



Java secondo contatto

Soluzione Esercizio - Classe Vettore



Esercizio per Casa

- Classe Vettore (vettore geometrico con origine nel centro degli assi)

```
public class Vettore{  
    //crea un vettore con ascissa e ordinata specificati  
    public Vettore(double x, double y){...}  
    //ritorna il valore dell'ascissa  
    public double getAscissa(){...}  
    //ritorna il valore dell' ordinata  
    public double getOrdinata(){...}  
    //stampa su out il vettore nella forma (x,y)  
    public void stampa() {...}  
    //somma al vettore corrente quello passato come parametro e modifica l'oggetto  
    public void somma(Vettore f) {...}  
    //calcola il modulo del vettore  
    public double calcolaModulo() {...}  
    //moltiplica il vettore corrente per uno scalare  
    public double moltiplicaPerScalare(double s) {...}  
}
```



Soluzione/1

```
■ public class Vettore{
■     double x;
■     double y;
■     //crea un vettore con ascissa e ordinata specificati
■     public Vettore(double x, double y){
■         this.x=x;
■         this.y=y;
■     }
■     //ritorna il valore dell'ascissa
■     public double getAscissa(){
■         return x;
■     }
■     //ritorna il valore dell' ordinata
■     public double getOrdinata(){
■         return y;
■     }
■     ...
```



Soluzione/2

- `//calcola il modulo del vettore`
- `public double calcolaModulo() {`
- `return Math.sqrt(x*x + y*y);`
- `}`
- `//moltiplica il vettore corrente per uno scalare`
- `public void moltiplicaPerScalare(double s) {`
- `this.x=x*s;`
- `this.y=y*s;`
- `}`
- `...`



Soluzione/3

//main

```
■ public static void main(String [] s){  
■     Vettore v=new Vettore(3,4);  
■     v.stampa();  
■     System.out.println("Modulo: "+v.calcolaModulo());  
■     v.moltiplicaPerScalare(5);  
■     v.stampa();  
■     Vettore v1=new Vettore(1,2);  
■     Vettore v2=new Vettore(6,7);  
■     v2.somma(v1);  
■     v2.stampa();  
■ }  
■ }
```



Utilizzo delle variabili

- Variabili di istanza (campi della classe)
 - Rappresentano lo stato dell'oggetto e contengono dati utili anche all'uscita dei metodi di classe
- Variabili locali
 - Utilizzate all'interno dei metodi; servono come variabili d'appoggio; al termine del metodo vengono eliminate
- Parametri
 - Utilizzate nei metodi e vengono usate per ricevere valori dall'esterno
- return
 - Variabile implicita per restituire il valore al chiamante