Universidad Autónoma del Estado de Hidalgo

Área Académica de Computación y Electrónica
Licenciatura en Ciencias Computacionales
Programación Orientada a Objetos

Tema: palabras clave **final** y **static**

Elaboración: M.C.C. Iliana Castillo Pérez

Fecha de elaboración: Marzo/2019.

ICBI

Tema
Palabra clave **final**

Clases finales

El lenguaje Java permite aplicar la palabra clave final a las clases.

Cuando se hace, no es posible generar subclases a partir de dicha clase.

Ejemplo:

La clase *String* es una clase final y se ha declarado así por razones de seguridad garantizando que si un método hace referencia a una cadena, será una cadena definida de la clase *String* y no una cadena de una posible subclase modificada.

Tema
Palabra clave **final**

Métodos final

Los métodos marcados con la palabra final no pueden sobrescribirse.

Por razones de seguridad, un método debe señalarse como final si contiene una implementación que no debería modificarse y es fundamental para mantener la coherencia del objeto.

Tema
Palabra clave **final**

Variables final

Las variables marcadas como final se convierten en constantes.

Cualquier intento por cambiar el valor en este tipo de variables provoca un error de compilación.

Ejemplo de su declaración:

```
Private static final double SALARIO_BASE = 15000.00;
```

Tema
Palabra clave **final**

Variables final

Si se marca como final a una variable con un tipo de referencia (es decir, cualquier tipo de clase), esa variable no podrá hacer referencia a ningún otro objeto. No obstante, sí es posible cambiar el contenido del objeto, porque sólo es final la referencia en sí.

Tema
Palabra clave **final**

Variables final vacías

Una variable final vacía es aquella que no se inicializa en su declaración. La inicialización se pospone. Este tipo de variables deben asignarse en un constructor, pero sólo pueden definirse una vez.

Si la variable final vacía es local, puede definirse en cualquier momento en el cuerpo del método, pero sólo puede definirse una vez.

Tema
Palabra clave **final**

Ejemplo:

```
public class Customer{
   private final long customerID;

   public Customer(){
       customerID = createID();
   }

    public long getID(){
      return customerID;
    }

    private long createID(){
      return … //algoritmo que genera un nuevo ID
    }
…  // otras sentencias
}
```

Tema
Palabra clave **static**

La palabra **static** declara miembros (atributos, métodos y clases) que están asociados a una clase en vez de a una instancia de la clase.

Atributos de clase

A veces resulta útil tener una variable compartida por todas las instancias de la clase. Esta variable podría utilizarse, por ejemplo, como base para la comunicación entre instancias o para llevar el control del número de instancias que se han creado.

Tema
Palabra clave **static**

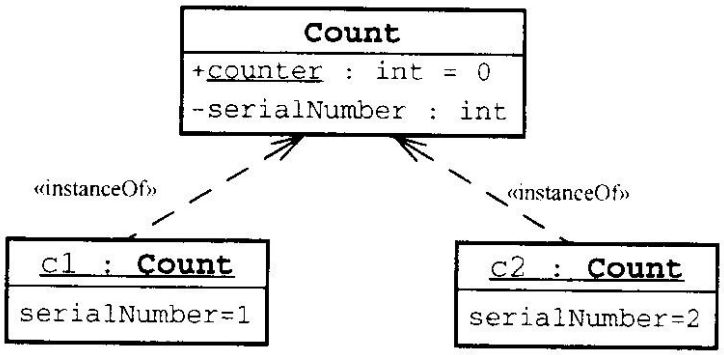

Diagrama UML de la Clase Count con dos instancias.
Imagen tomada de (Sun Microsystems, 2008).

El modo compartido se consigue marcando la variable con la palabra static. Este tipo de variables a veces se denominan variables de clase para distinguirlas de los miembros o variables de instancia, que no se comparten.

Tema
Palabra clave **static**

Ejemplo:

```
public class Counter{
   private int serialNumber;
   public static int counter = 0;

   public Counter(){
        counter++;
        serialNumber = counter;
   }
}
```

En este ejemplo, a cada objeto que se crea se le asigna un número exclusivo que inicia en 1 y continúa ascendentemente. Todas las instancias comparten la variable counter.

Referencias

Sun Microsystems (2008), El lenguaje de programación Java, Sun Microsystems.

Datos de contacto

M.C.C. Iliana Castillo Pérez
Profesora – Investigadora
Grupo de Investigación de Tecnología Computacional Educativa
Instituto de Ciencias Básicas e Ingeniería
Universidad Autónoma del Estado de Hidalgo
Área Académica de Computación y Electrónica
Correo: ilianac@uaeh.edu.mx