

## Aerei: ereditarietà, interfacce e tanto polimorfismo!

In un aerodromo si vuol mantenere informazioni sugli aeromobili.

Di ogni aeromobile si vuole sapere la sigla (string) che lo identifica.

Gli alianti sono aeromobili caratterizzati da un numero (int) che ne descrive l'efficienza aerodinamica. Degli aerei a motore si vuole sapere la potenza in CV del propulsore (double).

Gli aerei a motore si distinguono in: jet e turbo-elica. Dei jet è necessario conoscere il numero di motori che supportano, mentre dei turbo-elica il numero di giri che può raggiungere l'elica.

Scrivi un'interfaccia Aeromodello contenente un metodo String getTipoAeromobile() che restituisce la tipologia di aeromobile ("ALIANTE", "JET", "TURBO-ELICA") e un metodo boolean piuPotente(Aeromobile x) per verificare se un aeromobile (this) è più potente dell'aeromobile x. Tale metodo deve restituire true/false se gli oggetti coinvolti nel confronto sono compatibili, mentre deve lanciare un'eccezione se si tenta di effettuare confronti assurdi (es. E' assurdo confrontare la potenza di un ALIANTE con un JET, mentre un JET con un TURBO-ELICA anche no!)

Intercettare e gestire l'eccezione di incompatibilità di aeromobili e scrivere un messaggio del tipo:

***Un ALIANTE non può essere paragonato a un Jet.***

Oppure

***Un JET non può essere paragonato a un Aliante.***

Scrivi le classi pubbliche sapendo che Aliante e Aeromotore estendono Aeromobile e implementano l'interfaccia Aeromodello ....

Ciascuna classe deve fornire un costruttore, un metodo get (per accedere all'attributo specifico) e il toString .

Nel main istanziare 2 alianti, 2 jet e 2 turboelica. Prevedere e testare tutti i casi possibili di confronto delle potenze tra i vari aeromobili e fornire opportuni messaggi in output.

Prima di procedere creare un DIAGRAMMA UML relativo alla realtà descritta.