

Java Standard Web Programming

Módulo 3

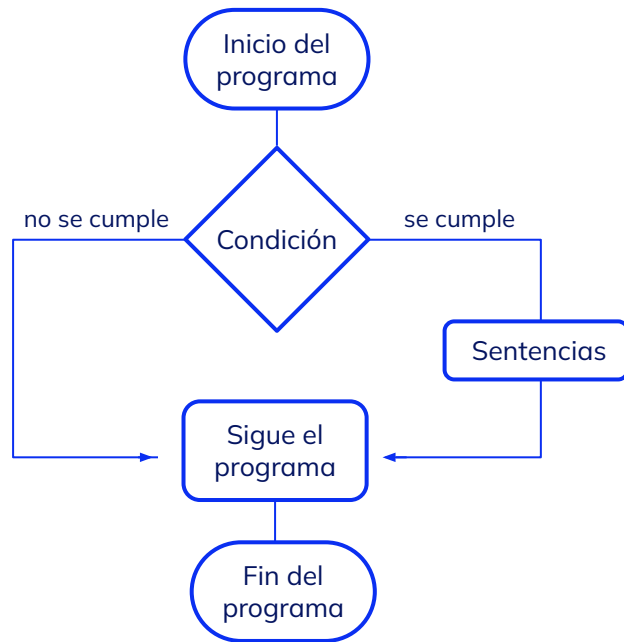


ForEach

Introducción a ForEach

Originalmente, Java no ofrecía un bucle de estilo **for-each** (por cada uno). Sin embargo, con el lanzamiento de JDK 5, el **bucle for se mejoró** para proporcionar esta opción. Es también conocido como el **bucle for extendido**.

Aunque se le conozca como **for-each** en otros lenguajes de programación en Java su sintaxis no posee la palabra **each**.



Sintaxis

for común

Para recorrer un arreglo usábamos el **for**, que vimos en clases anteriores, donde se declaraba una variable para poder iterar cada uno de los elementos.



```
// con llaves
for (inicio; condicion; despuesSentencias) {
    sentencia1;
    sentencia2;
}

// sin llaves
for (inicio; condicion; despuesSentencias)
    sentenciaUnica;
```

foreach

La palabra reservada **for**, seguida de **paréntesis que encierran la condición** (Tipo de elemento con su identificador : arreglo o colección) y las llaves que encerrarán el grupo de instrucciones a efectuar por cada iteración.

```
//con llaves
for (tipoDato tipoDato : arreglo) {
    sentencias1;
    sentencias2;
}

//sin llaves
for (tipoDato tipoDato : arreglo)
    sentenciaUnica;
```

Funcionamiento

Este bucle recorre un arreglo de principio a fin de forma secuencial:

```
String[] nombresAlumnos = { "Octavio", "Lautaro", "Daniela", "Analia" };  
  
for (String nombre : nombresAlumnos) {  
    System.out.println("El nombre del Alumno es: " + nombre);  
}
```

Características

- Como **recorre completamente el arreglo o colección**, la única forma de salir de él, antes de terminar de pasar por cada uno de los elementos, es con **break**.
- Al no tener un índice o valor por defecto **no funciona para modificar datos dentro del arreglo**, a menos que usemos un contador dentro del for por lo que no tendría lógica usarlo.
- Es ideal para **mostrar datos de un arreglo o colección**, y se lee mucho más fácil.
- Una vez que se comprende, es más **simple y cómodo** de trabajar.



**¡Sigamos
trabajando!**

