



Esercizi: Primi passi...

- **Lettura dell'input da tastiera**
- **La classe Calcolatrice**
- **La classe Spese**
- **La classe TimeOfDay**

Esempio 1: Welcome

- Scrivere un programma che legge da tastiera il nome dell'utente, lo memorizza nella variabile *nome* e visualizza a video la scritta

Welcome, <nome>

- **Main problem**
 - Leggere l'input da tastiera
 - In JAVA non esiste una funzione (simile a quella per visualizzare un messaggio a video) per leggere l'input da tastiera
 - In C esiste scanf, in Pascal esiste read e readln

[Soluzione: Welcome]

```
package labIngegneria.esercizi;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;

public class Welcome {
    public static void main(String args[]) {
        // Allocazione input stream e buffer di lettura
        InputStreamReader input = new InputStreamReader(System.in);
        BufferedReader tastiera = new BufferedReader(input);

        // Allocazione variabile nome
        String nome = new String();

        // Lettura input da tastiera utilizzando il buffer di lettura
        try {
            nome = tastiera.readLine();
        } catch (IOException e) {
            System.exit(1);
        }

        // Visualizzazione messaggio
        System.out.println("Welcome, " + nome);
    }
}
```

Key points: Welcome - 1

■ Lettura input da tastiera

- Necessario creare un buffer di lettura a partire da uno stream di input

```
InputStreamReader inputStream = new InputStreamReader(System.in);  
BufferedReader reader = new BufferedReader(inputStream);
```

- Necessario catturare le eccezioni (errori di IO)

```
try {  
    // Operazioni critiche  
} catch (Exception e) {  
    // Gestione di eventuali errori  
}
```

Key points: Welcome - 2

- Utilizzando un buffer di lettura tutto ciò che viene letto è di tipo String
 - Necessaria una conversione per ottenere un valore numerico
 - Utilizzo dei metodi parseXXX delle classi wrapper

```
String s = "5";

int x = Integer.parseInt(s);
double y = Double.parseDouble(s);

System.out.println(x);
// System.out.println(String.valueOf(x));
```

Esercizio 1: Calcolatrice - 1

- Scrivere una classe Calcolatrice che simula il comportamento di una calcolatrice a soli due operandi
 - Attributi
 - I due operandi
 - Metodi (operazioni)
 - Somma: $a + b$
 - Differenza: $a - b$
 - Prodotto: $a * b$
 - Divisione: a / b
 - Attenzione a quando $b = 0$
 - Elevamento a potenza: a^b
 - Attenzione a quando $b < 0$ o $b = 0$

Esercizio 1: Calcolatrice - 2

- Scrivere un programma che permetta di utilizzare le funzioni messe a disposizione dalla classe Calcolatrice
 - Visualizzare un menu con le diverse operazioni
 - Gestire eventuali scelte errate
 - Il programma termina quando lo decide l'utente
 - Aggiungere una voce al menù per uscire dal programma

Esercizio 2: Gestione spese - 1

- **Creare una classe Spese che consente di gestire le spese giornaliere di n giorni**
 - **Attributi**
 - Array delle spese
 - Numero di spese giornaliere inserite (dimensione logica array)
 - **Metodi**
 - Inserimento nuova spesa giornaliera
 - Spesa giorno k
 - Spesa media
 - Spesa massima
 - Spesa minima
 - Elenco spese

Esercizio 2: Gestione spese - 2

- **Scrivere un programma che permetta di utilizzare le funzioni messe a disposizione dalla classe Spese**
 - Visualizzare un menu con le diverse operazioni
 - Gestire eventuali scelte errate
 - Il programma termina quando lo decide l'utente
 - Aggiungere una voce al menù per uscire dal programma

[**Esercizio 3: TimeOfDay**]

- **Realizzare una classe TimeOfDay che memorizza un istante di tempo compreso tra 00:00:00 e 23:59:59. Fornire un costruttore TimeOfDay(int hours, int minutes, int seconds) e metodi di accesso che restituiscano il valore di ore, minuti e secondi. Fornite i metodi:**
 - TimeOfDay addSeconds(int seconds)
 - int secondsFrom(TimeOfDay other)
- **Utilizzare 3 variabili intere per memorizzare le ore, i minuti e i secondi**