We, Hla Myint Myat, William Kim, and Leon declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. No part of this assignment has been copied manually or electronically from any other source (including web sites) or distributed to other students.

2- Specify what each member has done towards the completion of this work:

Name Task(s)

1. Hla Myint Myat Part 1 , 1 to 3NF, ERD, Creating Tables

2. William Kim Database Implementation, Part 2

3. Leon Part 3

BookingRecord Table

1NF >> breakdown repeating groups, atomic values only, setting PK

BOOKINGRECORD [BookingID, BookedOn]

FLIGHT [Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc]

FLIGHTOPERATION [Flight\_num, operatedBy, dep\_date, dep\_time, dep\_airport\_code, dep\_airport\_name, dep\_city]

ARRIVAL [Flight\_num, arr\_date, arr\_time, arrival\_airport\_code, arrival\_airport\_name, arr\_city]

2NF >> Checking Partial dependencies

BOOKINGRECORD [BookingID, BookedOn]

FLIGHT [Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc]

OPERATION [Flight\_num, operatedBy]

DEPARTURE [Flight\_num, dep\_date, dep\_time, dep\_airport\_code, dep\_airport\_name, dep\_city]

ARRIVAL [Flight\_num, arr\_date, arr\_time, arrival\_airport\_code, arrival\_airport\_name, arr\_city]

3NF >> no transitive dependency

BOOKINGRECORD [BookingID, BookedOn]

FLIGHT [Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc]

OPERATION [Flight\_num, operatedBy]

DEP\_FLIGHT [Flight\_num, dep\_date, dep\_time]

DEP\_DETAILS [ Flight\_num, dep\_airport\_code, dep\_airport\_name, dep\_city]

ARR\_FLIGHT [Flight\_num, arr\_date, arr\_time]

ARR\_DETAILS [ Flight\_num, arr\_airport\_code, arr\_airport\_name, arr\_city]

Ticket Table

Functional Dependencies

traveler\_Fname, traveler\_Lname --> Gender

travel\_id-->Meal\_preference

ticket\_price --> taxes\_fee

1NF: >> only atomic values

TICKET [eticket\_num, BookingID, traveler\_id, traveler\_Fname, traveler\_Lname, Gender, Meal\_preference, ticket\_price, taxes\_fees]

2NF: >> partial dependency

TICKET [eticket\_num, BookingID, Meal\_preference, ticket\_price, taxes\_fees, (FK)traveler\_id]

TRAVELER [traveler\_id, traveler\_Fname, traveler\_Lname, Gender]

3NF: >> no transitive dependency

TICKET [PK eticket\_num, BookingID, (FK)traveler\_id, (FK)ticketdetail\_id]

MEAL [(PK, FK) traveler\_id, Meal\_preference]

DETAIL\_PRICE [PK ticketdetail\_id, ticket\_price]

DETAIL\_TAX [(PK, FK) ticketdetail\_id, taxes\_fees]

TRAVELER [(PK, FK) traveler\_id, traveler\_Fname, traveler\_Lname]

GENDER [(PK, FK) traveler\_id, Gender]

***ERD***

A diagram of a company

Description automatically generated

Part 2.

1. CREATE USER DBS211\_Ass2\_Group\_11 IDENTIFIED BY password;
2. Includes
3. Both answers from 2 & 3 below

CREATE TABLE "BookingRecord" (

"BookingID" NUMBER,

"BookedOn" DATE,

PRIMARY KEY ("BookingID")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Flight" (

"Flight\_num" VARCHAR2(10),

"Airline\_code" VARCHAR2(10),

"Airline\_name" VARCHAR2(100),

"Aircraft\_code" VARCHAR2(10),

"Aircraft\_desc" VARCHAR2(100),

PRIMARY KEY ("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Operation" (

"Flight\_num" VARCHAR2(10),

"operatedBy" VARCHAR2(100),

PRIMARY KEY ("Flight\_num"),

FOREIGN KEY ("Flight\_num") REFERENCES "Flight"("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "DepFlight" (

"Flight\_num" VARCHAR2(10),

"dep\_date" DATE,

"dep\_time" TIMESTAMP,

PRIMARY KEY ("Flight\_num"),

FOREIGN KEY ("Flight\_num") REFERENCES "Flight"("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "DepDetails" (

"Flight\_num" VARCHAR2(10),

"dep\_airport\_code" VARCHAR2(10),

"dep\_airport\_name" VARCHAR2(100),

"dep\_city" VARCHAR2(100),

PRIMARY KEY ("Flight\_num"),

FOREIGN KEY ("Flight\_num") REFERENCES "Flight"("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "ArrFlight" (

"Flight\_num" VARCHAR2(10),

"arr\_date" DATE,

"arr\_time" TIMESTAMP,

PRIMARY KEY ("Flight\_num"),

FOREIGN KEY ("Flight\_num") REFERENCES "Flight"("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "ArrDetails" (

"Flight\_num" VARCHAR2(10),

"arr\_airport\_code" VARCHAR2(10),

"arr\_airport\_name" VARCHAR2(100),

"arr\_city" VARCHAR2(100),

PRIMARY KEY ("Flight\_num"),

FOREIGN KEY ("Flight\_num") REFERENCES "Flight"("Flight\_num")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Traveler" (

"traveler\_id" NUMBER,

"traveler\_Fname" VARCHAR2(50),

"traveler\_Lname" VARCHAR2(50),

PRIMARY KEY ("traveler\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Ticket" (

"eticket\_num" NUMBER,

"BookingID" NUMBER,

"traveler\_id" NUMBER,

"ticketdetail\_id" NUMBER,

PRIMARY KEY ("eticket\_num"),

FOREIGN KEY ("BookingID") REFERENCES "BookingRecord"("BookingID"),

FOREIGN KEY ("traveler\_id") REFERENCES "Traveler"("traveler\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Meal" (

"traveler\_id" NUMBER,

"Meal\_preference" VARCHAR2(100),

PRIMARY KEY ("traveler\_id"),

FOREIGN KEY ("traveler\_id") REFERENCES "Traveler"("traveler\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "DetailPrice" (

"ticketdetail\_id" NUMBER,

"ticket\_price" NUMBER,

PRIMARY KEY ("ticketdetail\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "DetailTax" (

"ticketdetail\_id" NUMBER,

"taxes\_fees" NUMBER,

PRIMARY KEY ("ticketdetail\_id"),

FOREIGN KEY ("ticketdetail\_id") REFERENCES "DetailPrice"("ticketdetail\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

CREATE TABLE "Gender" (

"traveler\_id" NUMBER,

"Gender" VARCHAR2(10),

PRIMARY KEY ("traveler\_id"),

FOREIGN KEY ("traveler\_id") REFERENCES "Traveler"("traveler\_id")

);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

INSERT ALL

    INTO BookingRecord (BookingID, BookedOn) VALUES (56753365, ‘28-Jan-19’)

    INTO BookingRecord (BookingID, BookedOn) VALUES (56753365, ‘28-Jan-19’)

    INTO BookingRecord (BookingID, BookedOn) VALUES (56753936, ‘25-Jan-19’)

    INTO BookingRecord (BookingID, BookedOn) VALUES (56753936, ‘25-Jan-19’)

    INTO Flight (Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc) VALUES              (‘AF393’, ‘AF’, ‘Air France’, 772, ‘BOEING 777 285-305 STD’)

    INTO Flight (Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc) VALUES             (‘AF5106’, ‘AF’, ‘Air France’, 332, ‘AIRBUS INDUSTRIE JET 200-345 STD’)

    INTO Flight (Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc) VALUES              (‘AF393’, ‘AF’, ‘Air France’, 772, ‘BOEING 777 285-305 STD’)

    INTO Flight (Flight\_num, Airline\_code, Airline\_name, Aircraft\_code, Aircraft\_desc) VALUES           (‘AF386’, ‘AF’, ‘Air France’, 772, ‘BOEING 777 285-305 STD SEATS’)

    INTO Operation (Flight\_num, operatedBy) VALUES (‘AF393’, ‘AF’)

    INTO Operation (Flight\_num, operatedBy) VALUES (‘AF5106’, ‘MEA’)

    INTO Operation (Flight\_num, operatedBy) VALUES (‘AF393’, ‘AF’)

    INTO Operation (Flight\_num, operatedBy) VALUES (‘AF386’, ‘AF’)

    INTO DepFlight (Flight\_num, dep\_date, dep\_time) VALUES (‘AF393’, ‘22-Jun-19’, ‘9:20 PM’)

    INTO DepFlight (Flight\_num, dep\_date, dep\_time) VALUES (‘AF5106’, ‘23-Jun-19’, ‘1:40 PM’)

    INTO DepFlight (Flight\_num, dep\_date, dep\_time) VALUES (‘AF393’, ‘22-Jun-19’, ‘9:20 PM’)

    INTO DepFlight (Flight\_num, dep\_date, dep\_time) VALUES (‘AF386’, ‘17-Jun-19’, ‘5:00 PM’)

    INTO DepDetails (Flight\_num, dep\_airport\_code, dep\_airport\_name, dep\_city) VALUES (‘AF393’, ‘YYZ’, ‘Toronto Pearson Intl, Ontario’, Toronto)

   INTO DepDetails (Flight\_num, dep\_airport\_code, dep\_airport\_name, dep\_city) VALUES (‘AF5106’, ‘CDG’, ‘Charles de Gaul’, Paris)

   INTO DepDetails (Flight\_num, dep\_airport\_code, dep\_airport\_name, dep\_city) VALUES (‘AF393’, ‘YYZ’, ‘Toronto Pearson Intl, Ontario’, Toronto)

   INTO DepDetails (Flight\_num, dep\_airport\_code, dep\_airport\_name, dep\_city) VALUES (‘AF386’, ‘CDG’, ‘Charles de Gaul’, Paris)

   INTO ArrFlight (Flight\_num, arr\_date, arr\_time) VALUES (‘AF393’, ‘23-Jun-19’, ‘10:50 AM’)

   INTO ArrFlight (Flight\_num, arr\_date, arr\_time) VALUES (‘AF5106’, ‘23-Jun-19’, ‘6:55 PM’)

   INTO ArrFlight (Flight\_num, arr\_date, arr\_time) VALUES (‘AF393’, ‘23-Jun-19’, ‘10:50 AM’)

   INTO ArrFlight (Flight\_num, arr\_date, arr\_time) VALUES (‘AF386’, ‘17-Jun-19’, ‘7:20 PM’)

  INTO ArrDetails (Flight\_num, arr\_airport\_code, arr\_airport\_name, arr\_city) VALUES (‘AF393’, ‘CDG’, ‘Charles de Gaul’, Paris)

  INTO ArrDetails (Flight\_num, arr\_airport\_code, arr\_airport\_name, arr\_city) VALUES (‘AF393’, ‘BEY’, ‘’, Beirut)

  INTO ArrDetails (Flight\_num, arr\_airport\_code, arr\_airport\_name, arr\_city) VALUES (‘AF393’, ‘CDG’, ‘Charles de Gaul’, Paris)

  INTO ArrDetails (Flight\_num, arr\_airport\_code, arr\_airport\_name, arr\_city) VALUES (‘AF393’, ‘YYZ’, ‘Toronto Pearson Intl, Ontario’, Toronto)

  INTO Traveler (traveler\_id, traveler\_Fname, traveler\_Lname) VALUES (111, ‘Andrew’, ‘Smith’)

  INTO Traveler (traveler\_id, traveler\_Fname, traveler\_Lname) VALUES (112, ‘Mariam’, ‘Daoud’)

  INTO Traveler (traveler\_id, traveler\_Fname, traveler\_Lname) VALUES (113, ‘Yasmine’, ‘Ch’)

  INTO Traveler (traveler\_id, traveler\_Fname, traveler\_Lname) VALUES (114, ‘ Hasan’, ‘Ch’)

  INTO Ticket (eticket\_num, BookingID, traveler\_id, ticketdetail\_id) VALUES (573480996631, 56753365, 111, 000)

  INTO Ticket (eticket\_num, BookingID, traveler\_id, ticketdetail\_id) VALUES (573480996619, 56753365, 112, 001)

  INTO Ticket (eticket\_num, BookingID, traveler\_id, ticketdetail\_id) VALUES (573480996620, 56753936, 113, 002)

  INTO Ticket (eticket\_num, BookingID, traveler\_id, ticketdetail\_id) VALUES (573480996621, 56753936, 114, 003)

  INTO Meal (traveler\_id, Meal\_preference) VALUES (111, Vegetarian)

  INTO Meal (traveler\_id, Meal\_preference) VALUES (112, Halal)

  INTO Meal (traveler\_id, Meal\_preference) VALUES (113, Vegetarian)

  INTO Meal (traveler\_id, Meal\_preference) VALUES (114, Halal)

  INTO DetailPrice (ticketdetail\_id, ticket\_price) VALUES (000, 1200)

  INTO DetailPrice (ticketdetail\_id, ticket\_price) VALUES (001, 1353)

  INTO DetailPrice (ticketdetail\_id, ticket\_price) VALUES (002, 1142)

  INTO DetailPrice (ticketdetail\_id, ticket\_price) VALUES (003, 1142)

  INTO DetailTax (ticketdetail\_id, taxes\_fees) VALUES (000, 182)

  INTO DetailPrice (ticketdetail\_id, taxes\_fees) VALUES (001, 182)

  INTO DetailPrice (ticketdetail\_id, taxes\_fees) VALUES (002, 180)

  INTO DetailPrice (ticketdetail\_id, taxes\_fees) VALUES (003, 180)

  INTO Gender (traveler\_id, Gender) VALUES (111, ‘Male’)

  INTO Gender (traveler\_id, Gender) VALUES (112, ‘Female’)

  INTO Gender (traveler\_id, Gender) VALUES (113, ‘Female’)

INTO Gender (traveler\_id, Gender) VALUES (114, ‘Male’

SELECT 1 FROM DUAL;

PART 3:

Q1 –

SELECT

F.Flight\_num AS Flight\_Number,

F.Airline\_code AS Airline\_Code,

F.Airline\_name AS Airline\_Name,

F.Aircraft\_code AS Aircraft\_Code,

F.Aircraft\_desc AS Aircraft\_Description,

D.dep\_airport\_code AS Departure\_Airport\_Code,

D.dep\_airport\_name AS Departure\_Airport\_Name,

A.arr\_airport\_code AS Arrival\_Airport\_Code,

A.arr\_airport\_name AS Arrival\_Airport\_Name

FROM

Flight F

JOIN

DepDetails D ON F.Flight\_num = D.Flight\_num

JOIN

ArrDetails A ON F.Flight\_num = A.Flight\_num

WHERE

F.Flight\_num = 'AF393';

Q2 –

SELECT

BR.BookingID,

BR.BookedOn AS Booking\_Date,

F.Flight\_num AS Flight\_Number,

DD.dep\_airport\_code AS Departure\_Airport\_Code,

DF.dep\_date AS Departure\_Date,

DF.dep\_time AS Departure\_Time,

AD.arr\_airport\_code AS Arrival\_Airport\_Code,

AF.arr\_date AS Arrival\_Date,

AF.arr\_time AS Arrival\_Time

FROM

BookingRecord BR

JOIN

Ticket T ON BR.BookingID = T.BookingID

JOIN

Flight F ON T.Flight\_num = F.Flight\_num

JOIN

DepFlight DF ON F.Flight\_num = DF.Flight\_num

JOIN

DepDetails DD ON F.Flight\_num = DD.Flight\_num

JOIN

ArrFlight AF ON F.Flight\_num = AF.Flight\_num

JOIN

ArrDetails AD ON F.Flight\_num = AD.Flight\_num

WHERE

BR.BookingID = 56753365

ORDER BY

DF.dep\_date, DF.dep\_time;

Q3 –

SELECT

BR.BookingID,

T.eticket\_num AS Eticket\_Number,

TR.traveler\_Fname AS Traveler\_First\_Name,

TR.traveler\_Lname AS Traveler\_Last\_Name

FROM

BookingRecord BR

JOIN

Ticket T ON BR.BookingID = T.BookingID

JOIN

Traveler TR ON T.traveler\_id = TR.traveler\_id

WHERE

BR.BookingID = 56753365;

Q4 –

SELECT

BR.BookingID,

SUM(DP.ticket\_price + DT.taxes\_fees) AS Total\_Fees

FROM

BookingRecord BR

JOIN

Ticket T ON BR.BookingID = T.BookingID

JOIN

DetailPrice DP ON T.ticketdetail\_id = DP.ticketdetail\_id

JOIN

DetailTax DT ON DP.ticketdetail\_id = DT.ticketdetail\_id

WHERE

BR.BookingID = 56753365;

Q5 –

CREATE VIEW Book\_ticket AS

SELECT

BR.BookingID,

COUNT(T.eticket\_num) AS Number\_of\_Tickets

FROM

BookingRecord BR

LEFT JOIN

Ticket T ON BR.BookingID = T.BookingID

GROUP BY

BR.BookingID;