Group ID: Names: Score: / 100 CS122A HW2

```
-- Since Maintenance Engineer, Pilot, Flight Attendant, and Operation Staff COVER
Employee, we don't need to create a separate Employee table.
-- Stores MaintenanceEngineer specific information.
CREATE TABLE MaintenanceEngineer (
        meid                    INTEGER,
        phone_number            VARCHAR(20),
        birthdate               DATE,
        ssn                     CHAR(9),   -- 123456789
        job_title               VARCHAR(50),
        address_street          VARCHAR(50),
        address_city            VARCHAR(20),
        address_state           VARCHAR(20),
        address_zipcode         CHAR(5),  -- 92627
        skill                   VARCHAR(50),
        PRIMARY KEY (meid)
);

-- Stores Pilot specific information.
CREATE TABLE Pilot (
        pid                     INTEGER,
        phone_number            VARCHAR(20),
        birthdate               DATE,
        ssn                     CHAR(9),   -- 123456789
        job_title               VARCHAR(50),
        address_street          VARCHAR(50),
        address_city            VARCHAR(20),
        address_state           VARCHAR(20),
        address_zipcode         CHAR(5),  -- 92627
        since                   INTEGER,
        PRIMARY KEY (pid)
);

-- Stores FlightAttendant specific information.
CREATE TABLE FlightAttendant (
        faid                    INTEGER,
        phone_number            VARCHAR(20),
        birthdate               DATE,
        ssn                     CHAR(9),   -- 123456789
        job_title               VARCHAR(50),
        address_street          VARCHAR(50),
        address_city            VARCHAR(20),
        address_state           VARCHAR(20),
```

```
        address_zipcode                 CHAR(5),  -- 92627
        service_year                    INTEGER,
        PRIMARY KEY (faid)
);

-- Stores OperationStaff specific information.
CREATE TABLE OperationStaff (
        osid                            INTEGER,
        phone_number                    VARCHAR(20),
        birthdate                       DATE,
        ssn                             CHAR(9),   -- 123456789
        job_title                       VARCHAR(50),
        address_street                  VARCHAR(50),
        address_city                    VARCHAR(20),
        address_state                   VARCHAR(20),
        address_zipcode                 CHAR(5),  -- 92627
        department                      VARCHAR(50),
        PRIMARY KEY (osid)
);

-- Stores Airplane specific information.
CREATE TABLE Airplane (
        registration_number             VARCHAR(10),
        model_number                    VARCHAR(10),
        purchased_year                  INTEGER,
        manufactured_year               INTEGER,
        capacity                        INTEGER,
        PRIMARY KEY (registration_number)
);

-- Stores Customer specific information.
CREATE TABLE Customer (
        cid                             INTEGER,
        ssn                             CHAR(9),   -- 123456789
        gender                          VARCHAR(6),
        email                           VARCHAR(30),
        address_street                  VARCHAR(50),
        address_city                    VARCHAR(20),
        address_state                   VARCHAR(20),
        address_zipcode                 CHAR(5),  -- 92627
        PRIMARY KEY (cid)
);

-- Stores each customer's credit card information.
CREATE TABLE CreditCard (
```

```
        cid                             INTEGER,
        card_number                     VARCHAR(20),
        expr_date                       CHAR(6),  -- YYYYMM (201603)
        PRIMARY KEY (card_number),
        FOREIGN KEY (cid) REFERENCES Customer(cid)
);

-- Stores Airport related information.
CREATE TABLE Airport (
        IATA_code                       CHAR(3),  --  SNA
        name                            VARCHAR(40),
        airport_city                    VARCHAR(20),
        airport_state                   VARCHAR(20),
        PRIMARY KEY (IATA_code)
);

-- Stores Flight related information.
CREATE TABLE Flight (
        flight_number                   VARCHAR(8),
        projected_departure_datetime    DATETIME,
        projected_arrival_datetime      DATETIME,
        aiplane_registration_number     VARCHAR(10),  -- Airplane assigned
        departure_airport_IATA_code     CHAR(3) NOT NULL,  -- Flight departure
Airport
        actual_departure_datetime       DATETIME,
        arrival_airport_IATA_code       CHAR(3) NOT NULL,  -- Flight arrival
Airport
        actual_arrival_datetime         DATETIME,
        PRIMARY KEY (flight_number,projected_departure_datetime),
        FOREIGN KEY (aiplane_registration_number) REFERENCES
Airplane(registration_number),
        FOREIGN KEY (departure_airport_IATA_code) REFERENCES
Airport(IATA_code),
        FOREIGN KEY (arrival_airport_IATA_code) REFERENCES Airport(IATA_code)
);

-- Stores Lounge related information.
CREATE TABLE Lounge (
        lid                             INTEGER,
        location                        VARCHAR(50),
        airport_IATA_code               CHAR(3) NOT NULL,  -- Airport where
lounge is
        PRIMARY KEY (lid),
        FOREIGN KEY (airport_IATA_code) REFERENCES Airport(IATA_code)
);
```

```
-- Stores DishOrder related information.
CREATE TABLE DishOrder (
        oid                             INTEGER,
        cid                             INTEGER NOT NULL,  -- A Customer
places an order
        lid                             INTEGER,  -- A Lounge serves an order
        order_datetime                  DATETIME,
        total_amount                    DECIMAL(7,2),  -- 00000.00
        PRIMARY KEY (oid),
        FOREIGN KEY (cid) REFERENCES Customer(cid),
        FOREIGN KEY (lid) REFERENCES Lounge(lid)
);

-- Stores Dish related information.
-- Since Dish is a weak entity, we need to associate it with the Lounge(lid)
CREATE TABLE Dish (
        lid                             INTEGER,  --  A lounge has Dish
        name                            VARCHAR(40),
        price                           DECIMAL(6,2),   -- 0000.00
        PRIMARY KEY (lid, name),
        FOREIGN KEY (lid) REFERENCES Lounge(lid) ON DELETE CASCADE
);


-- Stores the detailed information of each order.
-- Since this is M:N relationship between DishOrder and Dish,
-- We need to have a separate relation.
CREATE TABLE DishOrderContainsDish (
        oid                             INTEGER,
        lid                             INTEGER,
        name                            VARCHAR(40),
        quantity                        INTEGER,
        PRIMARY KEY (oid, lid, name),
        FOREIGN KEY (oid) REFERENCES DishOrder (oid),
        FOREIGN KEY (lid, name) REFERENCES Dish(lid, name)
);

-- Stores the detailed information of each reservation.
-- Since this is M:N relationship between Customer and Flight,
-- We need to have a separate relation.
CREATE TABLE CustomerReservesFlight (
        cid                             INTEGER,  -- Customer reserving a flight
        flight_number                   VARCHAR(8),  --  Flight to be reserved
        projected_departure_datetime    DATETIME,
```

```
	purchased_datetime		DATETIME,
	purchased_price			DECIMAL(7,2),  -- 00000.00
	quantity			INTEGER,
	PRIMARY KEY (cid, flight_number, projected_departure_datetime),
	FOREIGN KEY (cid) REFERENCES Customer (cid),
	FOREIGN KEY (flight_number, projected_departure_datetime) REFERENCES
Flight(flight_number, projected_departure_datetime)
);

-- Stores the detailed information of each flight's pilots.
-- Since this is M:N relationship between Pilot and Flight,
-- We need to have a separate relation.
CREATE TABLE PilotOperatesFlight (
	pid				INTEGER,  -- Operating pilot
	flight_number			VARCHAR(8),
	projected_departure_datetime	DATETIME,
	PRIMARY KEY (pid, flight_number, projected_departure_datetime),
	FOREIGN KEY (pid) REFERENCES Pilot (pid),
	FOREIGN KEY (flight_number, projected_departure_datetime) REFERENCES
Flight(flight_number, projected_departure_datetime)
);

-- Stores the detailed information of each flight's flight attendants.
-- Since this is M:N relationship between FlightAttendant and Flight,
-- We need to have a separate relation.
CREATE TABLE FlightAttendantParticipatesFlight (
	faid				INTEGER,  -- Participating flight attendant
	flight_number			VARCHAR(8),
	projected_departure_datetime	DATETIME,
	PRIMARY KEY (faid, flight_number, projected_departure_datetime),
	FOREIGN KEY (faid) REFERENCES FlightAttendant (faid),
	FOREIGN KEY (flight_number, projected_departure_datetime) REFERENCES
Flight(flight_number, projected_departure_datetime)
);

- Stores the detailed information of each airplane maintenance.
-- Since this is M:N relationship between MaintenanceEngineer and Airplane,
-- We need to have a separate relation.
CREATE TABLE MaintenanceEngineerMaintainsAirplane (
	meid				INTEGER,
	Aiplane_registration_number	VARCHAR(10),
	PRIMARY KEY (meid,aiplane_registration_number),
	FOREIGN KEY (meid) REFERENCES MaintenanceEngineer(meid),
	FOREIGN KEY (aiplane_registration_number) REFERENCES
Airplane(registration_number)
```

);