

# Test Ereditarietà e polimorfismo

1) Che cosa è l'ereditarietà? spiega in un breve?

L'ereditarietà è una relazione che esiste tra 2 classi. Consiste nell'avere una classe denominata padre e una classe derivata dalla precedente denominata figlia. Si può pensare all'ereditarietà proprio come ad una famiglia. Infatti la classe figlia eredita attributi e metodi presenti nella classe padre e può eventualmente aggiungerne dei nuovi. Una analogia nella vita reale si ha quando il figlio assume il cognome del padre, gli stessi geni etc. ma il figlio può avere cose in più ad esempio la passione per lo sport che magari il padre non aveva

2) Cosa si utilizza per definire una sotto classe?

- ☒ Extend
- ☐ Implemet
- ☐ Abstract

3) Scrivere la sintassi di un costruttore

```
Public NomeClasse(int a, int b){  
  
    Attr1 = a;  
  
    attr2 = b;  
  
}
```

4) Cosa succede se la super classe non ha alcun costruttore?

La JVM dà errore di compilazione se la supeclasse non possiede un costruttore di default

5) Cosa significa overriding in java?

Overriding significa ridefinizione di un metodo. Il metodo overridden deve avere la stessa signature, quindi stesso nome e stessi parametri nelle stesse posizioni

6) Che cosa è la classe object e cosa fa?

La classe object è la classe madre di tutte le altre classi, infatti tutte le altre classi ereditano da lei e possiede dei metodi che è bene overridingli nelle proprie classi.

7) Come possiamo disabilitare l'ereditarietà di una classe?

Aggiungendo la keyword final dopo il modificatore d'accesso della classe

8) Che cosa è il polimorfismo?

Il polimorfismo è una tecnica che consente a una istanza di una classe di far riferimento ad oggetti diversi (delle sue sottoclassi) nel corso del programma.

9) Quale delle due funzioni viene chiamata?

```
Auto a = new Auto();
```

```
Modello m = a;
```

```
m.annoDiFabricazione(2010);
```

Auto
annoDiFabbricazione

Modello
annoDiFabbricazione

Viene chiamata la funzione presente nella classe Auto

10) è necessario aggiungere nuove classi per modificare quelle già presenti?

- ☐ Si  
☒ No

motivare il perché.

Perché si può sfruttare l'ereditarietà e il polimorfismo per creare una nuova classe che eredita le stesse funzionalità e overridarle nel caso si voglia specificare con dettaglio la funzionalità della classe derivata.