model Airport

-- classes
class Airport
attributes
   name:String
end

class Flight
attributes
   departTime:Time
   arrivalTime:Time
   duration:Interval
   maxNrPassengers:Integer init: 100 --initialization example
end

class Airline
attributes
   name:String
end

class Passenger
attributes
   minAge:Integer
   age:Integer
   needsAssistance:Boolean
operations
   book(f:Flight)
end

class Time
attributes
   month:String
   day:Integer
   year:Integer
   hour:Integer
   minute:Integer
end

class Interval
attributes
   nrOfDays:Integer
   nrOfHours:Integer
   nrOfMinutes:Integer
operations
   equals(i:Interval):Boolean
end

-- associations

association one between
   Airport[1] role origin
   Flight[*] role departingFlights
end

association two between
   Airport[1] role destination
Flight[*] role arrivingFlights
end

association three between
  Flight[*] role flights
  Airline[1] role airline
end

association four between
  Airline[0..1] role airline
  Passenger[0..1] role CEO
end

association five between
  Flight[1] role flight
  Passenger[*] role passengers
end

-- constraints
context Flight
inv: self.maxNrPassengers <= 500
inv: origin <> destination

context Passenger
inv: self.age >= self.minAge

context Airline
inv: flights->select(maxNrPassengers >100)->notEmpty()

context Passenger::book(f:Flight)
pre: age < 90
post: age = age@pre