

Javascript cada día que pasa se usa más y existen mas librerías dentro de la plataforma. Java sigue siendo una de las tecnologías líderes del entorno servidor . Hoy por hoy podemos usar JavaScript dentro el mundo Java apoyandonos en **Nashorn** que es el motor de JavaScript embebido en Java 8

Un ejemplo de Java Nashorn JavaScript

Vamos a ver como utilizar el motor de JavaScript dentro del mundo de Java.

```
package com.arquitecturajava;

import javax.script.ScriptEngine;
import javax.script.ScriptEngineManager;
import javax.script.ScriptException;

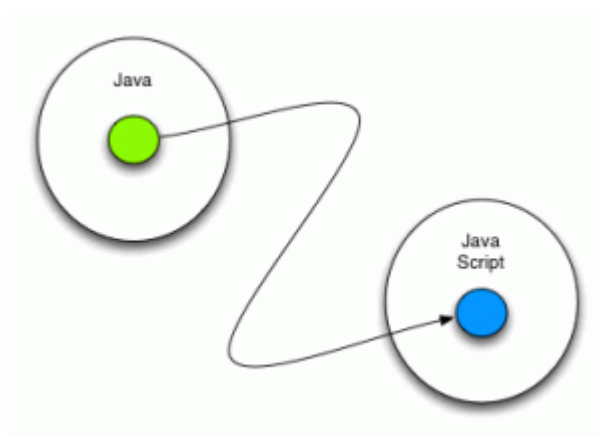
public class Principal {

    public static void main(String args[]) {

        ScriptEngineManager manager= new ScriptEngineManager();
        ScriptEngine motor= manager.getEngineByName("nashorn");
        try {
            Object resultado= motor.eval("print('hola')");
            Persona p= new Persona();
            p.setNombre("maria");
            motor.put("persona", p);
            motor.eval("print(persona.getNombre())");
        } catch (ScriptException e) {
            e.printStackTrace();
        }
    }
}
```

```
} catch (ScriptException e) {  
// TODO Auto-generated catch block  
e.printStackTrace();  
}  
  
}  
}  
  
&nbsp;
```

Como se puede ver es suficiente con usar la clase `ScriptEngineManager` que hace las funciones de Factoría y crea un nuevo Motor de Javascript (`ScriptEngine`) el cual a través de la función `eval` evalúa el código de JavaScript que queramos ejecutar. Además podemos pasar objetos de Java al mundo JavaScript de forma prácticamente transparente como se muestra en el ejemplo.



Esto nos permitirá una gran flexibilidad ya que podemos cambiar dinámicamente el código que recibe el `ScriptEngine` en tiempo de ejecución para cambiar el comportamiento.

Otros artículos relacionados : [jQuery Sizzle Engine](#) , [Java 8 Lambda](#)