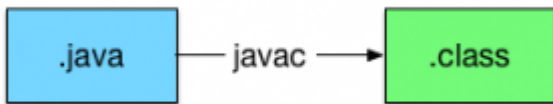


¿Cómo funciona el concepto de javac optimization? . Recordemos que javac es el compilador de Java y se encarga de compilar nuestro código “Java” en código para la maquina virtual.



**CURSO SPRING BOOT
GRATIS
APUNTATE!!**

Javac Optimization

En principio solemos pensar que se trata de una transformación del código directa , sin embargo no es exactamente así ya que el compilador es capaz de analizar el código y realizar optimizaciones en él. Vamos a ver un ejemplo sencillo de que cosas hace el compilador para mejorar nuestro código. El siguiente bloque de código esta lleno de constantes.

```
package com.arquitecturajava;  
  
public class MiClase {  
  
    public static final int asistentes=10;  
    public static final int gasto=5;  
    public static final int gastoTotal= asistentes*gasto;  
    public void imprimir () {
```

```

if (gastoTotal&gt;=50) {
System.out.println("hemos gastado mas " +"de 50 euros");
}
}

}

```

El código compila correctamente y nos generará un fichero .class . Si revisamos el código nos podemos dar cuenta de que realmente hay mucho código sobrante que con un análisis sencillo se eliminaría. Es momento de usar un decompilador de Java y ver cual es contenido real de nuestro fichero compilado. En este caso he usado uno online [JavaDecompiler](#) que nos muestra el código que realmente contiene nuestro fichero.

```

/*
 * Decompiled with CFR 0_114.
 */
package com.arquitecturajava;

import java.io.PrintStream;

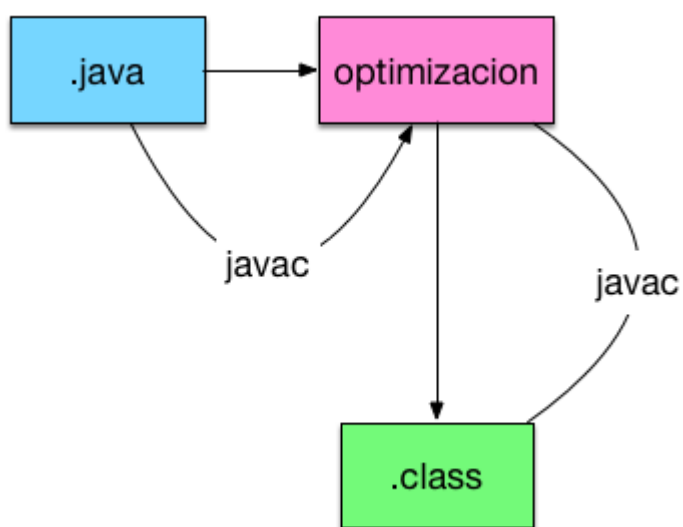
public class MiClase {
    public static final int asistentes = 10;
    public static final int gasto = 5;
    public static final int gastoTotal = 50;

    public void imprimir() {
        System.out.println("hemos gastado mas de 50 euros");
    }
}

```

```
}
```

El código para nuestra sorpresa será bastante diferente al que nosotros habíamos construido.



Es aquí donde el compilador de Java realiza sus optimizaciones y es capaz de entender que el gasto total es un valor fijo y que el if sobraba porque siempre el número será mayor de 50.

**CURSO SPRING BOOT
GRATIS
APUNTATE!!**

Otros artículos relacionados:

[Java StringBuilder](#)

Java WorkFlows

Java Flyweight