Project 1

CSE 3330-004

2/25/22

Group 8:
Carlos Uresti Sifuentes
Gerald Mumo Kimeu
Shivam Patel

Readme file

The Tools we used were:
Sqlite3

- Through Command Line to create our SQL schema and insert data into those tables.

SQL Browser

- To visualize how the data is stored in each table

CSE VM

- To run SQLite3

Diagram.net

- Used to create the ER diagram

Modification:
Created a Table called Available_on which holds the days in which each train is available and we also created a csv to display days in which each train is available.

CSV Data:

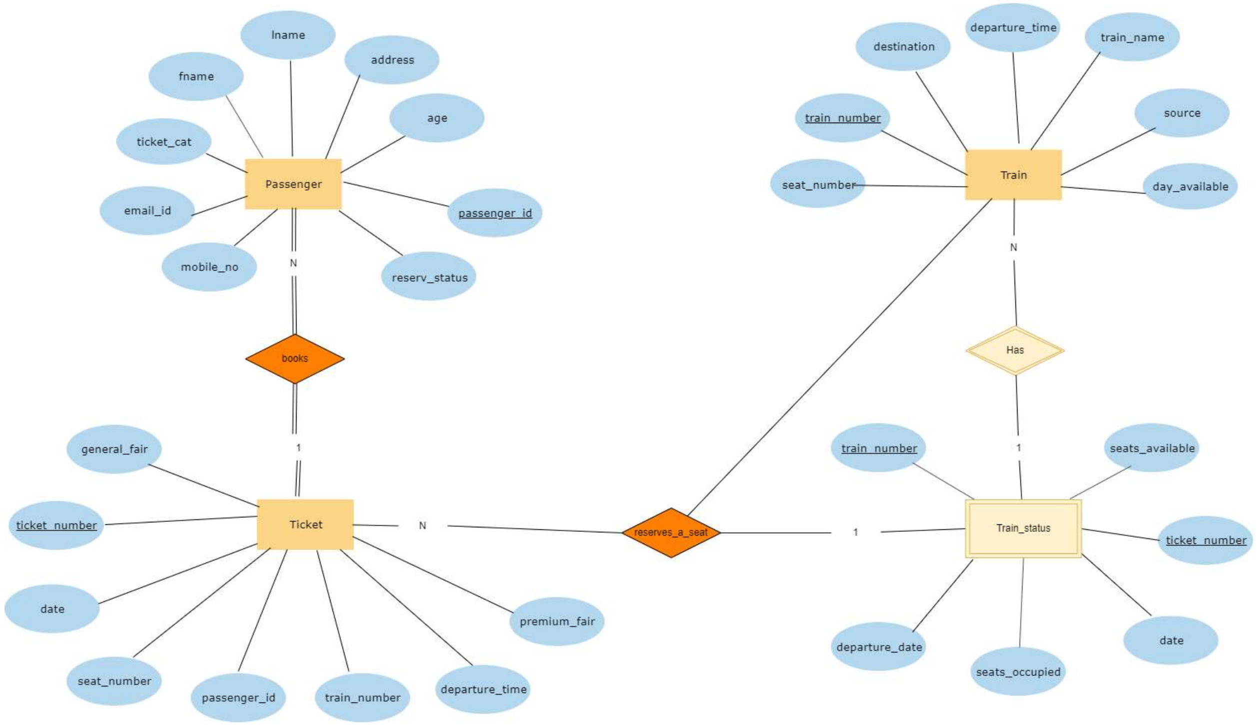
```
Orient Express,1,1,1,1,1,0,0
Flying Scottsman,0,0,0,0,1,1,1
Golden Arrow,1,1,1,0,0,0,0
Golden Chariot,0,0,0,0,0,1,1
Maharaja Express,0,0,1,1,1,0,0
```

Assumptions:

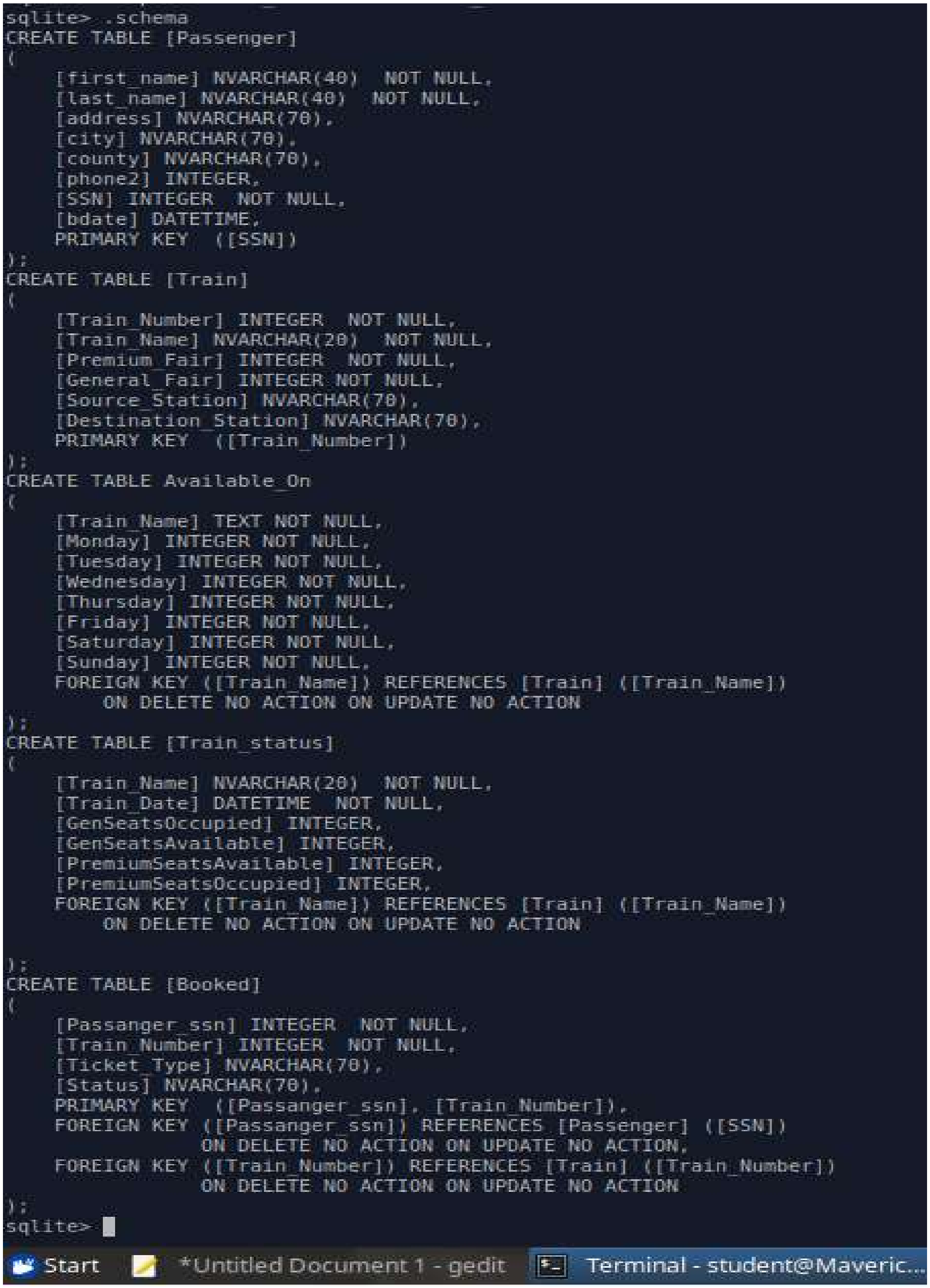
- Only two categories of tickets are available : Premium and General Ticket
- The total number of tickets can be booked in each category (Premium and General)
is 10
- Number of tickets in waiting is 2
- Total Number of trains are 5
- Any stops made by a train before its final destination and their bookings are not considered.

2. ER Diagram

Train



3. Source code of SQL CREATE statements, or screen-shots of how you created the tables



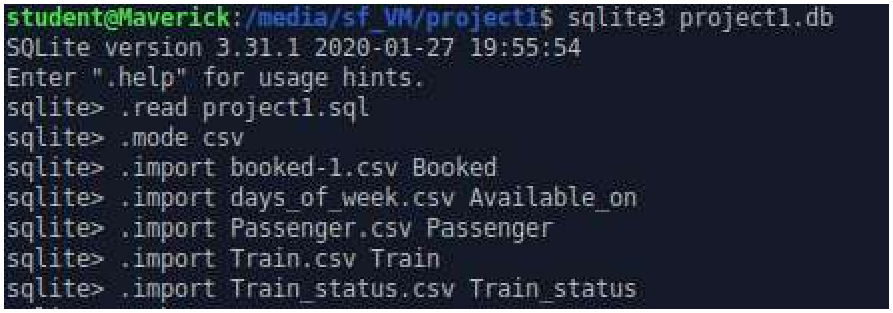
```
sqlite> .schema
CREATE TABLE [Passenger]
(
    [first_name] NVARCHAR(40)  NOT NULL,
    [last_name] NVARCHAR(40)  NOT NULL,
    [address] NVARCHAR(70),
    [city] NVARCHAR(70),
    [county] NVARCHAR(70),
    [phone2] INTEGER,
    [SSN] INTEGER  NOT NULL,
    [bdate] DATETIME,
    PRIMARY KEY  ([SSN])
);
CREATE TABLE [Train]
(
    [Train_Number] INTEGER  NOT NULL,
    [Train_Name] NVARCHAR(20)  NOT NULL,
    [Premium_Fair] INTEGER  NOT NULL,
    [General_Fair] INTEGER NOT NULL,
    [Source_Station] NVARCHAR(70),
    [Destination_Station] NVARCHAR(70),
    PRIMARY KEY  ([Train_Number])
);
CREATE TABLE Available_On
(
    [Train_Name] TEXT NOT NULL,
    [Monday] INTEGER NOT NULL,
    [Tuesday] INTEGER NOT NULL,
    [Wednesday] INTEGER NOT NULL,
    [Thursday] INTEGER NOT NULL,
    [Friday] INTEGER NOT NULL,
    [Saturday] INTEGER NOT NULL,
    [Sunday] INTEGER NOT NULL,
    FOREIGN KEY ([Train_Name]) REFERENCES [Train] ([Train_Name])
        ON DELETE NO ACTION ON UPDATE NO ACTION
);
CREATE TABLE [Train_status]
(
    [Train_Name] NVARCHAR(20)  NOT NULL,
    [Train_Date] DATETIME  NOT NULL,
    [GenSeatsOccupied] INTEGER,
    [GenSeatsAvailable] INTEGER,
    [PremiumSeatsAvailable] INTEGER,
    [PremiumSeatsOccupied] INTEGER,
    FOREIGN KEY ([Train_Name]) REFERENCES [Train] ([Train_Name])
        ON DELETE NO ACTION ON UPDATE NO ACTION

);
CREATE TABLE [Booked]
(
    [Passanger_ssn] INTEGER  NOT NULL,
    [Train_Number] INTEGER  NOT NULL,
    [Ticket_Type] NVARCHAR(70),
    [Status] NVARCHAR(70),
    PRIMARY KEY  ([Passanger_ssn], [Train_Number]),
    FOREIGN KEY ([Passanger_ssn]) REFERENCES [Passenger] ([SSN])
                ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY ([Train_Number]) REFERENCES [Train] ([Train_Number])
                ON DELETE NO ACTION ON UPDATE NO ACTION
);
sqlite> 
```

4. Explain which method you used to load the data into the tables.

We used the .csv files and used the IMPORT command to insert records into particular tables.

EXAMPLE:

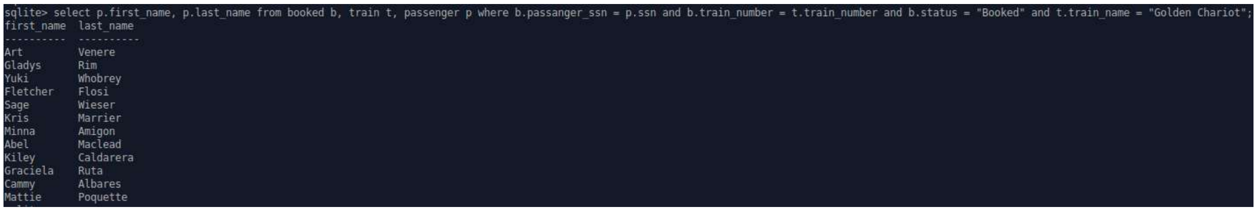


```
student@Maverick:/media/sf_VM/project1$ sqlite3 project1.db
SQLite version 3.31.1 2020-01-27 19:55:54
Enter ".help" for usage hints.
sqlite> .read project1.sql
sqlite> .mode csv
sqlite> .import booked-1.csv Booked
sqlite> .import days_of_week.csv Available_on
sqlite> .import Passenger.csv Passenger
sqlite> .import Train.csv Train
sqlite> .import Train_status.csv Train_status
```

5. Source code of SQL SELECT statements for each query executed, showing the query
result; OR screen shots of your simple Web interface that was used to execute the queries
and show the query results.

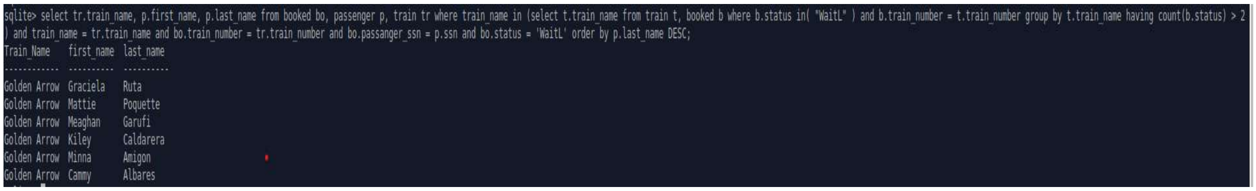
Answers

2. select p.first_name, p.last_name from booked b, train t, passenger p where b.passanger_ssn = p.ssn and b.train_number = t.train_number and b.status = "Booked" and t.train_name = "Golden Chariot";



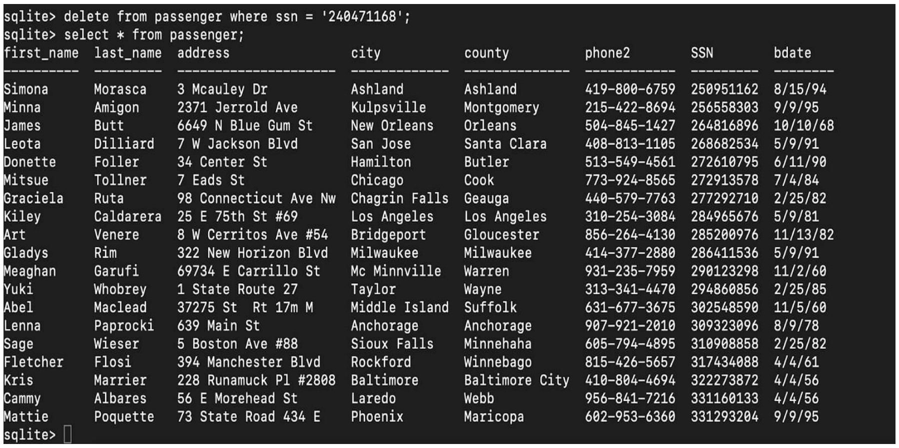
```
sqlite> select p.first_name, p.last_name from booked b, train t, passenger p where b.passanger_ssn = p.ssn and b.train_number = t.train_number and b.status = "Booked" and t.train_name = "Golden Chariot";
first_name  last_name
----------  ----------
Art         Venere
Gladys      Rim
Yuki        Whobrey
Fletcher    Flosi
Sage        Wieser
Kris        Marrier
Minna       Amigon
Abel        Maclead
Kiley       Caldarera
Graciela    Ruta
Cammy       Albares
Mattie      Poquette
```

3. select tr.train_name, p.first_name, p.last_name from booked bo, passenger p, train tr where train_name in (select t.train_name from train t, booked b where b.status in("WaitL") and b.train_number = t.train_number group by t.train_name having count(b.status) > 2) and train_name = tr.train_name and bo.train_number = tr.train_number and bo.passanger_ssn = p.ssn and bo.status = 'WaitL' order by p.last_name DESC;

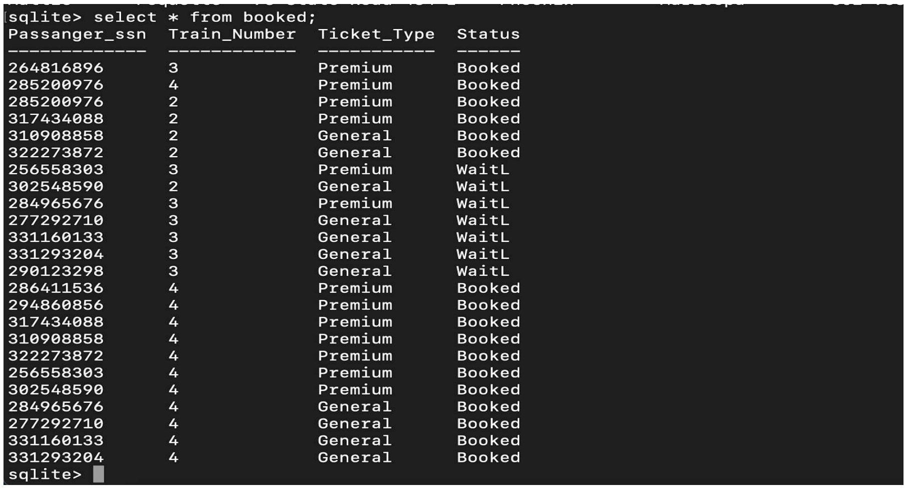


```
sqlite> select tr.train_name, p.first_name, p.last_name from booked bo, passenger p, train tr where train_name in (select t.train_name from train t, booked b where b.status in( "WaitL" ) and b.train_number = t.train_number group by t.train_name having count(b.status) > 2
) and train_name = tr.train_name and bo.train_number = tr.train_number and bo.passanger_ssn = p.ssn and bo.status = 'WaitL' order by p.last_name DESC;
Train_Name   first_name  last_name
-----------  ----------  ---------
Golden Arrow  Graciela   Ruta
Golden Arrow  Mattie     Poquette
Golden Arrow  Meaghan    Garufi
Golden Arrow  Kiley      Caldarera
Golden Arrow  Minna      Amigon
Golden Arrow  Cammy      Albares
```

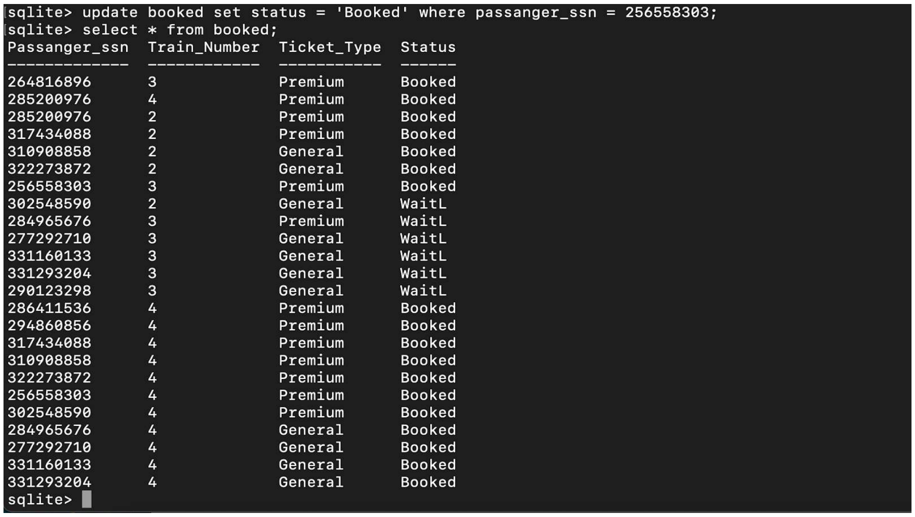
4. delete from passenger where ssn = '240471168';
update booked set status = 'Booked' where passanger_ssn = 256558303;



```
sqlite> delete from passenger where ssn = '240471168';
sqlite> select * from passenger;
first_name  last_name  address                city           county          phone2        SSN        bdate
----------  ---------  ---------------------  -------------  --------------  ------------  ---------  --------
Simona      Morasca    3 Mcauley Dr           Ashland        Ashland         419-800-6759  250951162  8/15/94
Minna       Amigon     2371 Jerrold Ave       Kulpsville     Montgomery      215-422-8694  256558303  9/9/95
James       Butt       6649 N Blue Gum St     New Orleans    Orleans         504-845-1427  264816896  10/10/68
Leota       Dilliard   7 W Jackson Blvd       San Jose       Santa Clara     408-813-1105  268682534  5/9/91
Donette     Foller     34 Center St           Hamilton       Butler          513-549-4561  272610795  6/11/90
Mitsue      Tollner    7 Eads St              Chicago        Cook            773-924-8565  272913578  7/4/84
Graciela    Ruta       98 Connecticut Ave Nw  Chagrin Falls  Geauga          440-579-7763  277292710  2/25/82
Kiley       Caldarera  25 E 75th St #69       Los Angeles    Los Angeles     310-254-3084  284965676  5/9/81
Art         Venere     8 W Cerritos Ave #54   Bridgeport     Gloucester      856-264-4130  285200976  11/13/82
Gladys      Rim        322 New Horizon Blvd   Milwaukee      Milwaukee       414-377-2880  286411536  5/9/91
Meaghan     Garufi     69734 E Carrillo St    Mc Minnville   Warren          931-235-7959  290123298  11/2/60
Yuki        Whobrey    1 State Route 27       Taylor         Wayne           313-341-4470  294860856  2/25/85
Abel        Maclead    37275 St  Rt 17m M     Middle Island  Suffolk         631-677-3675  302548590  11/5/60
Lenna       Paprocki   639 Main St            Anchorage      Anchorage       907-921-2010  309323096  8/9/78
Sage        Wieser     5 Boston Ave #88       Sioux Falls    Minnehaha       605-794-4895  310908858  2/25/82
Fletcher    Flosi      394 Manchester Blvd    Rockford       Winnebago       815-426-5657  317434088  4/4/61
Kris        Marrier    228 Runamuck Pl #2808  Baltimore      Baltimore City  410-804-4694  322273872  4/4/56
Cammy       Albares    56 E Morehead St       Laredo         Webb            956-841-7216  331160133  4/4/56
Mattie      Poquette   73 State Road 434 E    Phoenix        Maricopa        602-953-6360  331293204  9/9/95
sqlite> 
```

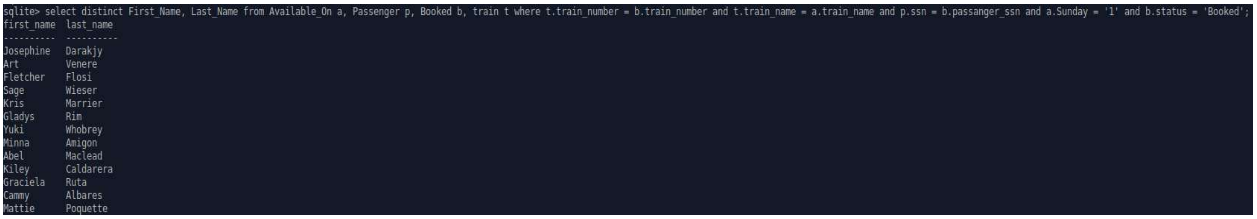


```
sqlite> select * from booked;
Passanger_ssn  Train_Number  Ticket_Type  Status
-------------  ------------  -----------  ------
264816896      3             Premium      Booked
285200976      4             Premium      Booked
285200976      2             Premium      Booked
317434088      2             Premium      Booked
310908858      2             General      Booked
322273872      2             General      Booked
256558303      3             Premium      WaitL
302548590      2             General      WaitL
284965676      3             Premium      WaitL
277292710      3             General      WaitL
331160133      3             General      WaitL
331293204      3             General      WaitL
290123298      3             General      WaitL
286411536      4             Premium      Booked
294860856      4             Premium      Booked
317434088      4             Premium      Booked
310908858      4             Premium      Booked
322273872      4             Premium      Booked
256558303      4             Premium      Booked
302548590      4             Premium      Booked
284965676      4             General      Booked
277292710      4             General      Booked
331160133      4             General      Booked
331293204      4             General      Booked
sqlite> 
```



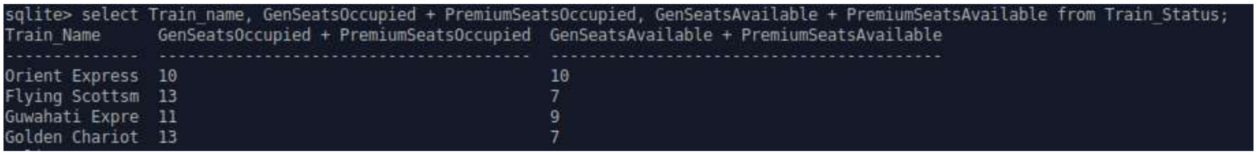
```
sqlite> update booked set status = 'Booked' where passanger_ssn = 256558303;
sqlite> select * from booked;
Passanger_ssn  Train_Number  Ticket_Type  Status
-------------  ------------  -----------  ------
264816896      3             Premium      Booked
285200976      4             Premium      Booked
285200976      2             Premium      Booked
317434088      2             Premium      Booked
310908858      2             General      Booked
322273872      2             General      Booked
256558303      3             Premium      Booked
302548590      2             General      WaitL
284965676      3             Premium      WaitL
277292710      3             General      WaitL
331160133      3             General      WaitL
331293204      3             General      WaitL
290123298      3             General      WaitL
286411536      4             Premium      Booked
294860856      4             Premium      Booked
317434088      4             Premium      Booked
310908858      4             Premium      Booked
322273872      4             Premium      Booked
256558303      4             Premium      Booked
302548590      4             Premium      Booked
284965676      4             General      Booked
277292710      4             General      Booked
331160133      4             General      Booked
331293204      4             General      Booked
sqlite> 
```

5. select distinct First_Name, Last_Name from Available_On a, Passenger p, Booked b, train t where t.train_number = b.train_number and t.train_name = a.train_name and p.ssn = b.passanger_ssn and a.Sunday = '1' and b.status = 'Booked';



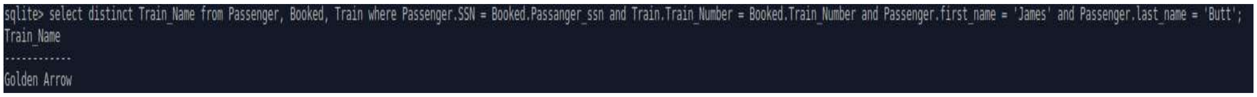
```
sqlite> select distinct First_Name, Last_Name from Available_On a, Passenger p, Booked b, train t where t.train_number = b.train_number and t.train_name = a.train_name and p.ssn = b.passanger_ssn and a.Sunday = '1' and b.status = 'Booked';
first_name  last_name
----------  ----------
Josephine   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
```

6. select Train_name, GenSeatsOccupied + PremiumSeatsOccupied, GenSeatsAvailable + PremiumSeatsAvailable from Train_Status;



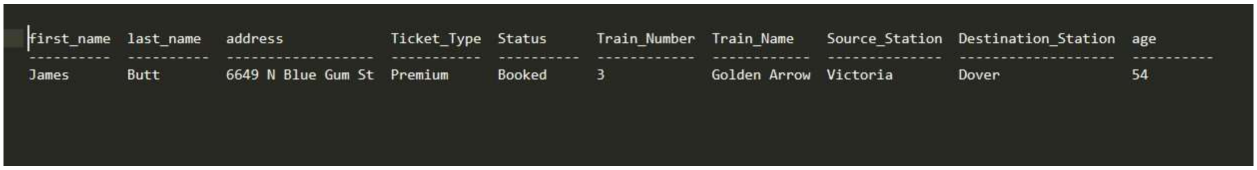
```
sqlite> select Train_name, GenSeatsOccupied + PremiumSeatsOccupied, GenSeatsAvailable + PremiumSeatsAvailable from Train_Status;
Train_Name      GenSeatsOccupied + PremiumSeatsOccupied  GenSeatsAvailable + PremiumSeatsAvailable
--------------  ---------------------------------------  -----------------------------------------
Orient Express  10                                       10
Flying Scottsm  13                                       7
Guwahati Expre  11                                       9
Golden Chariot  13                                       7
```

7. select distinct Train_Name from Passenger, Booked, Train where Passenger.SSN = Booked.Passanger_ssn and Train.Train_Number = Booked.Train_Number and Passenger.first_name = 'James' and Passenger.last_name = 'Butt';



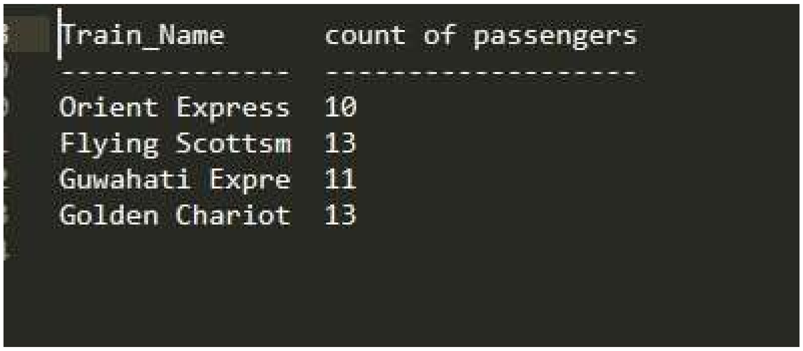
```
sqlite> select distinct Train_Name from Passenger, Booked, Train where Passenger.SSN = Booked.Passanger_ssn and Train.Train_Number = Booked.Train_Number and Passenger.first_name = 'James' and Passenger.last_name = 'Butt';
Train_Name
-----------
Golden Arrow
```

8. select p.first_name, p.last_name, p.address, b.ticket_type, b.status, t.train_number, t.train_name, t.source_station, t.Destination_Station, (cast(strftime('%Y', 'now')as int) - cast('19' || substr(substr('0000000000'|| p.bdate, length('0000000000'|| p.bdate)-9, 10), 9) as INTEGER)) as age from Passenger p , booked b, train t where age >= 50 and age <= 60 and b.passanger_ssn = p.ssn and t.train_number = b.train_number;



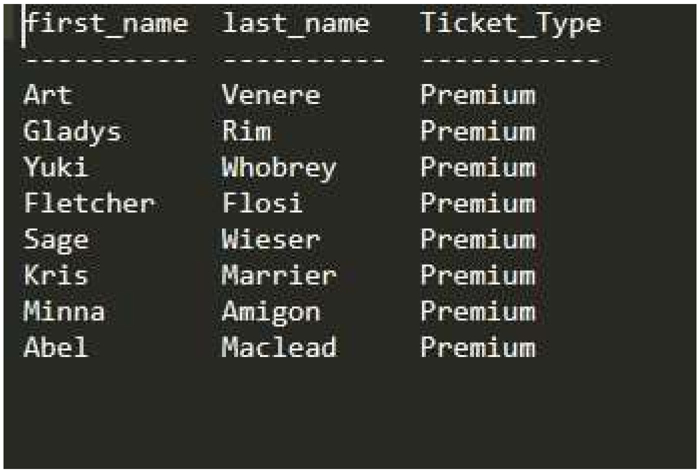
```
first_name  last_name   address             Ticket_Type  Status      Train_Number  Train_Name    Source_Station  Destination_Station  age
----------  ----------  ------------------  -----------  ----------  ------------  ------------  --------------  -------------------  ----------
James       Butt        6649 N Blue Gum St  Premium      Booked      3             Golden Arrow  Victoria        Dover                54
```

9. select Train_name, GenSeatsOccupied + PremiumSeatsOccupied as [count of passengers] from Train_Status;



```
Train_Name      count of passengers
--------------  --------------------
Orient Express  10
Flying Scottsm  13
Guwahati Expre  11
Golden Chariot  13
```

10. select distinct First_Name, Last_Name, Ticket_Type from Available_On a, Passenger p, Booked b, train t where t.train_number = b.train_number and t.train_name = a.train_name and p.ssn = b.passanger_ssn and a.Sunday = '1' and a.Saturday = '1' and b.ticket_type = 'Premium';



```
first_name  last_name   Ticket_Type
----------  ----------  -----------
Art         Venere      Premium
Gladys      Rim         Premium
Yuki        Whobrey     Premium
Fletcher    Flosi       Premium
Sage        Wieser      Premium
Kris        Marrier     Premium
Minna       Amigon      Premium
Abel        Maclead     Premium
```

6. Contribution list.

Worked on all of them as a group.