

# Le variabili e le costanti

---

- Una variabile è un'area di memoria identificata da un nome
- Il suo scopo è di contenere un valore di un certo tipo
- Serve per memorizzare dati durante l'esecuzione di un programma
- Il nome di una variabile è un **identificatore**
- può essere costituito da lettere, numeri e underscore
- non deve coincidere con una parola chiave del linguaggio
- è meglio scegliere un **identificatore** che sia **significativo** per il programma

## esempio

```
public class Triangolo {  
    public static void main ( String [] args ) {  
  
        int base , altezza ;  
        int area ;  
  
        base = 5;  
        altezza = 10;  
        area = base * altezza / 2;  
  
        System.out.println ( area );  
    }  
}
```

Usando le variabili il programma risulta essere **più chiaro**:

- Si capisce meglio quali siano la base e l'altezza del triangolo
- Si capisce meglio che cosa calcola il programma

---

## Dichiarazione

- In Java ogni variabile deve essere **dichiarata prima del suo uso**
- Nella dichiarazione di una variabile se ne specifica il **nome** e il **tipo**
- Nell'esempio, abbiamo dichiarato tre variabili con nomi base, altezza e area, tutte di tipo int (numeri interi)
  - int base , altezza ;
  - int area ;

**ATTENZIONE!** Ogni variabile deve essere dichiarata **UNA SOLA VOLTA** (la prima volta che compare nel programma)

```
base =5;  
altezza =10;
```

```
area = base * altezza /2;
```

---

## Assegnazione

- Si può memorizzare un valore in una variabile tramite l'operazione di assegnazione
- Il valore da assegnare a una variabile può essere un letterale o il risultato della valutazione di un'espressione
- Esempi:

```
base =5;  
altezza =10;  
area = base * altezza /2;
```

- I valori di base e altezza vengono letti e usati nell'espressione
- Il risultato dell'espressione viene scritto nella variabile area

---

## Dichiarazione + Assegnazione

Prima di poter essere usata in un'espressione una variabile deve:

- essere stata dichiarata
- essere stata assegnata almeno una volta (inizializzata)
- NB: **si possono combinare dichiarazione e assegnazione.**

Ad esempio:

```
int base = 5;  
int altezza = 10;  
int area = base * altezza / 2;
```

---

## Costanti

Nella dichiarazione delle variabili che **NON DEVONO** mai cambiare valore si può utilizzare il modificatore **final**

```
final double IVA = 0.22;
```

- Il modificatore **final** trasforma la variabile in una costante
- Il compilatore si occuperà di controllare che il valore delle costanti non venga **mai modificato** (ri-assegnato) dopo essere stato inizializzato.

- Aggiungere il modificatore **final** non cambia funzionamento programma, ma serve a prevenire errori di programmazione
  - Si chiede al compilatore di controllare che una variabile non venga ri-assegnata per sbaglio
  - Sapendo che una variabile non cambierà mai valore, il compilatore può anche eseguire delle **ottimizzazioni** sull'uso di tale variabile.
- 

## Input dall'utente

- Per ricevere valori in input dall'utente si può usare la classe Scanner, contenuta nel package **java.util**
- La classe Scanner deve essere richiamata usando la direttiva import prima dell'inizio del corpo della classe