

## Tipus de dades

Tipus de dades		Paraula clau
Enters	Byte	byte
	Enter curt	short
	Enter simple	int
	Enter llarg	long
Reals	Real de precisió simple	float
	Real de precisió doble	double
Booleà		boolean
Caràcter		char
Cadena de caràcters		String

## Paraules reservades en Java

Paraules reservades de Java			
abstract	else	interface	super
assert	enum	<b>long</b>	switch
<b>boolean</b>	extends	native	synchronized
break	<b>false</b>	new	this
<b>byte</b>	<b>final</b>	<b>null</b>	throw
case	finally	<b>package</b>	transient
catch	<b>float</b>	private	<b>true</b>
<b>char</b>	for	protected	try
<b>class</b>	if	<b>public</b>	<b>void</b>
continue	implements	return	volatile
default	import	<b>short</b>	while
do	instanceof	<b>static</b>	
<b>double</b>	<b>int</b>	strictfp	

## Format de sortida

`System.out.format()`

`System.out.printf()`

## Entrada de dades (classe Scanner)

```
import java.util.Scanner;
```

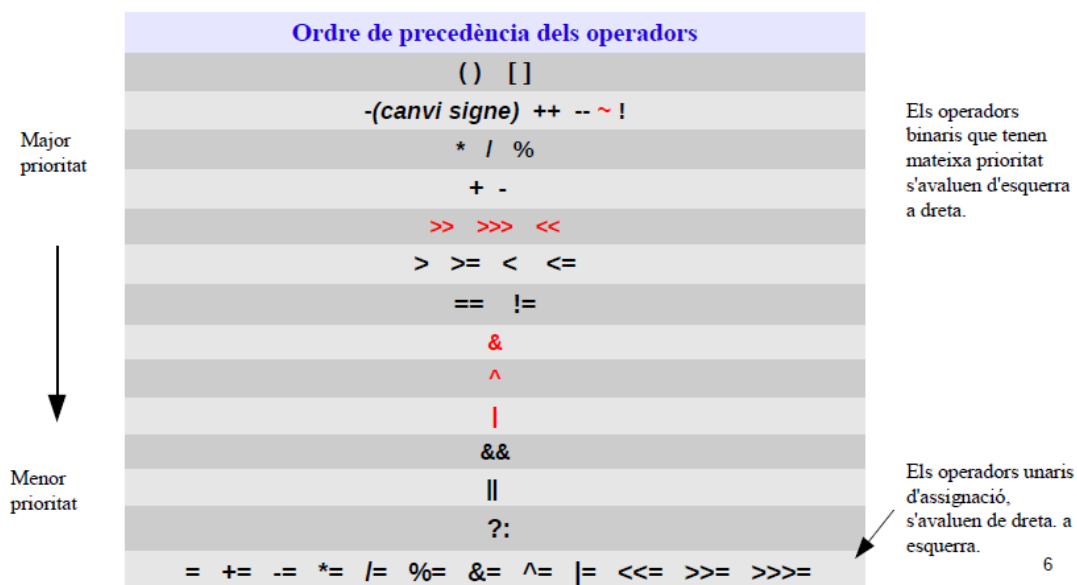
```
Scanner lector = new Scanner(System.in);
```

- **next():** recull una paraula i la torna amb tipus String.
- **nextLine():** recull una línia de text i la torna amb tipus String.
- **nextInt():** recull un nombre enter i el torna de tipus int;
- **nextDouble():** recull un nombre double i el torna de tipus double;
- ...

## Validació de dades (classe Scanner)

- **hasNextInt():** Comprova si el número introduït és enter
- **hasNextDouble():** Comprova si el número introduït és double
- **hasNextFloat():** Comprova si el número introduït és float
- ...

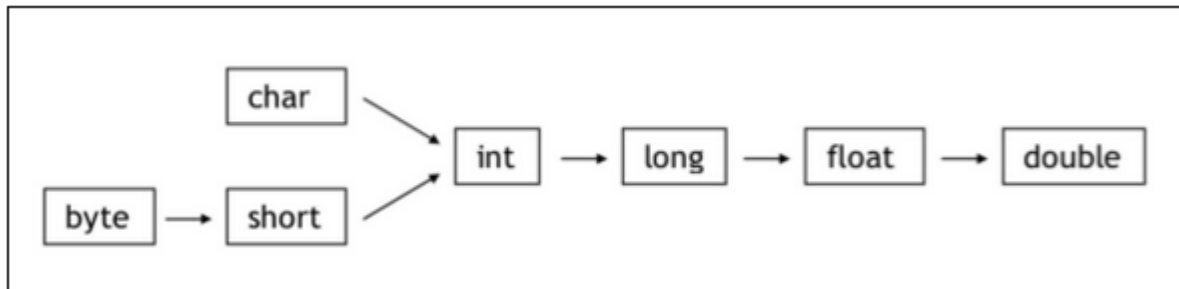
## Ordre de precedència dels operadors



## Operador ternari

(condició) ? retorn si cert : retorn si fals

## Conversió de tipus de dades Implícites



### Operacions entre tipus short o entre tipus byte

Quan fas una operació entre dos números byte o short el resultat es converteix automàticament a int.

## Conversió de tipus de dades Explícites

```
// Conversions explícites  
b = (byte) num;  
s = (short) num;  
i = (int) num;  
l = (long) num;  
f = (float) num;
```

## Estructures de selecció

```
if (condició) {  
    ...  
}
```

```
if (condició) {  
    ...  
}  
else {  
    ...  
}
```

```
if (condició) {  
    ...  
}  
else if (condició) {  
    ...  
}  
else {  
    ...  
}
```

```
switch (opció) {  
    case 1:  
        break;  
    //opcional  
    case 2:  
        ...  
    default: //opcional  
        ...  
}
```

## Estructures repetitives

```
while (condició) {  
    ...  
}
```

```
for (declaració i inicialització; condició, actualització) {  
    ...  
}
```

```
do {  
    ...  
} while (condició);
```

## Mètodes classe String

**int "String".length():** Retorna la quantitat de caràcters del String

**char charAt(int i):** Retorna el caràcter a l'índex i

**String substring(int i):** Retorna la subcadena des del i-èssim caràcter d'índex (inclòs), fins al final

**String substring (int i, int j):** Retorna la subcadena des de l'índex i a j-1, inclosos (l'índex j no està inclòs).

**int indexOf (String s):** Retorna l'índex dins de la cadena, de la primera aparició de la "subcadena" (o del caràcter) especificada.

**int indexOf (String s, int i):** Retorna l'índex dins de la cadena, de la primera aparició de la cadena (o el caràcter) especificada, començant a l'índex especificat.

**boolean equals (objecte otroObjeto):** Compara aquest String amb l'objecte especificat.

**boolean equalsIgnoreCase (String otroString):** Compara l'string amb un altre String, ignorant les consideracions de majúscules i minúscules.

**String toLowerCase():** Retorna una altra String on tots els caràcters són en minúscules.

**String toUpperCase():** Retorna una altra String on tots els caràcters són en majúscules.

**String replace (char oldChar, char newChar):** Retorna una nova cadena, resultant de substituir totes les ocurrències de oldChar amb newChar.

**String replaceAll(String text1, String text2):** Retorna una nova cadena resultant de substituir totes les ocurrències de la cadena text1 amb text2.

**String toCharArray():** Converteix una cadena de text en un array de caràcters.

exemple:

```
String sCadena = "Esto es una cadena de texto";  
char[] aCaracteres= sCadena.toCharArray();
```

## Classe Integer:

**int parseInt(String s[, int]):** Converteix una cadena de text en un nombre enter. Podem indicar el sistema numèric que representa la cadena, informant el paràmetre int 10, 2, 8 o 16 (decimal, binari, octal o hexadecimal).

**String toString(int):** Converteix un enter en un String.

Ex: String text = Integer.toString(i);

## Classe Double:

**int parseDouble(String s):** Converteix una cadena de text en un nombre de tipus double.

## Classe Random

### Mètodes

- **nextInt()** retorna un número enter positiu o negatiu dins de tot el rang dels enters ( $2^{32}$ ).
- **nextInt(int n)** retorna un número enter  $\geq 0$  i menor que n.
- **nextDouble()** retorna un número positiu de tipus double,  $\geq 0.0$  i menor que 1.0

En general, per generar enters a partir del valor dels dos límits **DESDE**, **HASTA**, ambdós inclosos:

```
Random rnd = new Random();  
num = rnd.nextInt(HASTA - DESDE + 1) + DESDE;
```