



Flujo, ciclos y métodos

Manejo básico de flujo

***Estructurar un algoritmo
lógico utilizando estructuras
de control y expresiones
para dar solución a un
problema de baja
complejidad acorde al
lenguaje Java***

- Unidad 1: Flujo, ciclos y métodos
- Unidad 2: Arreglos y archivos
- Unidad 3: Programación orientada a objetos
- Unidad 4: Pruebas unitarias y TDD



Te encuentras aquí



¿Qué aprenderás en esta sesión?

- *Construir algoritmos que tomen decisiones en base al valor de una variable para controlar de mejor manera el flujo y la lógica del programa.*
- *Aplicar los operadores de comparación y lógicos para comprobar si se cumple una determinada condición.*

¿Cómo podemos
tomar decisiones en
base a valores?



/* Recordando diagrama de flujo */

Diagrama de flujo

Analizamos

El rombo corresponde a una evaluación condicional. En este punto se toma una decisión y el programa continúa su flujo en función de esa decisión.

A continuación, aprenderemos a escribir programas que implementen estas decisiones.



/* La instrucción IF */

IF

Java y muchos otros lenguajes de programación tienen instrucciones para implementar condiciones. La más utilizada es la instrucción IF.

```
if(condicion){  
    //código que se ejecutará solo si se cumple la condición  
}
```

Lo anterior se lee como: "Si se cumple la condición, entonces ejecuta el código".

Ejemplo de IF

Crearemos un programa que pregunte la edad al usuario. Si la edad es mayor o igual a 18, entonces le diremos que es mayor de edad.

```
System.out.println("¿Qué edad tienes?");  
int edad = sc.nextInt();  
if(edad >= 18) {  
    System.out.println("Eres mayor de edad");  
}
```

Si ejecutamos el programa e introducimos un valor mayor o igual a 18 veremos el mensaje "Eres mayor de edad", en caso contrario, no veremos ningún mensaje.

/* Operadores de comparación */

¿Qué son los operadores de comparación?

Son aquellos que comparan dos valores y obtienen como resultado true (verdadero) o false (falso). A este tipo de objeto, resultado de una comparación, se le conoce como booleano.

Operador	Nombre	Ejemplo	Resultado
==	igual a	2 == 2	true
!=	distinto a	2 != 2	false
>	mayor a	3 < 4	false
>=	mayor o igual a	3 >= 3	true
<	menor a	4 < 3	false
<=	menor o igual a	3 <= 4	true

Ejemplo

Realicemos una prueba donde el usuario ingrese los valores y veamos si el primero es mayor que el segundo.

```
int a = sc.nextInt(); //1
int b = sc.nextInt(); //6
System.out.println(a > b); //false
```

Operadores de comparación en String

Aunque en la tabla solo hayamos mostrado números, podemos comparar dos objetos utilizando un operador de comparación.

```
String a = "texto 1";  
String b = "texto 2";  
System.out.println(a == b); //false
```

Operadores de comparación en String

¿Puede ser un texto mayor que otro?

La respuesta es No.

Los objetos (como lo es los Strings) no tienen implementado el operador < o >.

Para comparar variables de tipo String, tenemos básicamente tres métodos de la clase String que nos permiten evaluar o hacer comparaciones entre dos String. Las más utilizadas son las siguientes:

- equals
- equalsIgnoreCase
- compareTo

Método equals

El método equals permite evaluar si dos objetos String son o no iguales. De ser iguales, este devuelve true o de lo contrario devuelve false:

```
String s1 = "Hola"; //s1, objeto String que contiene la cadena Hola
String s2 = "Hola"; //s2, objeto String que contiene la cadena Hola
System.out.println(s1.equals(s2)); //true
```

Este método evalúa exactamente que los dos objetos, es decir, s1 y s2 sean iguales, por lo que si agregamos un mínimo cambio en uno de los dos objetos, por ejemplo, una letra de minúscula a mayúscula, el resultado será **false**.

```
//Se agrega la última letra en mayúscula
String s1 = "HolA"; //s1, objeto String que contiene la cadena HolA
String s2 = "Hola"; //s2, objeto String que contiene la cadena Hola
System.out.println(s1.equals(s2)); //false
```

Método equalsIgnoreCase

El método equalsIgnoreCase es similar a equals, con la diferencia de que este ignora las letras mayúsculas y minúsculas, por lo que el ejemplo anterior, utilizando este método, dará como resultado true.

```
String s1 = "Hola"; //s1, objeto String que contiene la cadena Hola
String s2 = "HoIA"; //s2, objeto String que contiene la cadena HoIA
boolean sonIgualesIgnoreCase = s1.equalsIgnoreCase(s2)
System.out.println("Las palabras son iguales sin tomar en cuenta los
caracteres en mayúsculas"); //true
```


Método compareTo

Este método permite comparar dos objetos de tipo String y dar como resultado si son iguales, si uno es menor que otro o bien cuál de los dos es mayor.

compareTo retorna:

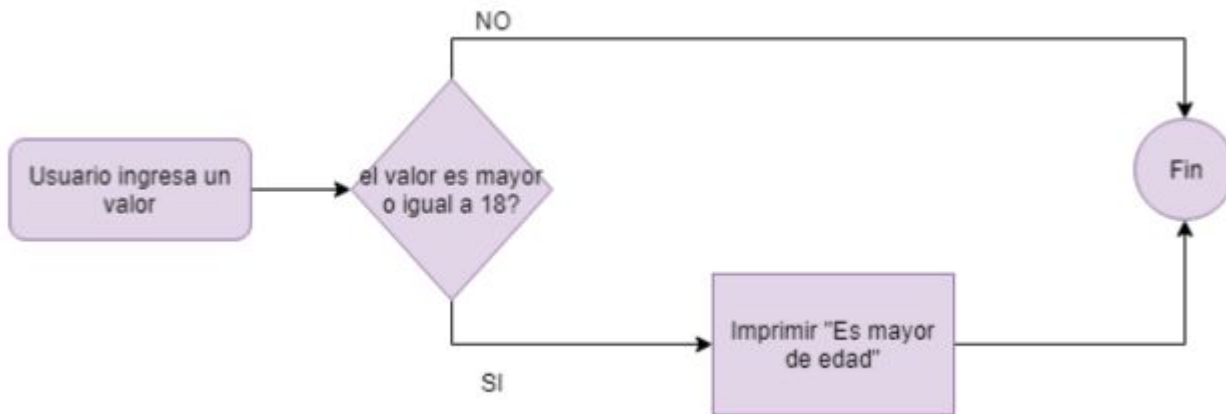
- 0 si los String son iguales
- -1 si el primero es menor
- 1 si el primero es mayor

```
String uno = "Hola";  
String dos = "Chao";  
if(uno.compareTo(dos) < 0) {  
    System.out.println("uno es menor");  
}
```

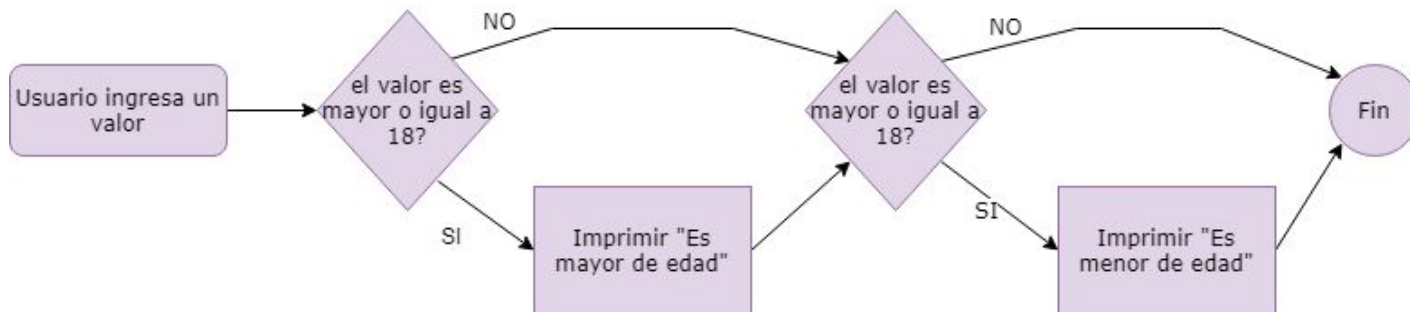
No podemos decir si un string es mayor en realidad que otro, sino que lo que se compara es carácter a carácter, en los caracteres ASCII.

Retomando el Ejemplo de IF

Ahora que ya conocemos los operadores de comparación, volvamos a nuestro código anterior y analicemos el diagrama de flujo.



En caso contrario



```
System.out.println("¿Qué edad tienes?");
int edad = sc.nextInt();
if(edad >= 18) {
    System.out.println("Eres mayor de edad");
}
if(edad < 18) {
    System.out.println("Eres menor de edad");
}
```

Ejercicio guiado



Comparar dos cadenas de texto

Requerimientos

- Comparar dos cadenas de texto.
 - Si son iguales, imprimimos en pantalla que ambas cadenas de texto son iguales.
 - Si no, imprimimos que son distintas.



Lo primero que debemos hacer es declarar las variables que almacenarán estas cadenas de texto:

```
String cadenaUno = "Bienvenido a Desafío Latam";  
String cadenaDos = "Bienvenidos a Desafío Latam";
```

Luego tenemos que comparar ambas cadenas e imprimir el texto según se cumpla la condición.

```
if(cadenaUno.equals(cadenaDos)){  
    System.out.println("Las cadenas son iguales");  
}else{  
    System.out.println("Las cadenas no son iguales");  
}
```



Ejercicio propuesto



Requerimientos

- Declarar dos cadenas de texto.
- Comparar las cadenas de texto con **equals**, y mostrar un mensaje en pantalla según sea el caso.
- Comparar las cadenas de texto con **compareTo** y ver cuál es mayor, menor o iguales.



¿Qué hacen los
operadores `==` y `>=`?





Próxima sesión...

- *Utilizar las condiciones para evaluar los datos contenidos en variables.*
- *Resolver algoritmos de una forma más eficaz mediante las estructuras de control de flujo.*

{desafío}
latam_

*Academia de
talentos digitales*

