11-29)
- INSERT INTO Driver VALUES (0080015868, 'Romeo Ghjuvanni', 844672084, 81000, 'ghjuvanni@gmail.com', 1993-10-23)

### 23. Engineer

- INSERT INTO Engineer VALUES (5612081378, 'Hodiyah Semele', 856716271, 112000, 'semele@gmail.com', 1964-08-23)
- INSERT INTO Engineer VALUES (1664253687, 'Rita Cristiana', 671854520, 120000, 'cristiana@gmail.com', 1968-09-06)
- INSERT INTO Engineer VALUES (3772031155, 'Noemí Fraser', 634385809, 89000, 'fraser@gmail.com', 1984-04-03)
- INSERT INTO Engineer VALUES (3634046822, 'Jani Walerian', 984718442, 97000, walerian@gmail.com', 1992-03-19)
- INSERT INTO Engineer VALUES (3433393223, 'Musa Minty', 291357204, 102000, 'minty@gmail.com', 1998-10-09)

### 24. Technician

- INSERT INTO Technician VALUES ('2957869336', 'Anna Rembert', '219703456', 51000, 'rembert@gmail.com', 1982-04-21)
- INSERT INTO Technician VALUES ('9246381570', 'Rowina Nelu', '950763812', 48000, 'nelu@gmail.com', 2000-01-20)
- INSERT INTO Technician VALUES ('1408726359', 'Mani Ayumu', '602348719', 57000, 'ayumu@gmail.com', 1972-07-15)
- INSERT INTO Technician VALUES ('7653204981', 'Oscar Arthur', '318297064', 60000, arthur@gmail.com', 1996-09-20)
- INSERT INTO Technician VALUES ('8675012943', 'Augusta Marcela', '612934507', 52000, marcela@gmail.com', 1993-02-03)

### 25. Administrator

- INSERT INTO Administrator VALUES (7356186759, 'Renia Bowie', 466112854, 57000, bowie@gmail.com, 1977-12-29)
- INSERT INTO Administrator VALUES (5251211307, 'Antiochos Tiwonge', 337005509, 62000, tiwonge@gmail.com, 1962-01-16)
- INSERT INTO Administrator VALUES (1375020921, 'Mirco Rosica', 095398894, 67000, rosica@gmail.com, 1981-04-16)
- INSERT INTO Administrator VALUES (4567109086, 'Leyla Blakely', 528305896, 56000, blakely@gmail.com, 1993-09-20)
- INSERT INTO Administrator VALUES (2750293038, 'Jenaro Ronan', 069734351, 61000, ronan@gmail.com, 1973-08-30)

### 26. Operates\_In

- INSERT INTO Operates\_In VALUES (8663d3, 'Bus14', 2019-07-31, 2022-06-27)

- INSERT INTO Operates\_In VALUES (9e8723, 'CanadaLine', 2021-01-21, null)
- INSERT INTO Operates\_In VALUES (4b6e41, 'Bus49', 2022-05-10, null)
- INSERT INTO Operates\_In VALUES (1db9a1, 'MilleniumLine', 2020-05-28, 2022-07-28)
- INSERT INTO Operates\_In VALUES (a73372, busR4', 2021-07-20, 2022-02-15)

## 27. Services

- INSERT INTO Services VALUES ('2957869336', '05ef9d', 'Oil Change', 2019-02-05)
- INSERT INTO Services VALUES ('9246381570', 'c6173a', 'Brake Check', 2023-01-28)
- INSERT INTO Services VALUES ('1408726359', '967c1w', 'Scheduled Maintenance', 2011-06-07)
- INSERT INTO Services VALUES ('7653204981', '2c994m', 'Air Filter Replacement', 2011-06-07)
- INSERT INTO Services VALUES('8675012943', '0ffa92', 'Battery Replacement', 2010-10-28)

## 28. Service\_Cost

- INSERT INTO Service\_Cost VALUES ("Oil Change", 60)
- INSERT INTO Service\_Cost VALUES ("Brake Check" 200)
- INSERT INTO Service\_Cost VALUES ("Battery Replacement" 300)
- INSERT INTO Service\_Cost VALUES ("Air Filter Replacement", 100)
- INSERT INTO Service\_Cost VALUES ("Scheduled Maintenance", 250)

## 29. Drives

- INSERT INTO Drives VALUES (7529032867, 8663d3, 'Bus14', 'UBC Exchange', 'Northbound Homer St', '2020-02-23, 2022-06-27)
- INSERT INTO Drives VALUES (0080015868, 9e8723, 'CanadaLine', 'Waterfront Station', 'YVR-Airport Station', 2022-05-10, 2022-08-18)
- INSERT INTO Drives VALUES (5810205722, 4b6e41, 'Bus49', 'Metrotown', 'UBC Exchange', 2021-03-12, 2023-01-28)
- INSERT INTO Drives VALUES (3850072567, 1db9a1, 'MilleniumLine', 'VCC-Clark Station', 'Lafarge Lake-Douglas Station', 2020-05-28, 2022-02-09)
- INSERT INTO Drives VALUES (5303928084, a73372, 'BusR4', 'Joyce Station', 'UBC Exchange', 2021-12-02, 2022-02-15)