

# Operatori

## Operatori aritmetici

- Di assegnazione: = += -= \*= /= &= |= ^=
- Di assegnazione/incremento: ++ -- %=
- Operatori Aritmetici: + - \* / %

Operatore	Significato
+	addizione
-	sottrazione

# Operatori di assegnazione

Operatore	Significato
=	addizione
+=	addizione assegnazione
-=	sottrazione assegnazione
*=	motiplicazione assegnazione
/=	divisione assegnazione
%=	resto assegnazione

# Operatori relazionali

== != > < >= <=

Operatore	Significato
<	minore di
<=	minore di o uguale a
>	maggiore di
>=	maggiore di o uguale a
==	uguale a
!=	non uguale / diverso

# Operatori per Booleani

- Bitwise (interi): & | ^ << >> ~

Operatore	Significato
&&	short circuit AND
	short circuit OR
!	NOT

## Attenzione:

- Gli operatori logici agiscono **solo su booleani**
  - Un intero NON viene considerato un booleano
  - Gli operatori relazionali forniscono valori booleani

# Operatori su reference

**Per i riferimenti/reference, sono definiti:**

- Gli operatori relazionali == e !=
  - test sul riferimento all'oggetto, **NON sull'oggetto**
- Le assegnazioni
- L'operatore "punto"
- NON è prevista l'aritmetica dei puntatori, vengono gestiti dalla JVM

# Operatori matematici

Operazioni matematiche complesse sono permesse dalla **classe Math** (package `java.lang`)

- `Math.sin (x)` calcola  $\sin(x)$
- `Math.sqrt (x)` calcola  $x^{(1/2)}$
- `Math.PI` ritorna `pi`
- `Math.abs (x)` calcola  $|x|$
- `Math.exp (x)` calcola  $e^x$
- `Math.pow (x, y)` calcola  $x^y$

## Esempio

- `z = Math.sin (x) - Math.PI / Math.sqrt(y)`

# Caratteri speciali

Literal	Represents
<code>\n</code>	New line
<code>\t</code>	Horizontal tab
<code>\b</code>	Backspace
<code>\r</code>	Carriage return
<code>\f</code>	Form feed
<code>\\</code>	Backslash
<code>\"</code>	Double quote
<code>\ddd</code>	Octal character