

Concepto de variable estática y concepto de Final

Fernando González

28 de septiembre de 2018

En Java, como en todos los lenguajes de programación podemos declarar variables, las cuales también son palabras clave en los lenguajes. En este caso hablaremos sobre la variable estática y a variable (*keyword*) final. Una variable estática es una variable que puede ser declarada dentro de una clase o una función y que se mantiene con vida durante toda la ejecución del programa. Cuando declaramos una variable estática en una clase, todos los objetos que sean instancia de esa clase, usaran la misma variable de miembro estática. Por lo tanto, esta variable declarada en una clase, siempre utilizara la misma posición de memoria independientemente del numero de instancias que creamos a esa clase. Estas variables mantienen su valor a lo largo de toda la ejecución del programa, y por consiguiente, si una variable estática esta declarada fuera de las funciones, sera accesible únicamente por el código que le sigan en la misma instancia de su declaración. Por otra lado, si una variable estática está declarada en una función, esta solo sera accesible desde esa función y mantendrá su valor entre ejecuciones de la función. Es importante saber que a pesar de que estas variables se declaran en el mismo lugar que el resto de variables en una función, estas al adquirir valores nuevos con cada ejecución, conservan estos valores entre ejecuciones. En la *Figura 1*, se demuestra un ejemplo de una función estática.

La variable *Final* es utilizada para indicar que una variable es de tipo constante, es decir, que no admitirá cambios después de su declaración y asignación de valor. Esta variable determina que un atributo no puede ser sobrescrito o re-definido, como ya antes dicho, esta variable no funciona como una variable normal, sino como una variable constante. Un ejemplo de una variable *final* es π . Final solo puede ser asignada una vez. Algunas datos importantes de esta variable es que una clase de tipo final no puede ser sub-clase y, que es una variable que no necesita ser inicializada. Un ejemplo de una variable final se puede visualizar en la *Figura 2*. Para concluir, podemos decir que ambas variables son parecidas dado a que ambas conservan el valor que fue por ultima vez asignado, pero dado los conceptos y las características de cada uno, podemos concluir que ambas son diferentes y que cumplen con funciones especificas diferentes. La variable estática puede cambiar su valor si en una función va incrementando o variando esta variable, pero esta variable conserva el valor hasta terminar ejecución del programa. En cambio, la variable final no puede ser modificada en ningún momento, como ya antes dicho, es una variable constante.

Referencias

Geeks, G. f. (2017). Geeks for Geeks. Retrieved from <https://www.geeksforgeeks.org/final-keyword-java/>

Madrid, U. I. (n.d.). Arquitectura de Sistemas. Retrieved from <http://www.it.uc3m.es/pbasant>

Rodriguez, A.(2018(consultado)).Aprendeaprogramar.Retrievedfrom<https://aprenderaprogramar.com/contentview=articleid=639:static-final-en-java-palabras-clave-variables-de-clase-o-campos-estaticos-y-constantes-ejemplos-cu00673bcatid=68:curso-aprender-programacion-java-desde-ceroItemid=188>

Rodriguez, F.(n.d.).AulaMacedonia.Retrievedfrom<http://macedoniamagazine.frodrig.com/poo8.htm>

Figuras

```
public class Ex {  
  
    public static void main(String[] args){  
  
        Hello();  
  
    }  
  
    private static void Hello(){  
  
        System.out.println("Hello World!");  
  
    }  
  
}
```

Figura 1: Ejemplo de una clase estática

```
public class Sphere {  
  
    public static final double PI = 3.141592653589793;  
  
    public final double radius;  
    public final double xPos;  
    public final double yPos;  
    public final double zPos;  
  
    Sphere(double x, double y, double z, double r) {  
        radius = r;  
        xPos = x;  
        yPos = y;  
        zPos = z;  
    }  
  
}
```

Figura 2: Ejemplo de una variable final