University of British Columbia, Vancouver

Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2
Date: October 20, 2023
Group Number: 90

Name	Student Number	CS Aliases (Userid)	Preferred E-mail Address
Brandon Yuen	40390817	i8w2b	brandonyuen2001@gmail.com
Celine Chen	44176873	g4l8c	celinechen1114@gmail.com
Joshua Chew	95081204	u9b3b	joshuagchew@gmail.com

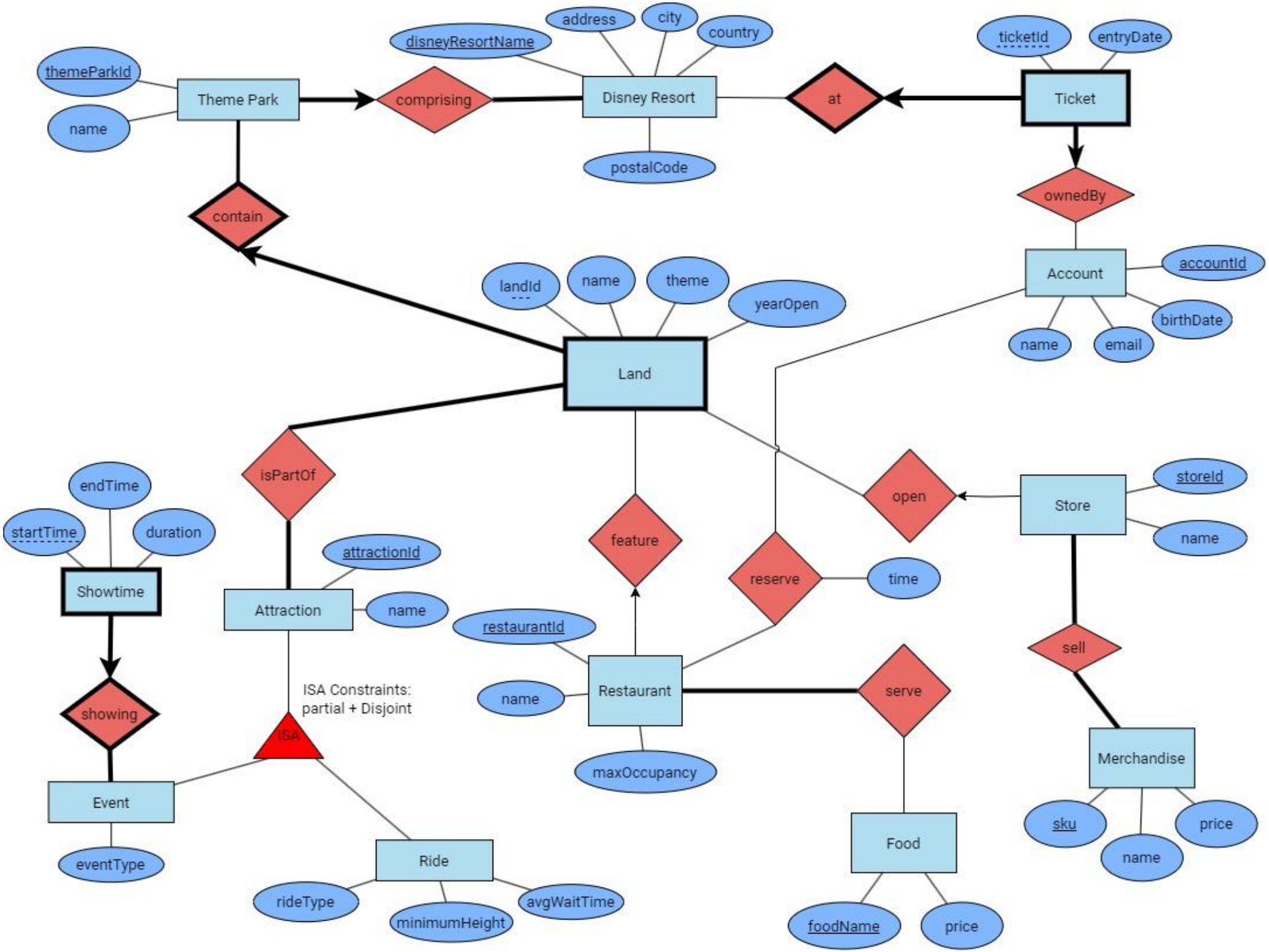
By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address,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

The domain of this application is the entertainment industry, specifically focused on Disney Resorts. It encompasses various aspects of providing information to guests about these resorts, their residing theme parks, and the smaller themed “lands” within those. This database will facilitate the efficient management of guest information, merchandise, food, and attractions found in each resort, such as rides and shows.

Changes Made to Our ER Diagram

1. Removed the entities - *People*, *Guests*, and *Employees* - and replaced them with *Account* to simplify the use case to an application for guests to get information about Disney Resorts instead of a general application for managing Disney Resorts.
2. Changed the *Reservation* entity name to *Ticket* because Ticket is a better suited name.
3. Added *Restaurant*, *Food*, *Merchandise*, and *Store* entities and their respective relationships: *feature, reserve, open, serve*, and *sell* to meet the minimum entity and relationship requirements.
4. Added a name attribute to the Attraction entity so that it has a non-primary key attribute.
5. Updated the naming convention of “id” attributes to also include [entity_name] in front of “id.” E.g. from “id” to “themeParkId.”
6. Added a postalCode attribute to Disney Resort and a duration attribute to Showtime so that there would be more non-trivial functional dependencies.


themeParkId
name
Theme Park
comprising
disneyResortName
address
city
country
Disney Resort
postalCode
at
ticketId
entryDate
Ticket
ownedBy
Account
accountId
birthDate
name
email
contain
landId
name
theme
yearOpen
Land
isPartOf
endTime
startTime
duration
Showtime
showing
Event
eventType
attractionId
Attraction
name
ISA Constraints:
partial + Disjoint
ISA
Ride
rideType
minimumHeight
avgWaitTime
feature
restaurantId
name
Restaurant
maxOccupancy
reserve
time
open
Store
storeId
name
sell
serve
Merchandise
sku
name
price
Food
foodName
price

Schema

1. DisneyResort(disneyResortName: varchar, address: varchar, city: varchar, country: varchar, postalCode: varchar)
   PK: disneyResortName
   CK: disneyResortName, {address, city, country}

2. ComprisingThemePark(themeParkId: integer, name: varchar, **disneyResortName**: varchar)
   PK: themeParkId
   CK: themeParkId, disneyResortName
   FK: disneyResortName references DisneyResort(disneyResortName)
   Not Null (participation): disneyResortName
   Not Null (semantic): name

3. LandContainThemePark(**themeParkId**: integer, landId: integer, name: varchar, theme: varchar, yearOpen: integer)
   PK: {landId, themeParkId}
   CK: {landId, themeParkId}, {name, themeParkId}
   FK: themeParkId references ComprisingThemePark(themeParkId)
   Not Null (semantic): name

4. IsPartOf(**themeParkId**: integer, **landId**: integer, **attractionId**: integer)
   PK: {themeParkId, landId, attractionId}
   CK: {themeParkId, landId, attractionId}
   FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandContainThemePark(landId), attractionId references Attraction(attractionId)

5. Attraction(attractionId: integer, name: varchar)
   PK: attractionId
   CK: attractionId, name
   Not Null (semantic): name

6. Ride(**attractionId**: integer, rideType: varchar, minimumHeight: integer, avgWaitTime: integer)
   PK: attractionId
   CK: attractionId
   FK: attractionId references Attraction(attractionId)

7. Event(**attractionId**: integer, eventType: varchar)
   PK: attractionId
   CK: attractionId
   FK: attractionId references Attraction(attractionId)

8. ShowtimeShowingEvent(**attractionId**: integer, startTime: integer, endTime: integer, duration: integer)
   PK: {attractionId, startTime}
   CK: {attractionId, startTime}
   FK: attractionId references Event(attractionId)

9. FeatureRestaurant(restaurantId: integer, name: varchar, maxOccupancy: integer, **themeParkId**: integer, **landId**: integer)
   PK: restaurantId
   CK: restaurantId
   FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandContainThemePark(LandId)
   Not Null (semantic): name

10. Serve(**restaurantId**: integer, **foodName**: varchar)
    PK: {restaurantId, foodName}
    CK: {restaurantId, foodName}
    FK: restaurantId references FeatureRestaurant(restaurantId), foodName references Food(foodName)

11. Food(foodName: varchar, price: float)
    PK: foodName
    CK: foodName

12. Account(accountId: integer, name: varchar, email: varchar, birthDate: integer)
    PK: accountId
    CK: accountId, email
    Not Null (semantic): email

13. Reserve(**accountId**: integer, **restaurantId**: integer, time: integer)
    PK: {accountId, restaurantId}

CK: {accountId, restaurantId}
FK: accountId references Account(accountId), restaurantId references FeatureRestaurant(restaurantId)

14. TicketAtDisneyResortOwnedByAccount(ticketId: integer, **disneyResortName**: varchar, entryDate: integer, **accountId**: integer)
    PK: {ticketId, disneyResortName}
    CK: {ticketId, disneyResortName}
    FK: disneyResortName references DisneyResort(disneyResortName), accountId references Account(accountId)
    Not Null (participation): accountId

15. OpenStore(storeId: integer, name: varchar, **themeParkId**: integer, **landId**: integer)
    PK: storeId
    CK: storeId
    FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandContainThemePark(landId)

16. Sell(**storeId**: integer, **sku**: integer)
    PK: {storeId, sku}
    CK: {storeId, sku}
    FK: storeId references OpenStore(storeId), sku references Merchandise(sku)

17. Merchandise(sku: integer, name: varchar, price: float)
    PK: sku
    CK: sku, name
    Not Null (semantic): name, price

Functional Dependencies

(valid FD other those identified by a PK or CK)

1. DisneyResort(disneyResortName, address, city, country, postalCode)
    a. disneyResortName -> address, city, country, postalCode,
    b. address, city, country -> disneyResortName, postalCode,
    c. postalCode -> city, country

2. ComprisingThemePark(themeParkId, name, disneyResortName)
    a. themeParkId -> name, disneyResortName
    b. name -> themeParkId, disneyResortName

3. LandContainThemePark(themeParkId, landId, name, theme, yearOpen)
    a. landId, themeParkId -> name, theme, yearOpen
    b. themeParkId, theme -> name
    c. name -> theme

4. IsPartOf(themeParkId, landId, attractionId)
    a. No non-trivial functional dependencies

5. Attraction(attractionId, name)
    a. attractionId -> name
    b. name -> attractionId

6. Ride(attractionId, rideType, minimumHeight, avgWaitTime)
    a. attractionId -> rideType, minimumHeight, avgWaitTime
    b. minimumHeight -> rideType

7. Event(attractionId, eventType)
    a. attractionId-> eventType

8. ShowtimeShowingEvent(attractionId, startTime, endTime, duration)
    a. startTime, attractionId-> endTime, duration
    b. startTime, endTime -> duration
    c. startTime, duration -> endTime

9. FeatureRestaurant(restaurantId, name, maxOccupancy, themeParkId, landId)
    a. restaurantId -> name, maxOccupancy, themeParkId, landId

10. Serve(restaurantId, foodName)
    a. No non-trivial functional dependencies

11. Food(foodName, price)
    a. foodName -> price

12. Account(accountId, name, email, birthDate)
    a. accountId -> name, email, birthDate
    b. email -> accountId, name, birthDate

13. Reserve(accountId, restaurantId, time)
    a. accountId, restaurantId -> time

14. TicketAtDisneyResortOwnedByAccount(ticketId, disneyResortName, entryDate, accountId)
    a. ticketId, disneyResortName -> entryDate, accountId

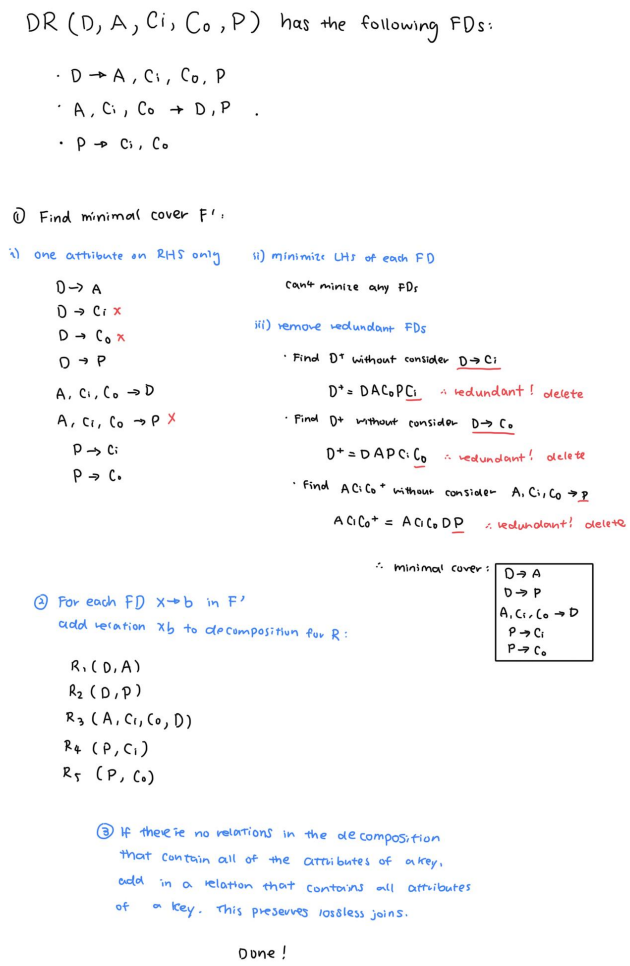
15. OpenStore(storeId, name, themeParkId, landId)
    a. storeId -> name, themeParkId, landId

16. Sell(storeId, sku)
    a. No non-trivial functional dependencies

17. Merchandise(sku, name, price)
    a. sku -> name, price
    b. name -> sku, price

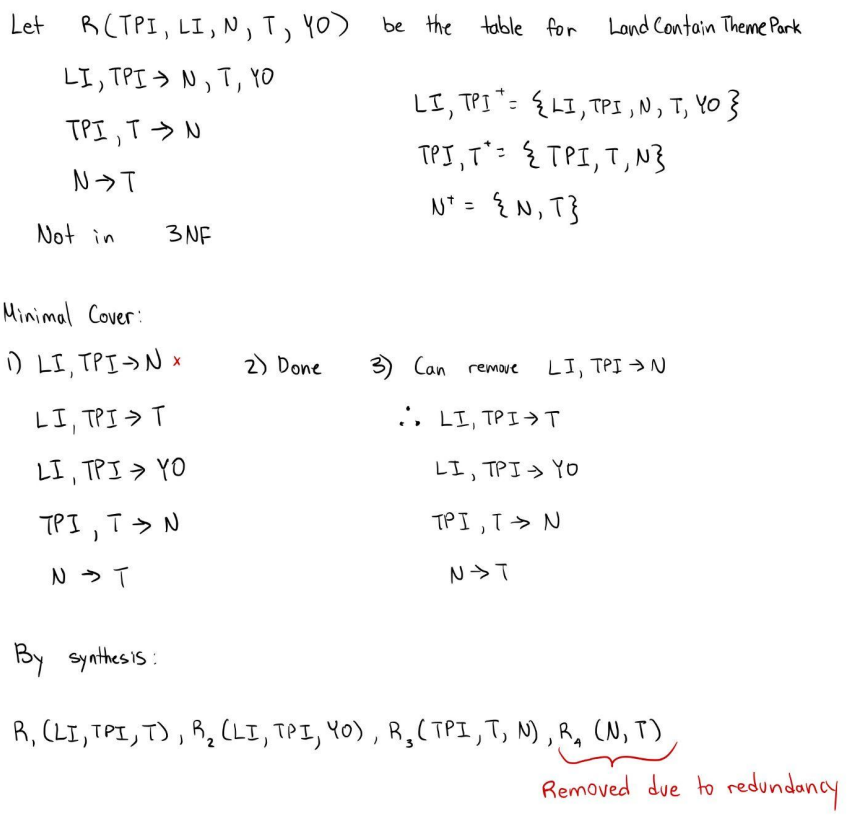
Normalization

1. (Original) DisneyResort(disneyResortName: varchar, address: varchar, city: varchar, country: varchar, postalCode: varchar)
   Post Normalization:
   a. DisneyResortAddress(disneyResortName: varchar, address: varchar)
   b. DisneyResortPostal(disneyResortName: varchar, postalCode: varchar)
   c. AddressCityCountry(disneyResortName: varchar, address: varchar, city: varchar, country: varchar)
   d. PostalCodeCity(postalCode: varchar, city: varchar)
      PK: postalCode
      CK: postalCode
   e. PostalCodeCountry(postalCode: varchar, country: varchar)
      PK: postalCode
      CK: postalCode



2. ComprisingThemePark(themeParkId: integer, name: varchar, **disneyResortName**: varchar)
    a. Already in BCNF
    FK: disneyResortName references DisneyResortAddress(disneyResortName)

3. (Original) LandContainThemePark(**themeParkId**: integer, landId: integer, name: varchar, theme: varchar, yearOpen: integer)
    Post Normalization:
    a. LandTheme(**themeParkId**: integer, landId: integer, theme: varchar)
    FK: themeParkId references ComprisingThemePark(themeParkId)
    b. LandYearOpen(**themeParkId**: integer, landId: integer, yearOpen: integer)
    FK: themeParkId references ComprisingThemePark(themeParkId)
    c. ThemeParkThemeName(**themeParkId**: integer, theme: varchar, name: varchar)
    PK: {themeParkId, theme}
    CK: {themeParkId, theme}
    FK: themeParkId references ComprisingThemePark(themeParkId)



4. IsPartOf(**themeParkId**: integer, **landId**: integer, **attractionId**: integer)
    a. Already in BCNF
    FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandTheme(landId), attractionId references Attraction(attractionId)

5. Attraction(attractionId: integer, name: varchar)
    a. Already in BCNF

6. (Original) Ride(**attractionId**: integer, rideType: varchar, minimumHeight: integer, avgWaitTime: integer)
    Post Normalization:
    a. RideMinimumHeight(**attractionId**: integer, minimumHeight: integer)
    FK: attractionId references Attraction(attractionId)
    b. RideAvgWaitTime(**attractionId**: integer, avgWaitTime: integer)
    FK: attractionId references Attraction(attractionId)
    c. MinimumHeightRideType(minimumHeight: integer, rideType: varchar)
    PK: minimumHeight
    CK: minimumHeight

Let R(AI, RT, MH, AWT) be the table for Ride

AI → RT, MH, AWT

MH → RT

Not in 3NF

$AI^+ = \{AI, RT, MH, AWT\}$

$MH^+ = \{MH, RT\}$

Minimal Cover:

1) AI → RT ✗
   AI → MH
   AI → AWT
   MH → RT

2) Done

3) Can remove AI → RT

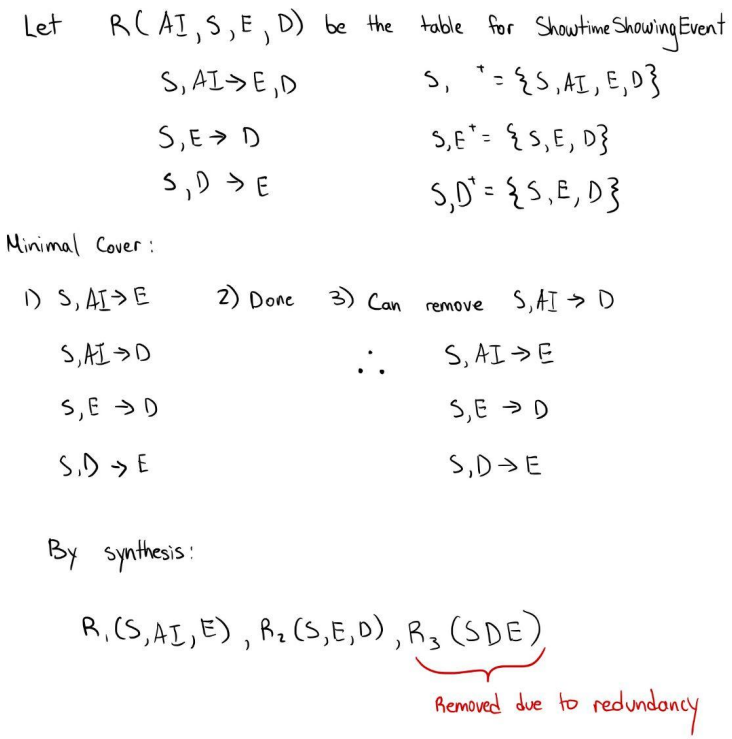
∴ AI → MH
AI → AWT
MH → RT

By synthesis

$R_1(AI, MH), R_2(AI, AWT), R_3(MH, RT)$

7. Event(**attractionId**: integer, eventType: varchar)
    a. Already in BCNF
    FK: attractionId references Attraction(attractionId)

8. (Original) ShowtimeShowingEvent(**attractionId**: integer, startTime: integer, endTime: integer, duration: integer)
    Post Normalization:
    a. ShowtimeEvent(**attractionId**: integer, startTime: integer, endTime: integer)
    PK: {attractionId, startTime}
    CK: {attractionId, startTime}
    FK: attractionId references Event(attractionId)
    b. ShowtimeDuration(startTime: integer, endTime: integer, duration: integer)
    PK: {startTime, endTime}
    CK: {startTime, endTime}



9. FeatureRestaurant(restaurantId: integer, name: varchar, maxOccupancy: integer, **themeParkId**: integer, **landId**: integer)
    a. Already in BCNF
    FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandTheme(LandId)

10. Serve(**restaurantId**: integer, **foodName**: varchar)
    a. Already in BCNF
       FK: restaurantId references FeatureRestaurant(restaurantId), foodName references Food(foodName)

11. Food(foodName: varchar, price: float)
    a. Already in BCNF

12. Account(accountId: integer, name: varchar, email: varchar, birthDate: integer)
    a. Already in BCNF

13. Reserve(**accountId**: integer, **restaurantId**: integer, time: integer)
    a. Already in BCNF
       FK: accountId references Account(accountId), restaurantId references FeatureRestaurant(restaurantId)

14. TicketAtDisneyResortOwnedByAccount(ticketId: integer, **disneyResortName**: varchar, entryDate: integer, **accountId**: integer)
    a. Already in BCNF
       FK: disneyResortName references DisneyResortAddress(disneyResortName), accountId references Account(accountId)

15. OpenStore(storeId: integer, name: varchar, **themeParkId**: integer, **landId**: integer)
    a. Already in BCNF
       FK: themeParkId references ComprisingThemePark(themeParkId), landId references LandTheme(landId)

16. Sell(**storeId**: integer, **sku**: integer)
    a. Already in BCNF
       FK: storeId references OpenStore(storeId), sku references Merchandise(sku)

17. Merchandise(sku: integer, name: varchar, price: float)
    a. Already in BCNF

SQL DDL

```sql
CREATE TABLE DisneyResortAddress(
    disneyResortName VARCHAR PRIMARY KEY,
    address VARCHAR
)
```

```sql
CREATE TABLE DisneyResortPostal(
    disneyResortName VARCHAR PRIMARY KEY,
    postalCode VARCHAR
)
```

```sql
CREATE TABLE AddressCityCountry(
    disneyResortName VARCHAR PRIMARY KEY,
    address VARCHAR,
    city VARCHAR,
    country VARCHAR
)
```

```sql
CREATE TABLE PostalCodeCity(
    postalCode VARCHAR PRIMARY KEY,
    city VARCHAR
)
```

```sql
CREATE TABLE PostalCodeCountry(
    postalCode VARCHAR PRIMARY KEY,
    country VARCHAR
)
```

```sql
CREATE TABLE ComprisingThemePark(
    themeParkId INT PRIMARY KEY,
    name VARCHAR UNIQUE NOT NULL,
    disneyResortName VARCHAR NOT NULL,
    FOREIGN KEY (disneyResortName)
        REFERENCES DisneyResort(disneyResortName)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

7.
```
CREATE TABLE LandTheme(
    themeParkId INT,
    landId INT,
    theme VARCHAR,
    PRIMARY KEY(themeParkId, landId),
    FOREIGN KEY (themeParkId)
        REFERENCES ComprisingThemePark(themeParkId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

8.
```
CREATE TABLE LandYearOpen(
    themeParkId INT,
    landId INT,
    theme VARCHAR,
    PRIMARY KEY(themeParkId, landId),
    FOREIGN KEY (themeParkId)
        REFERENCES ComprisingThemePark(themeParkId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

9.
```
CREATE TABLE ThemeParkThemeName(
    themeParkId INT,
    theme VARCHAR,
    name VARCHAR,
    PRIMARY KEY (themeParkId, theme),
    FOREIGN KEY (themeParkId)
        REFERENCES ComprisingThemePark(themeParkId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

10.
```
CREATE TABLE IsPartOf(
    themeParkId INT,
    landId INT,
    attractionId INT,
    PRIMARY KEY (themeParkId, landId, attractionId),
    FOREIGN KEY (themeParkId)
        REFERENCES LandTheme(themeParkId)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (landId)
        REFERENCES LandTheme(landId)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (attractionId)
        REFERENCES Attraction(attractionId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

11.
```
CREATE TABLE Attraction(
    attractionId INT PRIMARY KEY,
    name VARCHAR UNIQUE NOT NULL,
)
```

12.
```
CREATE TABLE RideMinimumHeight(
    attractionId INT PRIMARY KEY,
    minimumHeight INT,
    FOREIGN KEY (attractionId)
        REFERENCES Attraction(attractionId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

13.
```
CREATE TABLE RideAvgWaitTime(
    attractionId INT PRIMARY KEY,
    avgWaitTime INT,
    FOREIGN KEY (attractionId)
        REFERENCES Attraction(attractionId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

14.
```
CREATE TABLE MinimumHeightRideType(
    minimumHeight INT PRIMARY KEY,
    rideType VARCHAR
)
```

15.
```
CREATE TABLE Event(
    attractionId INT PRIMARY KEY,
    eventType VARCHAR,
    FOREIGN KEY (attractionId)
        REFERENCES Attraction(attractionId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

16.
```
CREATE TABLE ShowtimeEvent(
    attractionId INT,
    startTime INT,
    endTime INT,
    PRIMARY KEY (attractionId, startTime),
    FOREIGN KEY (attractionId)
        REFERENCES Event(attractionId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

17.
```
CREATE TABLE ShowtimeDuration(
    startTime INT,
    endTime INT,
    duration INT,
    PRIMARY KEY (startTime, endTime)
)
```

18.
```
CREATE TABLE FeatureRestaurant(
    restaurantId INT PRIMARY KEY,
    name VARCHAR NOT NULL,
    maxOccupancy INT,
    themeParkId INT,
    landId INT,
    FOREIGN KEY (themeParkId)
        REFERENCES LandTheme(themeParkId),
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (landId)
        REFERENCES LandTheme(landId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

19.
```
CREATE TABLE serve(
    restaurantId VARCHAR,
    foodName VARCHAR,
    PRIMARY KEY (restaurantId, foodName)
    FOREIGN KEY (restaurantId)
        REFERENCES FeatureRestaurant(restaurantId)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (foodName)
        REFERENCES Food(name)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

```
20. CREATE TABLE Food(
        foodName VARCHAR PRIMARY KEY,
        price INT
    )
```

```
21. CREATE TABLE Account(
        accountId INT PRIMARY KEY,
        name VARCHAR,
        email VARCHAR UNIQUE NOT NULL,
        birthDate INT
    )
```

```
22. CREATE TABLE Reserve(
        accountId INT,
        restaurantId INT,
        PRIMARY KEY (accountId, restaurantId),
        FOREIGN KEY (accountId)
            REFERENCES Account(accountId)
            ON DELETE CASCADE
            ON UPDATE CASCADE,
        FOREIGN KEY (restaurantId)
            REFERENCES FeatureRestaurant(restaurantId)
            ON DELETE CASCADE
            ON UPDATE CASCADE
    )
```

23.
```
CREATE TABLE TicketAtDisneyResortOwnedByAccount(
    ticketId INT,
    disneyResortName VARCHAR DEFAULT 'none',
    entryDate INT,
    accountId INT NOT NULL DEFAULT -1,
    PRIMARY KEY (ticketId, disneyResortName)
    FOREIGN KEY (disneyResortName)
        REFERENCES DisneyResortAddress(disneyResortName)
        ON DELETE SET DEFAULT
        ON UPDATE CASCADE,
    FOREIGN KEY (accountId)
        references Account(accountId)
        ON DELETE SET DEFAULT
        ON UPDATE CASCADE
)
```

24.
```
CREATE TABLE OpenStore(
    storeId INT PRIMARY KEY,
    themeParkId INT,
    landId INT,
    FOREIGN KEY (themeParkId)
        REFERENCES LandTheme(themeParkId),
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (landId)
        REFERENCES LandTheme(landId)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
```

25.
```
CREATE TABLE sell(
    storeId INT,
    sku INT,
    PRIMARY KEY (storeId, sku),
    FOREIGN KEY (storeId)
        REFERENCES OpenStore(storeId)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (sku)
        REFERENCES Merchandise(sku)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
)
```

26.
```
CREATE TABLE Merchandise(
    sku INT PRIMARY KEY,
    name VARCHAR UNIQUE NOT NULL,
    price FLOAT NOT NULL,
)
```

Insert Statements

1. DisneyResortAddress(disneyResortName: varchar, address: varchar)
    a. INSERT INTO DisneyResortAddress VALUES (‘Disneyland Resort’, ‘1313 Disneyland Dr’)
    b. INSERT INTO DisneyResortAddress VALUES (‘Walt Disney World’, ‘1375 East Buena Vista Drive’)
    c. INSERT INTO DisneyResortAddress VALUES (‘Tokyo Disney Resort’, ‘1-1 Maihama’)
    d. INSERT INTO DisneyResortAddress VALUES (‘Disneyland Paris’, ‘Bd de Parc’)
    e. INSERT INTO DisneyResortAddress VALUES (‘Hong Kong Disneyland Resort’, ‘Park Promenade’)
    f. INSERT INTO DisneyResortAddress VALUES (‘Shanghai Disney Resort’, ‘No. 310 Huang Zhao Road’)

2. DisneyResortPostal(disneyResortName: varchar, postalCode: varchar)
    a. INSERT INTO DisneyResortPostal VALUES (‘Disneyland Resort’, ‘92802’)
    b. INSERT INTO DisneyResortPostal VALUES (‘Walt Disney World’, ‘98264’)
    c. INSERT INTO DisneyResortPostal VALUES (‘Tokyo Disney Resort’, ‘279-0031’)
    d. INSERT INTO DisneyResortPostal VALUES (‘Disneyland Paris’, ‘77700’)
    e. INSERT INTO DisneyResortPostal VALUES (‘Hong Kong Disneyland Resort’, ‘HKG’)
    f. INSERT INTO DisneyResortPostal VALUES (‘Shanghai Disney Resort’, ‘10011005’)

3. AddressCityCountry(disneyResortName: varchar, address: varchar, city: varchar, country: varchar)
    a. INSERT INTO AddressCityCountry VALUES (‘Disneyland Resort’, ‘1313 Disneyland Dr’, ‘Anaheim’, ‘USA’)
    b. INSERT INTO AddressCityCountry VALUES (‘Walt Disney World’, ‘1375 East Buena Vista Drive’, ‘Bay Lake’, ‘USA’)
    c. INSERT INTO AddressCityCountry VALUES (‘Tokyo Disney Resort’, ‘1-1 Maihama’, Chiba, ‘Japan’)

d. INSERT INTO AddressCityCountry VALUES ('Disneyland Paris', 'Bd de Parc', 'Coupvray', 'France')
e. INSERT INTO AddressCityCountry VALUES ('Hong Kong Disneyland Resort', 'Park Promenade', 'Hong Kong', 'Hong Kong SAR')
f. INSERT INTO AddressCityCountry VALUES ('Shanghai Disney Resort', 'No. 310 Huang Zhao Road', 'Shanghai', 'China')

4. PostalCodeCity(postalCode: varchar, city: varchar)
   a. INSERT INTO PostalCodeCity VALUES ('92802', 'Anaheim')
   b. INSERT INTO PostalCodeCity VALUES ('98264', 'Bay Lake')
   c. INSERT INTO PostalCodeCity VALUES ('279-0031', 'Chiba')
   d. INSERT INTO PostalCodeCity VALUES ('77700', 'Coupvray')
   e. INSERT INTO PostalCodeCity VALUES ('HKG', 'Hong Kong')
   f. INSERT INTO PostalCodeCity VALUES ('10011005', 'Shanghai')

5. PostalCodeCountry(postalCode: varchar, country: varchar)
   a. INSERT INTO PostalCodeCountry VALUES ('92802', 'USA')
   b. INSERT INTO PostalCodeCountry VALUES ('98264', 'USA')
   c. INSERT INTO PostalCodeCountry VALUES ('279-0031', 'Japan')
   d. INSERT INTO PostalCodeCountry VALUES ('77700', 'France')
   e. INSERT INTO PostalCodeCountry VALUES ('HKG', 'Hong Kong SAR')
   f. INSERT INTO PostalCodeCountry VALUES ('10011005', 'China')

6. ComprisingThemePark(themeParkId: integer, name: varchar, disneyResortName: varchar)
   a. INSERT INTO ComprisingThemePark VALUES (1, 'Disneyland Park', 'Disneyland Resort')
   b. INSERT INTO ComprisingThemePark VALUES (2, 'Disney California Adventure Park', 'Disneyland Resort')
   c. INSERT INTO ComprisingThemePark VALUES (3, 'Magic Kingdom Park', 'Walt Disney World Resort')
   d. INSERT INTO ComprisingThemePark VALUES (4, 'Epcot', 'Walt Disney World Resort')
   e. INSERT INTO ComprisingThemePark VALUES (5, 'Disney's Hollywood Studios', 'Walt Disney World Resort')
   f. INSERT INTO ComprisingThemePark VALUES (6, 'Disney's Animal Kingdom Theme Park', 'Walt Disney World Resort')

g. INSERT INTO ComprisingThemePark VALUES (7, 'Tokyo Disneyland', 'Tokyo Disney Resort')
h. INSERT INTO ComprisingThemePark VALUES (8, 'Tokyo DisneySea', 'Tokyo Disney Resort')
i. INSERT INTO ComprisingThemePark VALUES (9, 'Disneyland Park (Paris)', 'Disneyland Paris')
j. INSERT INTO ComprisingThemePark VALUES (10, 'Walt Disney Studios Park', 'Disneyland Paris')
k. INSERT INTO ComprisingThemePark VALUES (11, 'Hong Kong Disneyland', 'Hong Kong Disneyland Resort')
l. INSERT INTO ComprisingThemePark VALUES (12, 'Shanghai Disneyland Park', 'Shanghai Disneyland Resort')

7. LandTheme(themeParkId: integer, landId: integer, theme: varchar)
   a. INSERT INTO LandTheme VALUES (1, 1, 'American small towns during the early 20th Century')
   b. INSERT INTO LandTheme VALUES (1, 2, 'Remote jungles of Asia, Africa, South America, Oceania, the Caribbean Islands and the Middle East')
   c. INSERT INTO LandTheme VALUES (1, 3, '19th Century American Frontier, American History and North America')
   d. INSERT INTO LandTheme VALUES (1, 4, 'Disney's animated fairy tale films, Fantasy, the towns and villages of Europe')
   e. INSERT INTO LandTheme VALUES (1, 5, 'Future, technology, outer space, discovery and science fiction')
   f. INSERT INTO LandTheme VALUES (1, 6, '19th Century New Orleans')
   g. INSERT INTO LandTheme VALUES (1, 7, 'Land of bears and other animals')
   h. INSERT INTO LandTheme VALUES (1, 8, 'Mickey Mouse universe')
   i. INSERT INTO LandTheme VALUES (1, 9, 'Star Wars')
   j. INSERT INTO LandTheme VALUES (2, 10, '20th Century Art Deco/Mission street')
   k. INSERT INTO LandTheme VALUES (2, 11, 'Pixar/A Victorian era seaside amusement park')
   l. INSERT INTO LandTheme VALUES (2, 12, 'A Victorian era seaside amusement park')

m. INSERT INTO LandTheme VALUES (2, 13, '1950s National Recreation Area')
n. INSERT INTO LandTheme VALUES (2, 14, '1930s Hollywood')
o. INSERT INTO LandTheme VALUES (2, 15, 'Cars, Radiator Springs')
p. INSERT INTO LandTheme VALUES (2, 16, 'Marvel Cinematic Universe')
q. INSERT INTO LandTheme VALUES (3, 5, 'Future, technology, outer space, discovery and science fiction')

8. LandYearOpen(themeParkId: integer, landId: integer, yearOpen: integer)
    a. INSERT INTO LandYearOpen VALUES (1, 1, 1955)
    b. INSERT INTO LandYearOpen VALUES (1, 2, 1955)
    c. INSERT INTO LandYearOpen VALUES (1, 3, 1955)
    d. INSERT INTO LandYearOpen VALUES (1, 4, 1955)
    e. INSERT INTO LandYearOpen VALUES (1, 5, 1955)
    f. INSERT INTO LandYearOpen VALUES (1, 6, 1966)
    g. INSERT INTO LandYearOpen VALUES (1, 7, 1972)
    h. INSERT INTO LandYearOpen VALUES (1, 8, 1993)
    i. INSERT INTO LandYearOpen VALUES (1, 9, 2019)
    j. INSERT INTO LandYearOpen VALUES (2, 10, 2001)
    k. INSERT INTO LandYearOpen VALUES (2, 11, 2001)
    l. INSERT INTO LandYearOpen VALUES (2, 12, 2001)
    m. INSERT INTO LandYearOpen VALUES (2, 13, 2001)
    n. INSERT INTO LandYearOpen VALUES (2, 14, 2001)
    o. INSERT INTO LandYearOpen VALUES (2, 15, 2012)
    p. INSERT INTO LandYearOpen VALUES (2, 16, 2021)
    q. INSERT INTO LandYearOpen VALUES (3, 5, 1971)

9. ThemeParkThemeName(themeParkId: integer, theme: varchar, name: varchar)
    a. INSERT INTO ThemeParkThemeName VALUES (1, 'American small towns during the early 20th Century', 'Main Street, U.S.A.')
    b. INSERT INTO ThemeParkThemeName VALUES (1, 'Remote jungles of Asia, Africa, South America, Oceania, the Caribbean Islands and the Middle East', 'Adventureland')
    c. INSERT INTO ThemeParkThemeName VALUES (1, '19th Century American Frontier, American History and North America', 'Frontierland')

d. INSERT INTO ThemeParkThemeName VALUES (1, ‘Disney's animated fairy tale films, Fantasy, the towns and villages of Europe’, ‘Fantasyland’)
e. INSERT INTO ThemeParkThemeName VALUES (1, ‘Future, technology, outer space, discovery and science fiction’, ‘Tomorrowland’)
f. INSERT INTO ThemeParkThemeName VALUES (1, ‘19th Century New Orleans’, ‘New Orleans Square’)
g. INSERT INTO ThemeParkThemeName VALUES (1, ‘Land of bears and other animals’, ‘Critter Country’)
h. INSERT INTO ThemeParkThemeName VALUES (1, ‘Mickey Mouse universe’, ‘Mickey’s Toontown’)
i. INSERT INTO ThemeParkThemeName VALUES (1, ‘Star Wars’, ‘Star Wars: Galaxy’s Edge’)
j. INSERT INTO ThemeParkThemeName VALUES (2, ‘20th Century Art Deco/Mission street’, ‘Buena Vista Street’)
k. INSERT INTO ThemeParkThemeName VALUES (2, ‘Pixar/A Victorian era seaside amusement park’, ‘Pixar Pier’)
l. INSERT INTO ThemeParkThemeName VALUES (2, ‘A Victorian era seaside amusement park’, ‘Paradise Gardens Park’)
m. INSERT INTO ThemeParkThemeName VALUES (2, ‘1950s National Recreation Area’, ‘Grizzly Peak’)
n. INSERT INTO ThemeParkThemeName VALUES (2, ‘1930s Hollywood’, ‘Hollywood Land’)
o. INSERT INTO ThemeParkThemeName VALUES (2, ‘Marvel Cinematic Universe’, ‘Avengers Campus’)
p. INSERT INTO ThemeParkThemeName VALUES (2, ‘Cars, Radiator Springs’, ‘Cars Land’)
q. INSERT INTO ThemeParkThemeName VALUES (3, ‘Future, technology, outer space, discovery and science fiction’, ‘Tomorrowland’)
r. INSERT INTO ThemeParkThemeName VALUES (12, ‘Remote jungles of Asia, Africa, South America, Oceania, the Caribbean Islands and the Middle East’, ‘Adventure Isle’)

10. IsPartOf(themeParkId: integer, landId: integer, attractionId: integer)
    a. INSERT INTO IsPartOf VALUES (1, 5, 1)
    b. INSERT INTO IsPartOf VALUES (1, 5, 5)
    c. INSERT INTO IsPartOf VALUES (1, 3, 3)
    d. INSERT INTO IsPartOf VALUES (1, 6, 4)

e. INSERT INTO IsPartOf VALUES (2, 11, 2)
f. INSERT INTO IsPartOf VALUES (3, 5, 1)

11. Attraction(attractionId: integer, name: varchar)
    a. INSERT INTO Attraction VALUES (1, ‘Space Mountain’)
    b. INSERT INTO Attraction VALUES (2, ‘Incredicoaster’)
    c. INSERT INTO Attraction VALUES (3, ‘Big Thunder Mountain Railroad’)
    d. INSERT INTO Attraction VALUES (4, ‘Pirates of the Caribbean’)
    e. INSERT INTO Attraction VALUES (5, ‘Buzz Lightyear Astro Blasters’)
    f. INSERT INTO Attraction VALUES (6, ‘Walt Disney's Enchanted Tiki Room’)
    g. INSERT INTO Attraction VALUES (7, ‘“Believe...In Holiday Magic" Fireworks Spectacular’)
    h. INSERT INTO Attraction VALUES (8, ‘A Christmas Fantasy Parade’)
    i. INSERT INTO Attraction VALUES (9, ‘Disney Junior Dance Party!’)
    j. INSERT INTO Attraction VALUES (10, ‘Fireworks at Disneyland Park’)

12. RideMinimumHeight(attractionId: integer, minimumHeight: integer)
    a. INSERT INTO RideMinimumHeight VALUES (1, 102)
    b. INSERT INTO RideMinimumHeight VALUES (2, 122)
    c. INSERT INTO RideMinimumHeight VALUES (3, 102)
    d. INSERT INTO RideMinimumHeight VALUES (4, 0)
    e. INSERT INTO RideMinimumHeight VALUES (5, 0)

13. RideAvgWaitTime(attractionId: integer, avgWaitTime: integer)
    a. INSERT INTO RideAvgWaitTime VALUES (1, 50)
    b. INSERT INTO RideAvgWaitTime VALUES (2, 50)
    c. INSERT INTO RideAvgWaitTime VALUES (3, 35)
    d. INSERT INTO RideAvgWaitTime VALUES (4, 40)
    e. INSERT INTO RideAvgWaitTime VALUES (5, 25)

14. MinimumHeightRideType(minimumHeight: integer, rideType: varchar)
    a. INSERT INTO MinimumHeightRideType VALUES (122, 'Big Drops')
    b. INSERT INTO MinimumHeightRideType VALUES (117, 'Medium Drops')
    c. INSERT INTO MinimumHeightRideType VALUES (107, 'Small Drops')
    d. INSERT INTO MinimumHeightRideType VALUES (102, 'Mini Drops')
    e. INSERT INTO MinimumHeightRideType VALUES (0, 'All Heights')

15. Event(attractionId: integer, eventType: varchar)
    a. INSERT INTO Event VALUES (6, 'Show')
    b. INSERT INTO Event VALUES (7, 'Fireworks')
    c. INSERT INTO Event VALUES (8, 'Parade')
    d. INSERT INTO Event VALUES (9, 'Show')
    e. INSERT INTO Event VALUES (10, 'Fireworks')

16. ShowtimeEvent(attractionId: integer, startTime: integer, endTime: integer)
    a. INSERT INTO Event VALUES (6, 0800, 0815)
    b. INSERT INTO Event VALUES (6, 0930, 0945)
    c. INSERT INTO Event VALUES (7, 2100, 2115)
    d. INSERT INTO Event VALUES (8, 1300, 1340)
    e. INSERT INTO Event VALUES (9, 1015, 1045)
    f. INSERT INTO Event VALUES (9, 1115, 1145)
    g. INSERT INTO Event VALUES (10, 2030, 2040)

17. ShowtimeDuration(startTime: integer, endTime: integer, duration: integer)
    a. INSERT INTO ShowtimeDuration VALUES (0800, 0815, 15)
    b. INSERT INTO ShowtimeDuration VALUES (09030, 0945, 15)
    c. INSERT INTO ShowtimeDuration VALUES (2100, 2115, 15)
    d. INSERT INTO ShowtimeDuration VALUES (1300, 1340, 40)
    e. INSERT INTO ShowtimeDuration VALUES (1015, 1045, 30)
    f. INSERT INTO ShowtimeDuration VALUES (1115, 1145, 30)
    g. INSERT INTO ShowtimeDuration VALUES (2030, 2040, 10)

18. FeatureRestaurant(restaurantId: integer, name: varchar, maxOccupancy: integer, themeParkId: integer, landId: integer)
    a. INSERT INTO FeatureRestaurant VALUES (1, 'Hungry Bear Restaurant', 150, 1, 7)
    b. INSERT INTO FeatureRestaurant VALUES (2, 'Carthay Circle Restaurant', 200, 2, 10)

c. INSERT INTO FeatureRestaurant VALUES (3, ‘Plaza Inn’, 400, 1, 1)
d. INSERT INTO FeatureRestaurant VALUES (4, ‘Blue Bayou Restaurant’, 300, 1, 6)
e. INSERT INTO FeatureRestaurant VALUES (5, ‘Paradise Garden Grill’, 130, 2, 12)

19. Serve(restaurantId: integer, foodName: varchar)
    a. INSERT INTO Serve VALUES (1, ‘Honey-Spiced Chicken Sandwich’)
    b. INSERT INTO Serve VALUES (1, ‘Classic Cheeseburger’)
    c. INSERT INTO Serve VALUES (1, BBQ Chicken Salad’)
    d. INSERT INTO Serve VALUES (1, ‘Pumpkin Churro Funnel Cake’)
    e. INSERT INTO Serve VALUES (1, ‘French Fries’)

20. Food(foodName: varchar, price: float)
    a. INSERT INTO Food VALUES (‘Honey-Spiced Chicken Sandwich’, 12.99)
    b. INSERT INTO Food VALUES (‘Classic Cheeseburger’, 12.79)
    c. INSERT INTO Food VALUES (‘BBQ Chicken Salad’, 12.49)
    d. INSERT INTO Food VALUES (‘Pumpkin Churro Funnel Cake’, 11.99)
    e. INSERT INTO Food VALUES (‘French Fries’, 4.49)

21. Account(accountId: integer, name: varchar, email: varchar, birthDate: integer)
    a. INSERT INTO Account VALUES (001, ‘Celine’, ‘celine@ubc.ca’, 20031114)
    b. INSERT INTO Account VALUES (002, ‘Brandon’, ‘brandon@ubc.ca’, 20010629)
    c. INSERT INTO Account VALUES (003, ‘Josh’, ‘josh@ubc.ca’, 20010424)
    d. INSERT INTO Account VALUES (004, ‘Melissa’, ‘melissa@ubc.ca’, 20010925)
    e. INSERT INTO Account VALUES (005, ‘Rachael’, ‘rachael@ubc.ca’, 19750101)

22. Reserve(accountId: integer, restaurantId: integer, time: integer)
    a. INSERT INTO Reserve VALUES (001, 1, 1200)
    b. INSERT INTO Reserve VALUES (002, 5, 2015)
    c. INSERT INTO Reserve VALUES (003, 3, 1400)
    d. INSERT INTO Reserve VALUES (001, 4, 1700)
    e. INSERT INTO Reserve VALUES (001, 5, 1800)

23. TicketAtDisneyResortOwnedByAccount(ticketId: integer, disneyResortName: varchar, entryDate: integer, accountId: integer)
    a. INSERT INTO TicketAtDisneyResortOwnedByAccount VALUES (111, 'Shanghai Disney Resort', 20231114, 001)
    b. INSERT INTO TicketAtDisneyResortOwnedByAccount VALUES (112, 'Shanghai Disney Resort', 20231115, 001)
    c. INSERT INTO TicketAtDisneyResortOwnedByAccount VALUES (113, 'Disneyland Paris', 20231020, 003)
    d. INSERT INTO TicketAtDisneyResortOwnedByAccount VALUES (114, 'Disneyland Resort', 20191221, 002)
    e. INSERT INTO TicketAtDisneyResortOwnedByAccount VALUES (115, 'Shanghai Disney Resort', 20241021, 003)

24. OpenStore(storeId: integer, name: varchar, themeParkId: integer, landId: integer)
    a. INSERT INTO OpenStore VALUES (01, 'World of Disney®', NULL, NULL)
    b. INSERT INTO OpenStore VALUES (02, 'Emporium', 1, 1)
    c. INSERT INTO OpenStore VALUES (03, 'Little Green Men Store Command', 1, 5)
    d. INSERT INTO OpenStore VALUES (04, 'The LEGO® Store', NULL, NULL)
    e. INSERT INTO OpenStore VALUES (05, 'Pioneer Mercantile', 1, 3)

25. Sell(storeId: integer, sku: integer)
    a. INSERT INTO OpenStore VALUES (01, 55555)
    b. INSERT INTO OpenStore VALUES (01, 66666)
    c. INSERT INTO OpenStore VALUES (01, 77777)
    d. INSERT INTO OpenStore VALUES (01, 88888)
    e. INSERT INTO OpenStore VALUES (01, 99999)

26. Merchandise(sku: integer, name: varchar, price: float)
    a. INSERT INTO OpenStore VALUES (55555, 'Winnie-the-Pooh plush large', 65)
    b. INSERT INTO OpenStore VALUES (66666, 'Piglet plush large', 60)
    c. INSERT INTO OpenStore VALUES (77777, 'Tigger plush large', 60)
    d. INSERT INTO OpenStore VALUES (88888, 'Rabbit plush large', 60)
    e. INSERT INTO OpenStore VALUES (99999, 'Eeyore plush large', 60)