

# Lezione 7 – Esercitazione

*prof. Marcello Sette*

<mailto://marcello.sette@gmail.com>

<http://sette.dnsalias.org>

## Esercizio 1

Determinare gli errori nei seguenti sorgenti Java.

### 1. File: Test1.java

```
public class Test1 {
    public static void main(String[] args) {
        System.out.println("Va tutto bene?");
    }
}
public class TestAnother1 {
    public static void main(String[] args) {
        System.out.println("Va tutto bene?");
    }
}
```

### 2. File: Test2.java

```
public class Testing2 {
    public static void main(String[] args) {
        System.out.println("Va tutto bene?");
    }
}
```

### 3. File: Test3.java

```
public class Test3 {
    public static void main(String args) {
        System.out.println("Va tutto bene?");
    }
}
```

### 4. File: Test4.java

```

public class Test4 {
    public void main(String[] args) {
        System.out.println("Va tutto bene?");
    }
}

```

## Esercizio 2

In questo esercizio si vuole mostrare l'uso dell'incapsulazione. Si dovrà creare una classe `Veicolo` che permetta al programma di funzionare.

Sono proposte due versioni diverse dello stesso esercizio.

### Versione 1

File: `TestVeicolo.java`

```

public class TestVeicolo {
    public static void main(String[] args) {

        // Crea un veicolo che possa caricare fino a 10000 kg
        System.out.println(
            "Creazione di un veicolo con 10000 kg di carico massimo.");
        Veicolo veicolo = new Veicolo(10000.0);

        // Aggiungi alcune scatole
        System.out.println("Aggiunta della scatola #1 (500kg)");
        veicolo.carico = veicolo.carico + 500.0;

        System.out.println("Aggiunta della scatola #2 (250kg)");
        veicolo.carico = veicolo.carico + 250.0;

        System.out.println("Aggiunta della scatola #3 (5000kg)");
        veicolo.carico = veicolo.carico + 5000.0;

        System.out.println("Aggiunta della scatola #4 (4000kg)");
        veicolo.carico = veicolo.carico + 4000.0;

        System.out.println("Aggiunta della scatola #5 (300kg)");
        veicolo.carico = veicolo.carico + 300.0;

        // Stampa il carico finale del veicolo
        System.out.println(
            "Carico finale del veicolo: " + veicolo.getCarico() + " kg");
    }
}

```

## Versione 2

File: TestVeicolo.java

```
public class TestVeicolo {
    public static void main(String[] args) {

        // Crea un veicolo che possa caricare fino a 10000 kg
        System.out.println(
            "Creazione di un veicolo con 10000 kg di carico massimo.");
        Veicolo veicolo = new Veicolo(10000.0);

        // Aggiungi alcune scatole
        System.out.println(
            "Aggiunta della scatola #1 (500kg) : " + veicolo.addScatola(500.0));
        System.out.println(
            "Aggiunta della scatola #2 (250kg) : " + veicolo.addScatola(250.0));
        System.out.println(
            "Aggiunta della scatola #3 (5000kg) : " + veicolo.addScatola(5000.0));
        System.out.println(
            "Aggiunta della scatola #4 (4000kg) : " + veicolo.addScatola(4000.0));
        System.out.println(
            "Aggiunta della scatola #5 (300kg) : " + veicolo.addScatola(300.0));

        // Stampa il carico finale del veicolo
        System.out.println(
            "Carico finale del veicolo: " + veicolo.getCarico() + " kg");
    }
}
```

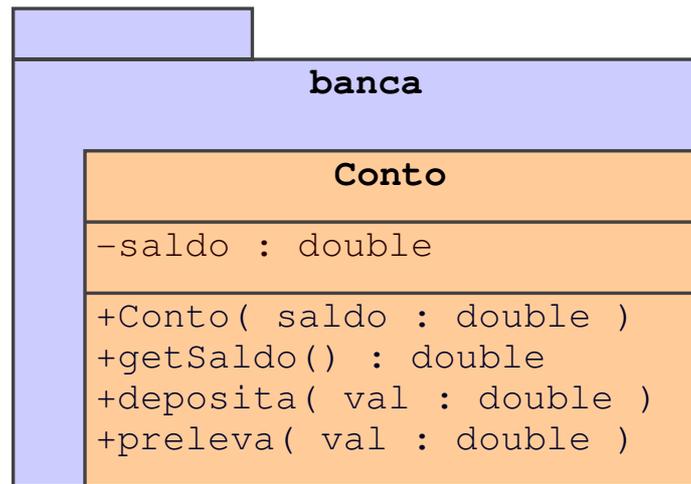
## Esercizio 3

### Obiettivo

Questo esercizio introdurrà la costruzione di un progetto di gestione bancaria che sarà riproposto e sviluppato ulteriormente nelle esercitazioni successive.

Il progetto consisterà (eventualmente) in una banca con alcuni clienti titolari di vari conti e destinatari di estratti conto.

In questo esercizio verrà chiesto di creare una semplice versione della classe **Conto** all'interno di un pacchetto **banca**. Nel pacchetto di default viene fornita una classe di test, **TestConto**, che crea un singolo conto, inizializza il saldo, esegue alcune semplici transazioni, mostra infine il saldo finale.



## Traccia

1. Posizionarsi all'interno della cartella corrispondente al pacchetto di default. In essa è presente il file `TestBanca.java`.
2. Creare la cartella corrispondente al pacchetto (`banca`).
3. Nella cartella `banca` creare la classe `Conto` all'interno del file `Conto.java`. La classe deve realizzare il modello del diagramma UML precedente.
  - Dichiarare un attributo privato: `saldo`.
  - Dichiarare un costruttore pubblico con parametro: `saldo`.
  - Dichiarare un metodo pubblico, `getSaldo`, che restituisca il valore del saldo corrente.
  - Dichiarare un metodo pubblico, `deposita`, che aggiunga una somma al saldo corrente.
  - Dichiarare un metodo pubblico, `preleva`, che sottragga una somma dal saldo corrente.
4. Nella cartella di default, compilare il file `TestBanca.java`. Verranno compilate a cascata tutte le classi usate nel programma. Nella cartella `banca`, controllare che sia stato compilato il file `Conto.java` in `Conto.class`. Il comando è

```
$ javac -d . TestBanca.java
```

5. Eseguire la classe `TestBanca`. Deve essere prodotto l'output seguente:

```
$ java TestBanca
Creazione di un conto con un saldo di 500.00
Prelievo 150.00
Deposito 22.50
Prelievo 47.62
Il conto ha attualmente un saldo di 324.88
```