

L2 - Metodi e costruttori - Avanzati

Java · [overloading](#), [piu' costruttori](#), [metodi che usano altri metodi](#)

Materiale didattico di Moreno Sardella · morenosardella.com

Prima di iniziare. Questi esercizi sono piu' impegnativi: combinano piu' concetti e si avvicinano al livello della verifica. Progetta prima su carta (classi, metodi, dati), poi scrivi il codice. Le soluzioni in fondo mostrano *una* via corretta: la tua puo' differire, conta che la logica funzioni e che il codice compili.

Esercizi

Esercizio 1 — Overloading

STESSO NOME, FIRME DIVERSE

Scrivi due metodi **area**: **area(double lato)** per il quadrato e **area(double b, double h)** per il rettangolo. Dimostra nel main che Java sceglie il metodo giusto in base ai parametri.

```
public static double area(double lato) { /*...*/ }
public static double area(double b, double h) { /*...*/ }
```

Esercizio 2 — Due costruttori

COSTRUTTORE DI DEFAULT

Dai a **Studente** (nome, media) due costruttori: uno con nome e media, e uno con il solo nome che imposta media a 6.0. Usa **this(...)** nel secondo per richiamare il primo.

```
public Studente(String nome) {
    this(nome, 6.0); // richiama l'altro costruttore
}
```

Esercizio 3 — Metodo che usa un metodo

RIUSO INTERNO

In **Conto** (saldo) scrivi **preleva(double x)** che preleva solo se c'e' saldo, e **bonifico(Conto dest, double x)** che usa preleva su this e versa sul conto destinatario.

```
public void bonifico(Conto dest, double x) {
    // preleva da this, versa su dest
}
```

Esercizio 4 — Distanza tra due punti

THIS + PARAMETRO OGGETTO

Crea **Punto** (x, y) con un metodo **distanzaDa(Punto p)** che restituisce la distanza euclidea tra l'oggetto corrente e p (usa Math.sqrt).

```
public double distanzaDa(Punto p) {
    // sqrt((this.x-p.x)^2 + (this.y-p.y)^2)
}
```

Soluzioni

Confronta solo dopo aver provato da solo. Esistono soluzioni equivalenti: conta la logica, non la copia esatta.

Soluzione 1

L'overloading distingue i metodi dal **numero/tipo** dei parametri, non dal valore di ritorno.

```
public static double area(double lato) { return lato * lato; }
public static double area(double b, double h) { return b * h; }
// area(5) usa il primo, area(4,3) il secondo
```

Soluzione 2

```
public Studente(String nome, double media) {
    this.nome = nome; this.media = media;
}
public Studente(String nome) {
    this(nome, 6.0);
}
```

Soluzione 3

```
public void preleva(double x) {
    if (x <= saldo) saldo -= x;
    else System.out.println("Saldo insufficiente");
}
public void versa(double x) { saldo += x; }
public void bonifico(Conto dest, double x) {
    if (x <= saldo) { preleva(x); dest.versa(x); }
}
```

Soluzione 4

```
public class Punto {
    private double x, y;
    public Punto(double x, double y) { this.x=x; this.y=y; }
    public double distanzaDa(Punto p) {
        double dx = this.x - p.x, dy = this.y - p.y;
        return Math.sqrt(dx*dx + dy*dy);
    }
}
```