Project #1

Railway Reservation System

CSE 3330

By:

Muhammad Muawiz Farooqi

Faith Gutierrez

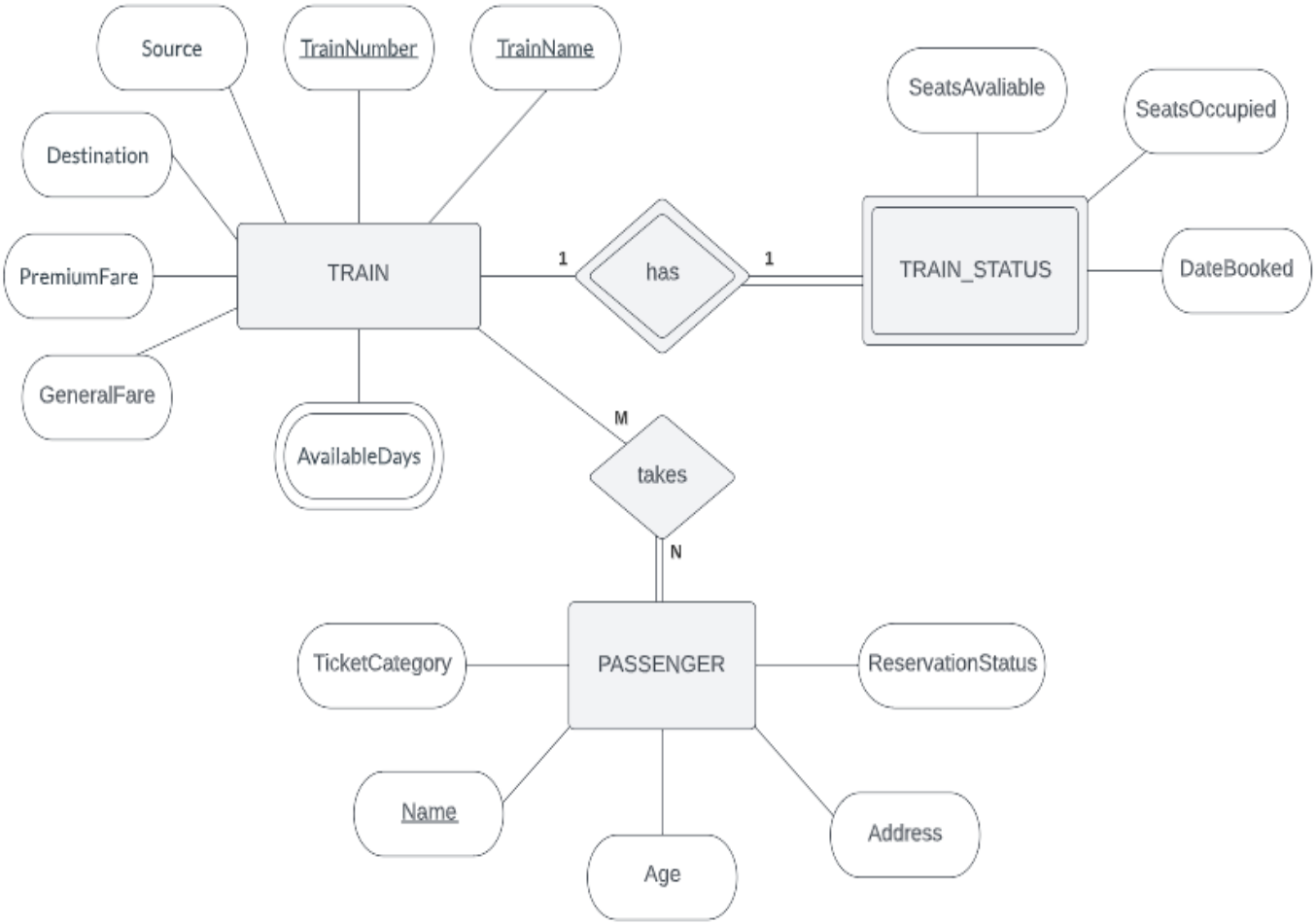
Tahera Fatima

Tools used for the Project (Readme File)

- Lucid Chart to create the ER Diagram
- MySQL to create the Create Statements
- Sqlite3 to write and execute the queries (screenshots shown below)
- Modified .csv files used for the project attached below:

Train.csv Passenger-1.csv booked-1.csv Train_status.csv

ER Diagram



SQL Create Statements

```sql
CREATE TABLE TRAIN (
    train_number INTEGER NOT NULL,
    train_name varchar(20) NOT NULL,
    premium_fare INTEGER NOT NULL,
    general_fare INTEGER NOT NULL,
    source varchar(20) NOT NULL,
    destination varchar(20) NOT NULL,
    available_days varchar(52) NOT NULL,
    PRIMARY KEY (train_number, available_days)
);

CREATE TABLE PASSENGER (
    first_name VARCHAR(15) NOT NULL,
    last_name VARCHAR(15) NOT NULL,
    address VARCHAR(30),
    city VARCHAR(15),
    county VARCHAR(15),
    phone CHAR(12),
    SSN CHAR(9) NOT NULL,
    bdate DATE,
    PRIMARY KEY (SSN)
);

```

```sql
CREATE TABLE BOOKED (
    passenger_SSN CHAR(9) NOT NULL,
    train_number INTEGER NOT NULL,
    ticket_type VARCHAR(9) NOT NULL,
    reservation_status VARCHAR(9) NOT NULL,
    PRIMARY KEY(passenger_SSN, train_number),
    FOREIGN KEY (passenger_SSN) REFERENCES PASSENGER(SSN),
    FOREIGN KEY (train_number) REFERENCES TRAIN(train_number)
);

CREATE TABLE TRAIN_STATUS(
    train_date DATE,
    train_name VARCHAR(20) NOT NULL,
    premiumseats_available INTEGER NOT NULL,
    genseats_available INTEGER NOT NULL,
    premiumseats_occupied INTEGER NOT NULL,
    genseats_occupied INTEGER NOT NULL,
    PRIMARY KEY (train_name)
);

```

Method to Load Data into Tables

```
sqlite> .mode csv
sqlite> .import booked-1.csv BOOKED
sqlite> .import Passenger-1.csv PASSENGER
Error: cannot open "Passenger-1.csv"
sqlite> .import Passenger-1.csv PASSENGER
sqlite> .import Train_status.csv TRAIN_STATUS
sqlite> INSERT INTO TRAIN VALUES (1, 'Orient Express',   800,  600,  'Paris',     'Istanbul', 'Monday');
sqlite> INSERT INTO TRAIN VALUES (1, 'Orient Express',   800,  600,  'Paris',     'Istanbul', 'Tuesday');
sqlite> INSERT INTO TRAIN VALUES (1, 'Orient Express',   800,  600,  'Paris',     'Istanbul', 'Wednesday');
sqlite> INSERT INTO TRAIN VALUES (1, 'Orient Express',   800,  600,  'Paris',     'Istanbul', 'Thursday');
sqlite> INSERT INTO TRAIN VALUES (1, 'Orient Express',   800,  600,  'Paris',     'Istanbul', 'Friday');
sqlite>
sqlite> INSERT INTO TRAIN VALUES (2, 'Flying Scotsman',  4000, 3500, 'Edinburgh', 'London',   'Friday');
sqlite> INSERT INTO TRAIN VALUES (2, 'Flying Scotsman',  4000, 3500, 'Edinburgh', 'London',   'Saturday');
sqlite> INSERT INTO TRAIN VALUES (2, 'Flying Scotsman',  4000, 3500, 'Edinburgh', 'London',   'Sunday');
sqlite>
sqlite> INSERT INTO TRAIN VALUES (3, 'Golden Arrow' ,    980,  860,  'Victoria',  'Dover',    'Monday');
sqlite> INSERT INTO TRAIN VALUES (3, 'Golden Arrow' ,    980,  860,  'Victoria',  'Dover',    'Tuesday');
sqlite> INSERT INTO TRAIN VALUES (3, 'Golden Arrow' ,    980,  860,  'Victoria',  'Dover',    'Wednesday');
sqlite>
sqlite> INSERT INTO TRAIN VALUES (4, 'Golden Chariot',   4300, 3800, 'Bangalore', 'Goa',      'Saturday');
sqlite> INSERT INTO TRAIN VALUES (4, 'Golden Chariot',   4300, 3800, 'Bangalore', 'Goa',      'Sunday');
sqlite>
sqlite> INSERT INTO TRAIN VALUES (5, 'Maharaja Express', 5980, 4510, 'Delhi',     'Mumbai',   'Wednesday');
sqlite> INSERT INTO TRAIN VALUES (5, 'Maharaja Express', 5980, 4510, 'Delhi',     'Mumbai',   'Thursday');
sqlite> INSERT INTO TRAIN VALUES (5, 'Maharaja Express', 5980, 4510, 'Delhi',     'Mumbai',   'Friday');
sqlite> DELETE FROM BOOKED WHERE ticket_type = 'Ticket_Type';
sqlite> DELETE FROM PASSENGER WHERE last_name = 'last_name';
sqlite> DELTE FROM TRAIN_STATUS WHERE train_date = 'TrainDate';
Parse error: near "DELTE": syntax error
  DELTE FROM TRAIN_STATUS WHERE train_date = 'TrainDate';
  ^--- error here
sqlite> DELETE FROM TRAIN_STATUS WHERE train_date = 'TrainDate';
```

We used .import to load the data into TRAIN_STATUS, PASSENGER, and BOOKED. Then we used INSERT statements to load the data into TRAIN. Lastly, we deleted the first row from TRAIN_STATUS, PASSENGER, and BOOKED to remove the column labels that were imported from the .csv files.

SQL Queries

2. Input a passenger's last name and first name and retrieve all trains they are booked on.

- **Example: First name = Art, Last name = Venere**

```
sqlite> SELECT DISTINCT t.train_name
   ...> FROM PASSENGER AS p, BOOKED AS b, TRAIN AS t
   ...> WHERE p.ssn = b.passenger_ssn AND p.first_name = 'Art' AND p.last_name = 'Venere'
   ...>       AND b.train_number = t.train_number AND b.reservation_status = 'Booked';
Golden Chariot
Flying Scotsman
```

3. Input the Day and list the passengers travelling on that day with confirmed tickets.

- **Example: Day = Saturday**

```
sqlite> SELECT DISTINCT p.first_name, p.last_name
   ...> FROM PASSENGER AS p, TRAIN AS t, BOOKED AS b
   ...> WHERE t.available_days = 'Saturday' AND t.train_number = b.train_number AND b.reservation_status = 'Booked'
   ...>       AND b.passenger_ssn = p.ssn;
first_name  last_name
----------  ---------
Josephine   Darakjy
Art         Venere
Fletcher    Flosi
Sage        Wieser
Kris        Marrier
Gladys      Rim
Yuki        Whobrey
Minna       Amigon
Abel        Maclead
Kiley       Caldarera
Graciela    Ruta
Cammy       Albares
Mattie      Poquette
```

4. User input the age of the passenger (50 to 60) and display the train information (Train Number, Train Name, Source and Destination) and passenger information (Name, Address, Category, ticket status) of passengers who are between the ages of 50 to 60.

- **Example: All passengers between the ages of 50 and 60 (inclusive)**

```
sqlite> SELECT DISTINCT p.first_name, p.last_name, p.address, b.ticket_type AS category, b.reservation_status AS ticket_status,
   ...>                 t.train_number, t.train_name, t.source, t.destination
   ...> FROM TRAIN AS t, PASSENGER AS p, BOOKED AS b
   ...> WHERE  ('2023-3-10') - bdate <= 60 AND ('2023-3-10') - bdate >= 50 AND p.ssn = b.passenger_ssn
   ...>       AND b.train_number = t.train_number
   ...> ORDER BY p.first_name, t.train_number;
first_name  last_name  address            category  ticket_status  train_number  train_name    source    destination
----------  ---------  -----------------  --------  -------------  ------------  ------------  --------  -----------
James       Butt       6649 N Blue Gum St  Premium   Booked         3             Golden Arrow  Victoria  Dover
```

5. List train name, day, and number of passengers on that train

```
sqlite> SELECT train_name, available_days,
   ...> SUM(total_seats_occupied) AS total_seats_occupied
   ...>  FROM (
(x1...> SELECT TRAIN.train_name, available_days, COUNT(*) AS total_seats_occupied
(x1...> FROM TRAIN, TRAIN_STATUS
(x1...> GROUP BY TRAIN.train_name, available_days
(x1...> ) AS subquery
   ...>  GROUP BY train_name;
Flying Scotsman|Friday|12
Golden Arrow|Monday|12
Golden Chariot|Saturday|8
Maharaja Express|Friday|12
Orient Express|Friday|20
```

6. Enter a train name and retrieve all the passengers with confirmed status traveling in that train.

- **Example: TRAIN.train_name= Flying Scotsman**

```
sqlite> SELECT DISTINCT PASSENGER.first_name, PASSENGER.last_name, TRAIN.train_name, BOOKED.reservation_status
   ...>  FROM PASSENGER
   ...> INNER JOIN BOOKED ON PASSENGER.SSN = BOOKED.passenger_SSN
   ...> INNER JOIN TRAIN ON BOOKED.train_number = TRAIN.train_number
   ...> WHERE TRAIN.train_name ='Flying Scotsman'
   ...> AND BOOKED.reservation_status ='Booked';
Josephine|Darakjy|Flying Scotsman|Booked
Art|Venere|Flying Scotsman|Booked
Fletcher|Flosi|Flying Scotsman|Booked
Sage|Wieser|Flying Scotsman|Booked
Kris|Marrier|Flying Scotsman|Booked
```

7. List passengers that are waitlisted including the name of the train.

```
sqlite>  SELECT DISTINCT BOOKED.passenger_SSN, TRAIN.train_name,
   ...>  PASSENGER.first_name, PASSENGER.last_name, BOOKED.reservation_status
   ...>  FROM BOOKED
   ...>  INNER JOIN TRAIN ON BOOKED.train_number = TRAIN.train_number
   ...>  INNER JOIN PASSENGER ON BOOKED.passenger_SSN = PASSENGER.SSN
   ...>  WHERE BOOKED.reservation_status = 'WaitL';
256558303|Golden Arrow|Minna|Amigon|WaitL
277292710|Golden Arrow|Graciela|Ruta|WaitL
284965676|Golden Arrow|Kiley|Caldarera|WaitL
290123298|Golden Arrow|Meaghan|Garufi|WaitL
302548590|Flying Scotsman|Abel|Maclead|WaitL
331160133|Golden Arrow|Cammy|Albares|WaitL
331293204|Golden Arrow|Mattie|Poquette|WaitL
```

8. List passengers that have '605' phone area code in descending order.

```
sqlite> SELECT first_name, last_name
   ...> FROM PASSENGER
   ...> WHERE phone LIKE '605%'
   ...> ORDER BY last_name, first_name DESC;
first_name  last_name
----------  ---------
Mattie      Poquette
Art         Venere
Sage        Wieser
```

9. List name of passengers that are traveling on Thursdays in ascending order.

```
sqlite> SELECT P.first_name, P.last_name FROM PASSENGER P, TRAIN T, BOOKED B WHERE P.SSN =
 B.passenger_SSN AND B.train_number = T.train_number AND B.reservation_status = 'Booked' A
ND (T.available_days = 'Thursday' OR T.available_days = 'Weekdays') ORDER BY P.last_name,
P.first_name ASC;
sqlite>
```

Contribution List

Muhammad Muawiz Farooqi:

- ER diagram: TRAIN entity, attributes, and relationships.
- Queries: 2-4
- Create Statements: TRAIN
- .csv files: modified date format in PASSENGER, added multivalued attributes in TRAIN.

Faith Gutierrez:

- ER diagram: PASSENGER and TRAIN_STATUS entity and attributes.
- Queries: 8 and 9.
- Create Statements: PASSENGER and BOOKED
- Explain method used to load data into tables.

Tahera Fatima:

- Queries: 5-7
- Create Statements: TRAIN_STATUS
- Tools used for the Project.

HONOR CODE

I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values hard work and honest effort in the pursuit of academic excellence.

I promise that I will submit only work that I personally create or that I contribute to group collaborations, and I will appropriately reference any work from other sources. I will follow the highest standards of integrity and uphold the spirit of the Honor Code.

Muhammad Muawiz Farooqi

Faith Gutierrez

Tahera Fatima