



UNIVERSIDAD DEL BÍO-BÍO

# Paradigmas de la Programación

Palabra clave final

# Uso de final en atributos

Cuando un atributo es declarado final, significa que solo tendrá como valor lo primero que se le asigne, después de eso arrojará error en tiempo de compilación si se intenta cambiar su valor.

En el caso de los atributos el valor debe ser asignado directamente en su declaración o en el constructor de la clase. Si no se asigna ningún valor se indicará error en tiempo de compilación.

**Observación:** En caso de las instancias, si es de tipo final no se podrá cambiar la referencia, pero si es posible cambiar su contenido interno.

```
public class Proceso {  
  
    public final int numero;  
    public final Estado estado;  
  
}
```

```
public class Proceso {  
  
    public final int numero = 10; //Asignando valor directamente  
    public final Estado estado;  
  
    public Proceso() {  
        estado = new Estado();  
    }  
  
}
```

# Uso de final en variables locales

Una variable final solo se puede inicializar una vez, ya sea a través de un inicializador o una declaración de asignación. No es necesario inicializarlo en el punto de la declaración (Esto se denomina variable "final en blanco")

```
public class Proceso {  
  
    public void subProceso() {  
        final int valor;  
  
        int i = 0;  
  
        valor = 20;  
    }  
}
```

# Uso de final en argumentos en métodos

La palabra clave final también es posible utilizar en los argumentos de un método. Un argumento final no se podrá cambiar su valor dentro del método.

```
public class Proceso {  
  
    public void subProceso(final int valor) {  
  
        valor = 1920;  
  
    }  
}
```

# Uso de final en métodos

Los métodos marcados como finales no pueden ser sobre-escritos. Cuando se diseña una clase y se necesita asegurar el funcionamiento de un método, y dicho método es utilizado por otras estructuras de código que dependen de el, se recomendaría indicar como un método definitivo.

```
public class Persona {  
  
    public String nombre;  
    public int edad;  
  
    public Persona(String nombre, int edad) {  
        this.nombre = nombre;  
        this.edad = edad;  
    }  
  
    public final void hablar() {  
        System.out.println("Hola");  
    }  
}
```

```
public class Alumno extends Persona {  
  
    public Alumno(String nombre, int edad) {  
        super(nombre, edad);  
    }  
  
    @Override  
    public void hablar()  
    {  
  
    }  
}
```

# Uso de final en clases

Las clases marcadas como finales no pueden ser heredadas. Si se observa el código de las bibliotecas principales de Java, se encontrará muchas clases finales. Un ejemplo es la clase String y Wrapper(Integer, Float, Boolean, etc).

```
public final class Persona {
```

```
    public String nombre;
```

```
    public int edad;
```

```
    public Persona(String nombre, int edad) {
```

```
        this.nombre = nombre;
```

```
        this.edad = edad;
```

```
    }
```

```
}
```

```
public class Alumno extends Persona {
```

```
    public Alumno(String nombre, int edad) {
```

```
        super(nombre, edad);
```

```
    }
```

```
}
```