**GROUP 65 SCHEMA**

person = (personID, firstName, lastName, location [fk12])

fk12: location → location.locID, location is not null

pilot = (personID [fk2], taxID, experience, flight\_commanded [fk11])

fk2: personID → person.personID

fk11: flight\_commanded → flight.flightID

license = (personID [fk7], license\_type)

fk7: personID → pilot.personID

passenger = (personID [fk3], miles, funds)

fk3: personID → person.personID

vacation = (personID [fk6], destination, sequence)

fk6: personID → passenger.personID

airline = (airlineID, revenue)

airplane = (airlineID [fk1], tail\_num, speed, seat\_cap, locID [fk5])

fk1: airlineID → airline.airlineID

fk5: locID → location.locID

support = (airlineID, tail\_num [fk98], flightID [fk99], progress, flight\_status, next\_time)

fk98: airlineID, tail\_num[fk98] → airplane.airlineID, airplane.tail\_num

fk99: flightID → flight.flightID

prop = (airlineID, tail\_num [fk17], props, skids)

fk17: airlineID, tail\_num → airplane.airlineID, airplane.tail\_num

jet = (airlineID, tail\_num [fk17], jet\_engines)

fk18: airlineID, tail\_num → airplane.airlineID, airplane.tail\_num

airport = (airportID, airport\_name, city, state, country, locID [fk4])

fk4: locID → location.locID

leg = (legID, distance, departure\_airport [fk14], arrival\_airport [fk15])

fk14: departure\_airport → airport.airportID, departure\_airport is not null

fk15: arrival\_airport → airport.airportID, arrival\_airport is not null

route = (routeID)

leg\_contains = (routeID [fk8], legID [fk9], sequence)

fk8: routeID → route.routeID

fk9: legID → leg.legID

flight = (flightID, cost, route [fk16])

fk16: route → route.routeID, route is not null

location = (locID)

**UNHANDLED CONSTRAINTS**

* Each person must be a pilot or passenger.
* Each airline must own one or more airplanes
  + From problem requirements: “Airlines own one or more airplanes that can be used to accomplish these tasks.”
* Each jet airplane must have 2 or more pilots assigned before takeoff
  + Pilot does not have any relationship with airplane, can not fulfill pilot 2 or more pilots for jet
* Each propeller airplane must have a pilot assigned before takeoff
  + Pilot does not have any relationship with airplane, can not fulfill pilot 1 or more pilots for propeller
* Number of passengers boarding a flight should not exceed the seating capacity of the airplane(s) it is supported by.
* Each passenger must have enough funds to board a flight.
  + Passenger has funds and flight has cost but there is no relationship between passenger and flight
* Each passenger must board flights that depart at their location and arrive at their desired destination.
  + Passenger and flight do not have a relationship between them
* Each airplane crew must be recycled at the end of a flight
* Each flight must be removed at the end of its route