location = (locationID)

route = (routeID)

airport = (airportID, airportName, city, state, country, locationID [fk15])
fk15: locationID → location(locationID)

airline = (airlineID, revenue)

airplane = (tailNumber, seat_cap, speed, airlineID [fk5], locationID [fk17], planeType, skids, props, planeEngines)
fk5: airlineID → airline(airlineID)
fk17: locationID → location(locationID)
planeType is a string
planeEngines are the jet's engines

flight = (flightID, cost, routeID [fk9], progress, flightStatus, next_time, tailID[fk6])
fk6: tailID → airplane(tailNumber)
fk9: routeID → route(routeID)

person = (personID, firstName, lastName, locationID[fk14])
fk14: locationID → location(locationID)

pilot = (experience, taxID, personID [fk10], flightID [fk16])
taxID is a unique key
fk10: personID → person(personID)
fk16: flightID → flight(flightID)

passenger = (miles, funds, personID[fk13])
fk13: personID → person(personID)

vacation = (destination, sequence, personID [fk11])
fk11: personID → passenger(personID)

license = (license, personID[fk12])
fk12: personID → pilot(personID)

leg = (legID, distance, departureID [fk1], arrivalID [fk2])
fk1: departureID → airport(airportID)

fk2: arrivalID → airport(airportID)

contains = (routeID [fk3], legID [fk4], sequence)
fk3: routeID → route(routeID)
fk4: legID → leg(legID)

Unhandled Constraints

- Ensure that every flight follows a route.
- Ensure that every leg departs and arrives at an airport.
- Ensure that every airline owns at least one airplane.
- Ensure that each person is either a passenger or a pilot.
- Ensure that each person occupies a location.
- Ensure that each leg contains a route.
- Ensure that prop and jet are distinct.
- Ensure that pilot and passenger are distinct.