

Autor: José Fco. Soto Camacho

16 de Diciembre de 2018

Práctica Tema 2

Desarrollo Web en Entorno Cliente

Índice

1 – Ejercicio 1	3
1.1 Código html (ejercicio1.html)	3
1.2 Código Javascript (ejercicio1.js)	3
1.3 Resultado	4
2 – Ejercicio 2	5
2.1 Código html (ejercicio2.html)	5
2.2 Código Javascript (ejercicio2.js)	5
2.3 Resultado	8
3 – Ejercicio 3	8
3.1 Código html (ejercicio3.html)	8
3.2 Código Javascript (ejercicio2.js)	9
3.3 Resultado	12
4 – Valoración Personal	13
5 – Referencias	13
5.1 Grupo Studium	13

1 – Ejercicio 1

1.1 Código html (ejercicio1.html)

```
<!DOCTYPE html>

<html>

  <head>

    <meta charset="UTF-8">

    <title>Ejercicio 1</title>

    <script src="ejercicio1.js"></script>

  </head>

  <body>

  </body>

</html>
```

1.2 Código Javascript (ejercicio1.js)

```
var Jugadores = function(nom, ape, ocu, niv, fac)
{
  this.nombre = nom
  this.apellido = ape
  this.ocupacion = ocu
  this.nivel = niv
  this.faccion = fac
}
```

```
}

Jugadores.prototype.toString = function jugadorToString()
{
    var resultado = this.nombre + " " + this.apellido + " es " + this.ocupacion + ", su nivel es " +
    this.nivel

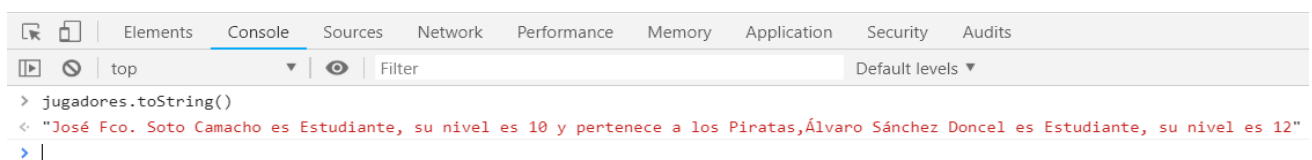
    if(this.faccion != undefined)
    {
        resultado += " y pertenece a los " + this.faccion;
    }

    return resultado;
}

jugadores = [new Jugadores("José Fco.", "Soto Camacho", "Estudiante", 10, "Piratas"), new
Jugadores("Álvaro", "Sánchez Doncel", "Estudiante", 12, "Vaqueros")]

delete jugadores[1].faccion
```

1.3 Resultado



2 – Ejercicio 2

2.1 Código html (ejercicio2.html)

```
<!DOCTYPE html>

<html>

  <head>

    <meta charset="UTF-8">

    <title>Ejercicio 2</title>

    <script src="ejercicio2.js"></script>

  </head>

  <body>

  </body>

</html>
```

2.2 Código Javascript (ejercicio2.js)

```
var Zombi = function(nombre, puntosVida, potencia)
{
  this.nombre = nombre;
  this.puntosVida = puntosVida;
  this.maxVida = puntosVida;
  this.potencia = potencia;
}
```

```
Zombi.prototype.atacar = function(objetivo)
{
    console.log(this.nombre + " ataca a " + objetivo + " y le causa " + this.potencia + " puntos de
daño");
}

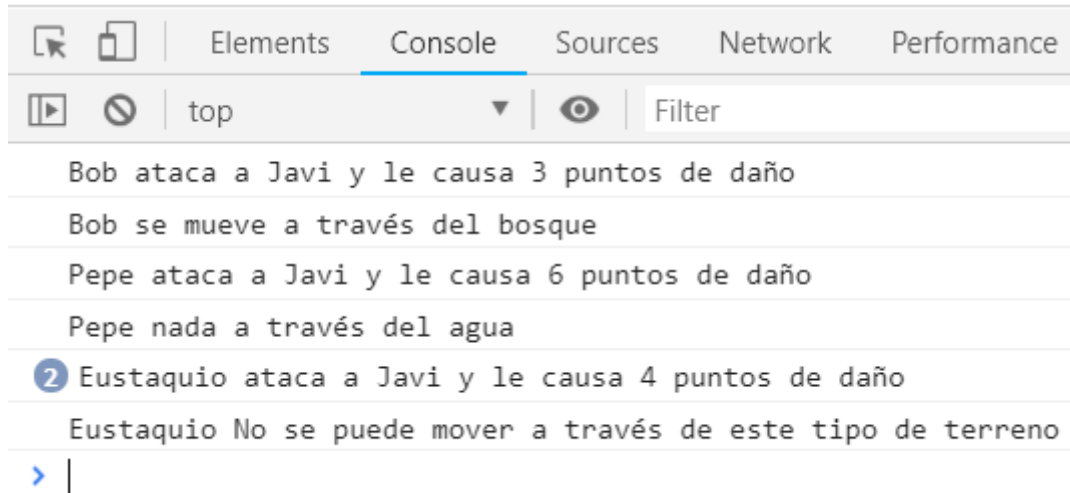
Zombi.prototype.mover = function(tipoTerreno)
{
    if (tipoTerreno === "agua")
    {
        console.log(this.nombre + " No se puede mover a través de este tipo de terreno");
    }
    else
    {
        console.log(this.nombre + " se mueve a través del " + tipoTerreno);
    }
}

var zombi = new Zombi('Bob', 2, 3);
zombi.atacar('Javi');
zombi.mover('bosque');

var monstruoAcuatico = new Zombi("Pepe", 4, 6)
monstruoAcuatico.atacar("Javi")
monstruoAcuatico.mover = function(tipoTerreno)
{
    if(tipoTerreno === "agua")
```

```
{  
    console.log(this.nombre + " nada a través del " + tipoTerreno);  
}  
  
else  
  
{  
    console.log(this.nombre + " No se puede mover a través de este tipo de terreno");  
}  
}  
  
monstruoAcuatico.mover("agua")  
  
  
var abominacion = new Zombi("Eustaquio", 8, 4)  
abominacion.atacar = function(objetivo)  
{  
    for(var i = 0; i < 2; i++)  
    {  
        console.log(this.nombre + " ataca a " + objetivo + " y le causa " + this.potencia + " puntos de  
daño");  
    }  
}  
  
abominacion.atacar("Javi")  
abominacion.mover("agua")
```


2.3 Resultado



3 – Ejercicio 3

3.1 Código html (ejercicio3.html)

```
<!DOCTYPE html>

<html>

  <head>

    <meta charset="UTF-8">

    <title>Ejercicio 3</title>

    <script src="ejercicio3.js"></script>

  </head>

  <body>

  </body>

</html>
```


3.2 Código Javascript (ejercicio2.js)

```
var Consumible = function(_nombre, _efecto)
{
    this.nombre = _nombre;
    this.efecto = _efecto;
}

Consumible.prototype.consumir = function(personaje)
{
    console.log(personaje.nombre + " consume " + this.nombre + " y " + this.efecto);
}

var manzana = new Consumible('Manzana', 'no tiene ningún efecto');

var ricardo =
{
    nombre: 'Ricardo',
    potencia: 1,
    puntosVida: 8,
    maxPuntosVida: 10,
    consumir: function(consumible)
    {
        consumible.consumir(this);
    }
}

ricardo.consumir(manzana);
```

```
var PlantaCurativa = function(_nombre, _efecto, _vidaRecuperada)
{
    Consumible.call(this, _nombre, _efecto)

    this.vidaRecuperada = _vidaRecuperada
}

PlantaCurativa.prototype = Object.create(Consumible.prototype)

PlantaCurativa.prototype.consumir = function(personaje)
{
    if(personaje.puntosVida == personaje.maxPuntosVida)
    {
        console.log(personaje.nombre + " consume " + this.nombre + " y no sucede nada")
    }
    else
    {
        if((this.vidaRecuperada + personaje.puntosVida) > personaje.maxPuntosVida)
        {
            console.log(personaje.nombre + " consume " + this.nombre + " y " + this.efecto + " " +
                (personaje.maxPuntosVida - personaje.puntosVida) + " puntos de vida.")
            personaje.puntosVida = personaje.maxPuntosVida
        }
        else
        {
            console.log(personaje.nombre + " consume " + this.nombre + " y " + this.efecto + " " +
```

```
        this.vidaRecuperada + " puntos de vida.")
        personaje.puntosVida += this.vidaRecuperada
    }
}
}

var plantaCurativaGrande = new PlantaCurativa("Planta Curativa Grande", "recupera", 10)
ricardo.consumir(plantaCurativaGrande)

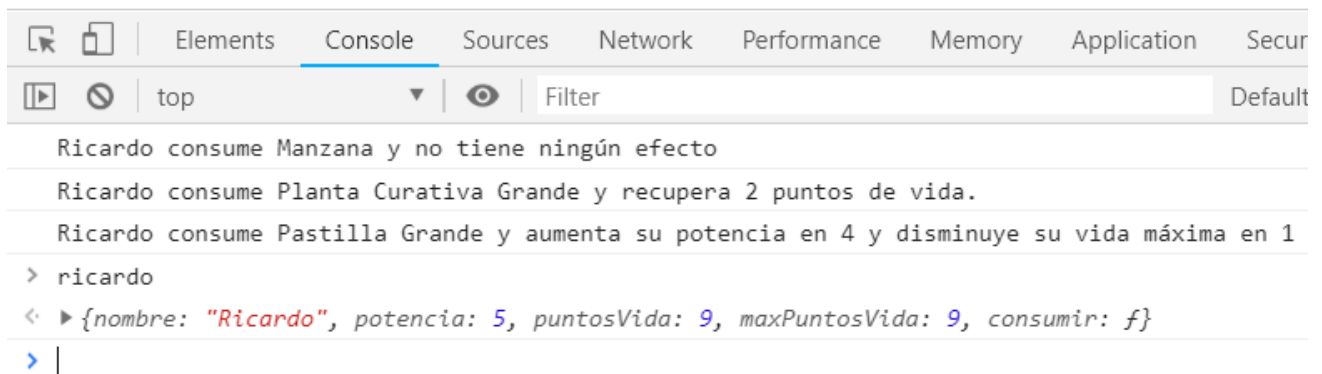
var Pastilla = function(_nombre, _efecto, _efecto2, _potenciaPlus, _vidaMenos)
{
    Consumible.call(this, _nombre, _efecto)
    this.efecto2 = _efecto2
    this.potenciaPlus = _potenciaPlus
    this.vidaMenos = _vidaMenos
}

Pastilla.prototype = Object.create(Consumible.prototype)

Pastilla.prototype.consumir = function(personaje)
{
    personaje.potencia += this.potenciaPlus
    if(personaje.puntosVida >= (personaje.maxPuntosVida - this.vidaMenos))
    {
        personaje.maxPuntosVida -= this.vidaMenos
        personaje.puntosVida = personaje.maxPuntosVida
    }
}
```

```
}  
  
else  
  
{  
  
    personaje.maxPuntosVida -= this.vidaMenos  
  
}  
  
console.log(personaje.nombre + " consume " + this.nombre + " y " + this.efecto + " " +  
this.potenciaPlus + " y " + this.efecto2 + " " + this.vidaMenos)  
  
}  
  
var pastillaGrande = new Pastilla ("Pastilla Grande", "aumenta su potencia en", "disminuye su vida  
máxima en", 4, 1)  
  
ricardo.consumir(pastillaGrande)
```

3.3 Resultado



```
Elements Console Sources Network Performance Memory Application Secur  
top Filter Default  
Ricardo consume Manzana y no tiene ningún efecto  
Ricardo consume Planta Curativa Grande y recupera 2 puntos de vida.  
Ricardo consume Pastilla Grande y aumenta su potencia en 4 y disminuye su vida máxima en 1  
> ricardo  
< ▶ {nombre: "Ricardo", potencia: 5, puntosVida: 9, maxPuntosVida: 9, consumir: f}  
> |
```

4 – Valoración Personal

En esta práctica he aprendido a usar los objetos con Javascript, a crear constructores y heredar atributos y funciones de otros, a usar el prototipo de ellos, etc.

Cada tema nuevo que se ve en clase de Javascript me parece más interesante que el anterior, sin lugar a dudas Javascript es realmente divertido de usar para el programador dadas las cosas que se pueden realizar con él, además con muchas facilidades.

5 – Referencias

5.1 Grupo Studium

Para la realización de esta práctica han sido necesarios los conocimientos adquiridos a través del temario “Tema 2: Objetos en JavaScript” del módulo Desarrollo Web en Entorno Cliente, del ciclo formativo de grado superior de Desarrollo de Aplicaciones Web, realizado por el centro privado Grupo Studium, colgado en su plataforma online.