

Actividad 3.2.2: Espacio de nombres

- Leer y contestar:

url1: <http://notasjs.blogspot.com.es/2012/06/espacios-de-nombres-en-javascript.html>

¿Para poder utilizar cualquier nombre de variable con seguridad en nuestros ficheros .JS que estrategia podemos adoptar?

Crear un objeto que haga de namespace o módulo definiendo dentro todos los atributos y métodos que va a utilizar el programa

¿Por qué razón funciona esa estrategia?

Porque de esta manera solo está expuesta una sola variable y no todas las del programa. Al estar definidas dentro de una clase para acceder a ellas siempre se hace desde la clase principal.

Da un ejemplo práctico en el que se explique el conflicto y su solución

Ejemplo: Coche tiene un atributo rueda, motor, volante... Podrá existir otra variable rueda, motor, volante... en el programa pero no coche.

De esta manera se protegen las variables por el grupo coche.

url2: <http://notasjs.blogspot.com.es/2012/04/el-patron-modulo-en-javascript.html>

Que es el patrón de Christian Heilmann?

Es una mejora del patrón módulo.

Consiste en definir todas las variables y funciones de la misma manera dentro del espacio de nombres, y en el return se colocan las variables y funciones que se quieren tener públicas.

Crear, siguiendo el patrón de Christian Heilmann, un módulo que se llame miCalculadora con Atributos privados: operador1 y operador2. Métodos privados: sumar() y restar(). Métodos públicos: operacionRandom(). El operador random hará una suma o una resta de manera aleatoria.

Index.html

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <title>miModuloCalculadora</title>
    <meta name="actividad 3.2.2" content="miCalculadora">
    <script type="text/javascript" src="miCalculadora.js"></script>
  </head>
  <body>
    <div id="resultado"></div>
  </body>
  <script>
    document.getElementById("resultado").innerHTML=
miCalculadora.operacionRandom(1,1);
  </script>
</html>
```

miCalculadora.js

```
var miCalculadora = (function () {
  //Atributos privados
  var operador1;
  var operador2;
  //Métodos privados
  function sumar(){
    return operador1 + operador2;
  }
  function restar(){
    return operador1 - operador2;
  }
  //Métodos públicos
  function operacionRandom(op1,op2){
    operador1 = op1;
    operador2 = op2;
    //aleatorio entre 0 - 1
    var num = Math.floor(Math.random() * (1 - 0 + 1)) + 0;
    if (num == 0){
      var resultado = sumar();
    }
    else{
      var resultado = restar();
    }
    return resultado;
  }
})
```

```
    }  
    //Return  
    return {  
        operacionRandom: operacionRandom  
    }  
}());
```