

## CURSO DE FUNDAMENTOS DE JAVA

# 3. SENTENCIAS DE CONTROL



Ing. Ubaldo Acosta

Por el experto: Ing. Ubaldo Acosta



## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Hola, te saluda nuevamente Ubaldo Acosta. Espero que estés listo para comenzar con esta lección..

Vamos a estudiar el tema de sentencias de control en Java, así como la forma en que aplicaremos el lenguaje de programación Java para resolver los problemas que nos planteemos en este curso.

¿Estás listo? ¡Vamos!

## DECISIONES

¿OPCIÓN  
A?

¿OPCIÓN  
B?

¿CAMINO  
CORTO?

¿CAMINO  
LARGO?



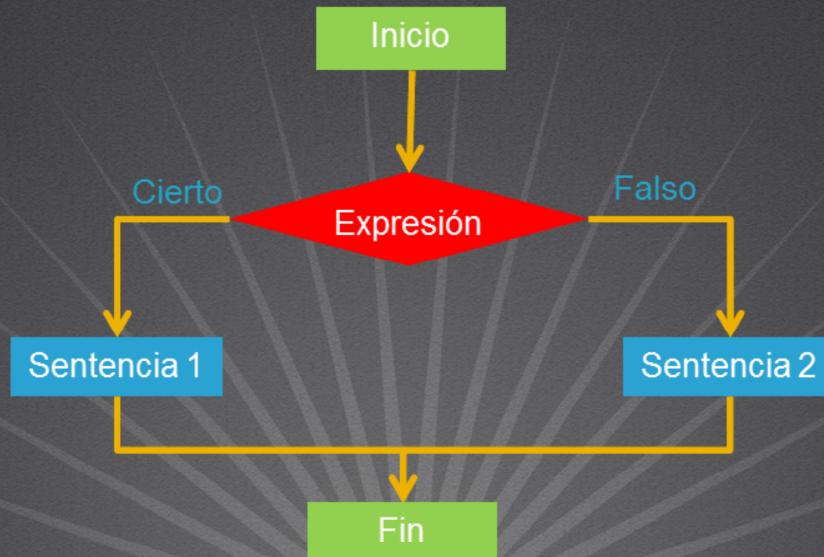
CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

A lo largo de nuestra vida, vivimos tomando decisiones. Decisiones como ¿Cuál carrera estudiar? ¿Qué trabajo elegir? ¿Casarme o no casarme? ¿Tener hijos? ¿Comenzar a ahorrar a temprana edad? Entre muchas decisiones más.

De la misma manera, cuando estamos resolviendo un problema computacional, nos enfrentamos a la toma de decisiones, y esto es lo que estudiaremos en esta lección.

## IF / ELSE



### CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Las decisiones existen en cualquier lenguaje de programación moderno, ya que son la base para el flujo que definimos en la solución de un problema.

Para manejar este concepto vamos a utilizar la estructura de control if. Con esto haremos la pregunta: si esto es verdadero, entonces haz esto.

También podemos utilizar la palabra else para que en caso de que la pregunta no sea verdadera, entonces haga algo en caso contrario.

La estructura básica se muestra en la figura.

La estructura if/else puede combinarse para generar casos más complejos. Por ejemplo utilizando if, else if, else if, else. La idea es poder agregar decisiones según los casos que se nos presenten al momento de resolver un problema.

También, la estructura if/else puede tener sentencias anidadas, es decir, un if dentro de otro.

Esto lo iremos poniendo en práctica conforme avancemos en el curso.

## SINTAXIS IF / ELSE

Valor tipo  
**boolean**

No lleva punto  
y coma

```
if( condicion ){
    //Sentencias a ejecutar si es verdadero
}
else{
    //Setencias a ejecutar si es falso
}
```

Las sentencias  
si llevan punto y  
coma

el **else** es  
opcional

CURSOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

En la figura podemos observar la sintaxis que se utiliza en Java al momento de utilizar la estructura if /else.

Para iniciar se evalúa una condición. Si esta es verdadera entonces se ejecutan las instrucciones contenidas en ese bloque de código. Un bloque de código en la estructura if puede llevar corchetes o no ( { ... } ), esto con el fin de delimitar las sentencias o instrucciones que debe ejecutar el if. En el caso de que no se usen corchetes, entonces sólo se podrá agregar una instrucción después de la palabra if, como sigue a continuación:

```
if( condicion )
    //Sólo una sentencia si no se usan corchetes
else
    //Sólo una sentencia si no se usan corchetes
```

Además, la sección del else es opcional, pudiendo tener solo la parte del if para ejecutar la lógica deseada.

Podemos observar en la figura la sintaxis más común al momento de utilizar la estructura if/else, y más adelante haremos algunos ejercicios para poner en práctica esta sintaxis.

## SINTAXIS IF / ELSE IF / ELSE

```
if( condicion1 ){
    //Sentencias a ejecutar si es verdadero
}
else if( condicion2 ){
    //Sencencias a ejecuta rsi es
}
//Mas else-if si es necesario
else{
    //Setencias a ejecutar si es falso
}
```

Valor tipo  
**boolean**

Puede haber  
muchos **else-if**

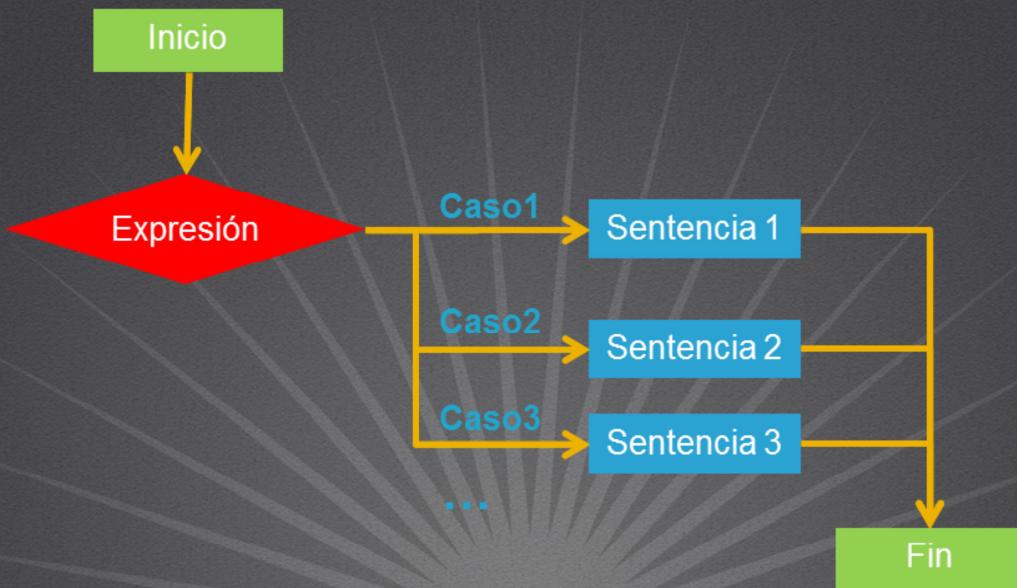
el **else** es  
opcional

También es posible utilizar la estructura if - else if – else. Esta estructura nos permite agregar más condiciones según sea necesario, utilizando else if.

Podemos observar que a diferencia de la estructura if/else, esta estructura puede agregar cualquier cantidad de else if agregando una nueva condición a evaluar. Y si esta condición es verdadera entonces se ejecutan las sentencias que incluye ese bloque de código. Solo se ejecutará una de ellas, es decir que en cuando se encuentre la primera condición que es verdadera, sólo se ejecutan las instrucciones de ese bloque de código y ya no evalúa las siguientes condiciones, ni tampoco las sentencias de else, si es que existiera.

Veremos a continuación que en algunos casos es más sencillo utilizar la estructura switch precisamente para no tener tanto else-if ya que es una sintaxis más compleja, comparada con la estructura switch.

# SWITCH



## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Otra forma de tomar decisiones es utilizando la estructura de control switch, que como su nombre lo indica, es como utilizar un interruptor.

A diferencia de la estructura if/else, la estructura switch la vamos a utilizar más cuando tenemos por ejemplo un menu de opciones. Por ejemplo, si tenemos 5 opciones, y cada una de ellas realiza una acción distinta, entonces, la estructura switch es ideal para manejar estas decisiones.

La estructura switch es menos utilizada que la estructura de control if/else, sin embargo nos servirá para cuestiones más específicas a la hora de tomar decisiones como veremos en el siguiente ejemplo.

Valores **byte, short, int, char o String**

## SINTAXIS SWITCH

```
switch (expresion) {
    case valor1:
        //Sentencias
        break;
    case valor2:
        //Setencias
        break;
    case valorN:
        //Setencias
        break;
    default:
        //Setencias
}
```

Cada case termina con dos puntos

Las sentencias si llevan punto y coma

el **default** es opcional

Podemos observar la sintaxis de la estructura switch. En las versiones anteriores al jdk 7, sólo se soportaba valores en la expresión a evaluar de tipo byte, short, int o char, y de la versión jdk 7 en adelante, se soporta también el tipo String, que aunque no es un tipo primitivo, es uno de los tipos Object que más utilizaremos.

Una vez que comienza la evaluación de cada uno de los casos (case), pueden incluirse uno o más casos separados por coma. Ej.

```
switch (expresion) {
    case valor1: case valor2: case valor3:
        //Sentencias
        break;
    case valorN:
        //Setencias
        break;
    default:
        //Setencias
}
```

La palabra break se utiliza para que el flujo se detenga y termine la evaluación de los demás casos, si no lo colocamos se seguirá la evaluación de cada caso. Por último, la palabra default se utiliza para ejecutar un bloque de código, si ninguno de los casos coincidió en su evaluación.

# OPERADORES LÓGICOS

Operador	Símbolo	Significado
AND	&&	AND corto circuito
OR		OR corto circuito
AND	&	AND lógico
OR		OR lógico
XOR	^	OR exclusivo
NOT	!	Negación

## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Además de los operadores ya mencionados, utilizaremos operadores lógicos para que en combinación con las estructuras de control podamos crear decisiones que brinden una solución a nuestras necesidades.

En la tabla pueden observar los operadores lógicos con los que vamos a estar trabajando.

A menos que deseemos trabajar con bits (unos y ceros), normalmente utilizaremos los operadores AND y OR en modo corto circuito. En la siguiente tabla explicaremos qué significa esto.

# TABLA DE VERDAD DE OPERADORES LÓGICOS

X	Y	X&&Y	X   Y	X^Y	!X
0	0	0	0	0	0
0	1	0	1	1	1
1	0	0	1	1	1
1	1	1	1	0	0

## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

En la figura podemos observar la aplicación de los operadores lógicos y la tabla de verdad que aplica a cada operador. Esta tabla debemos conocerla de memoria ya que es la que estaremos aplicando cuando utilicemos los operadores lógicos. El 1 significa verdadero y el 0 significa falso. Sin embargo no es difícil de memorizar. Lo único que tenemos que aplicar es lo siguiente.

En el caso del operador and (`&&`), el resultado será verdadero únicamente si ambos valores son verdaderos, de lo contrario el resultado final será falso.

En el caso del operador or (`||`), el resultado será verdadero si cualquiera de los operandos (x o y) es verdadero.

Anteriormente mencionamos los operadores AND y OR corto circuito. Lo que significa es que si se detecta que el resultado ya no tiene sentido evaluarlo, por ejemplo si el operador `&&` la evaluación del operando del lado izquierdo es falso, entonces toda la expresión será falsa y ya no se evalúa el operando restante. Y en el caso del operador `||` si el operando del lado izquierdo entonces el resultado es verdadero y por lo tanto ya no se evalúa el otro operando del lado derecho.

Y en el caso de operador de negación lo único que hace es invertir el valor original del operando. Así que será muy sencillo aplicar esta tabla de verdad por cada uno de los operadores si aplicamos las reglas descritas. En los próximos ejercicios pondremos en práctica este tipo de operadores en combinación con las estructuras de control.

# OPERADORES RELACIONALES

Operador	Acción
>	Mayor que
>=	Mayor o igual
<	Menor que
<=	Menor o igual
==	Igual
!=	Distinto

## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Al utilizar las estructuras de decisión, nos encontraremos con la necesidad de comparar valores. Es por ello que utilizaremos distintos tipos de operadores para esto.

Los operadores relacionales se enlistan en la lámina. Y los pondremos en práctica en combinación con las estructuras de decisión en los siguientes ejercicios.

## EJERCICIO CURSO FUNDAMENTOS DE JAVA

- **ABRIR LOS ARCHIVOS DE EJERCICIOS EN PDF.**
- **EJERCICIO 9:** Ejercicio Manejo de if/else con Java
- **EJERCICIO 10:** Ejemplo de if con Java
- **EJERCICIO 11:** Ejemplo switch con Java



**CURSO DE FUNDAMENTOS DE JAVA**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

## CURSO ONLINE

# FUNDAMENTOS DE JAVA

Por: Ing. Ubaldo Acosta



Experiencia y Conocimiento para tu vida

## CURSO DE FUNDAMENTOS DE JAVA

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

En Global Mentoring promovemos la Pasión por la Tecnología Java. Te invitamos a visitar nuestro sitio Web donde encontrarás cursos Java Online desde Niveles Básicos, Intermedios y Avanzados, y así te conviertas en un experto programador Java.

Además agregamos nuevos cursos para que continúes con tu preparación como programador Java profesional. A continuación te presentamos nuestro listado de cursos:

- |                          |  |
|--------------------------|--|
| ✓ Lógica de Programación | ✓ Hibernate Framework                    |
| ✓ Fundamentos de Java    | ✓ Spring Framework                       |
| ✓ Programación con Java  | ✓ JavaServer Faces                       |
| ✓ Java con JDBC          | ✓ Java EE (EJB, JPA y Web Services)      |
| ✓ HTML, CSS y JavaScript | ✓ JBoss Administration                   |
| ✓ Servlets y JSP's       | ✓ Android con Java                       |
| ✓ Struts Framework       | <u>Datos de Contacto:</u> ✓ HTML5 y CSS3 |

Sitio Web: [www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Email: [informes@globalmentoring.com.mx](mailto:informes@globalmentoring.com.mx)