

## Módulo 01: SINTAXIS JAVA

Matemáticos	
Suma	+
Resta	-
División	/
Multiplicación	*
Módulo	% (devuelve el resto de una división entera)
Operador unario sumar 1	++
Operador unario restar 1	--

Operadores lógicos	
Y	&& (devuelve verdadero si las dos evaluaciones son verdaderas)
O	(devuelve verdadero si una de las dos evaluaciones son verdaderas)
No	! (devuelve lo opuesto al resultado de la evaluación)

Operadores relacionales	
Mayor	>
Menor	<
Igual	== o .equals()
Mayor o igual	>=
Menor o igual	<=
No igual	!=

Tipos de datos primitivos	
byte	Números enteros entre -128 y 127
short	Números enteros entre -32768, 32767
int	Números enteros entre -2147483648 y 2147483647
long	Enteros muy grandes, entre -9223372036854775808 y 9223372036854775807
float	Número con coma -3.402823e38 a 3.402823e38
double	Número con coma, mayor capacidad -1.79769313486232e308 a 1.79769313486232e308
string	Cadena de caracteres
char	Un carácter (Ej: 'a') Unicode
boolean	Verdadero o falso (true /false)



# Estructuras

## Estructura de decisión:

```
if (condición){  
    //código que se corre si la condición es  
    verdadera
```

```
} else if(condición){  
    //código que se corre si la primera  
    condición no fue verdadera y la segunda  
    sí es verdadera
```

```
}else {  
    // código que se corre si ninguna  
    condición anterior fue verdadera  
}
```

---

```
switch (variable){  
    case valor1:  
        //código que se ejecuta si la  
        variable tiene valor1  
        break;  
    case valor2:  
        //código que se ejecuta si la variable  
        tiene valor2  
        break;  
    .  
    .  
    default:  
        //código que se ejecuta si la  
        variable tiene algún valor no  
        enumerado  
}
```

---

## Estructuras de repetición:

```
for(Integer i = 0; i < valorMaximo;  
i++){  
    //código que se ejecuta cada vez  
}
```

---

```
for(Object object : listaDeObjetos){  
    //código que se va a ejecutar por cada objeto  
    en la lista  
}
```

---

```
while(condición){  
    //hacer este código  
}
```