# Documentazione primo homework

## Flight

*Flight* è una superclasse, che sarà poi specializzata in *Departing* e *Arrival*, come specializzazione totale e disgiunta.

Per quanto riguarda gli attributi:

* L’ id deve essere univoco.
* status prende i possibili valori dall’ enumeration *FlightStatus*, può quindi essere: “*programmed”*,  
  *“cancelled”*,  
  *“departed”*,  
  *“delayed”*,  
  *“landed”*.
* max\_seats e free\_seats vengono utilizzati per vari controlli con l’inserimento/rimozione di eventuali prenotazioni e/o passeggeri.

È in relazione con diverse classi:

* *Booking*: c’è una relazione 1 a molti tra queste due classi, la visibilità è bidirezionale perché da un volo potrebbe essere utile, per motivi di gestione, conoscere tutte le prenotazioni. Sarà implementata in *Flight* con un ArrayList di *Booking*.
* *Passenger*: c’è una relazione 1 a molti, la visibilità è unidirezionale (*Flight* vede *Passenger*, ma non viceversa, per il motivo vedere *Passenger*). Sarà implementata con un ArrayList di *Passenger* in *Flight*.

## Departing

*Departing* è una sottoclasse di *Flight*, rappresenta gli aerei in partenza da Napoli.

Per quanto riguarda gli attributi, “departure\_delay” è stato rappresentato come un intero (inizialmente posto a 0), poiché abbiamo deciso di salvare i minuti di ritardo rispetto all’orario di partenza previsto, per poi convertirlo opportunamente a seconda di come ci serve trattarlo nei vari contesti.

È in relazione con la classe *Gate*, con una relazione 1 a molti, la visibilità è uni direzionale (il gate non deve vedere gli aerei che partono da esso, tanto al più ne partirà uno per volta). E questa relazione è implementata mettendo in *Departing* l’id del gate come attributo (inizialmente con valore null, sarà istanziato quando l’aereo starà per partire).

## Arriving

*Arriving* è una sottoclasse di *Flight*, rappresenta gli aerei in arrivo a Napoli.

Per quanto riguarda gli attributi, “arrival\_delay” è stato rappresentato come un intero (inizialmente posto a 0), poiché abbiamo deciso di salvare i minuti di ritardo rispetto all’orario di arrivo previsto, per poi convertirlo opportunamente a seconda di come ci serve trattarlo nei vari contesti.

## Gate

*Gate* rappresenta le strutture da cui ogni aereo in partenza può partire, è rappresentato da un id univoco ed è in una relazione 1 a molti con la classe *Departing*, con visibilità unidirezionale (il gate non vede gli aerei che partono da esso), e per questo motivo non ha attributi che si riferiscono a qualche aereo in partenza.