

Programación con Java Script I

Sesión sincrónica 4

# Objetos y arreglos



```
using System;  
using System.Data;  
  
namespace MyProject  
{  
    /// <summary>  
    /// Customer Master  
    /// </summary>  
    /// <remarks></remarks>  
    public class Customer  
    {  
        private string pCustomerCode;  
        private string pCustomerName;  
        private string pAddress;  
        private string pMobileNo;  
        private string pEmail;  
  
        public string CustomerCode  
        {  
            get { return pCustomerCode; }  
            set { pCustomerCode = value; }  
        }  
  
        public string CustomerName  
        {  
            get { return pCustomerName; }  
            set { pCustomerName = value; }  
        }  
    }  
}
```

# Bienvenida y actividad de bienestar

**Duración: 10 minutos.**

**Nombre de la práctica:** Fomentando la atención plena

**Descripción de la práctica:** Llevarás a cabo breves ejercicios de meditación para fomentar la atención plena en tus actividades diarias.

**Palabras clave:** Atención plena, fortalezas de carácter, autorregulación

**Instrucciones para el participante:** La meditación es una herramienta que ayuda a mejorar el desempeño de cualquier persona, ya que fomenta el desarrollo de la atención plena en una sola actividad. Para fomentar la atención plena y lograr cada vez más estar en una zona de concentración mientras realizas tus actividades cotidianas, puedes llevar a cabo los siguientes ejercicios de meditación:

Encuentra en algún momento del día cinco minutos para ti, siéntate en un lugar cómodo, donde no tengas distracciones.

1. Haz tres respiraciones profundas por nariz y exhala por nariz.
2. Comienza a hacer un repaso de tu día, de lo que más te acuerdes, Ej. Te levantaste, ¿qué hiciste? ¿Desayunaste? ¿Te bañaste? ¿Diste los buenos días?, etcétera. Si desayunaste, ¿qué fue lo que desayunaste, ¿te gustó? ¿Tomaste tu alimento despacio o apurado?, si estabas apurado. ¿qué era lo que te tenía en esa situación?
3. Sigue meditando en lo que te acuerdes: ¿te molestase con alguien? ¿Por qué? ¿Qué fue lo que pasó? ¿Crees que era posible haber reaccionado de alguna manera más pacífica?

Con este ejercicio te darás cuenta de que reaccionamos o hacemos cosas de manera automática, algunas veces si estamos más conscientes y presentes, podemos tener otra actitud sin que alguna situación nos afecte demasiado.

Fuente: <http://talentdevelop.com/articles/Page8.html>

\*El instructor deberá preguntar a los aprendedores sobre el tema estudiado y/o actividades que



# Actividad guiada

## Parte 1

**Duración: 75 minutos.**

### Ejercicio 1

En esta sesión vas a crear el juego “piedra, papel o tijera”, utilizando funciones y objetos.

Para comenzar el ejercicio:

1. Crea un archivo js en blanco, se sugiere que le pongas main.js.
2. Crea un archivo html con la estructura básica.
3. Agrega un link desde el html hacia el js.

El archivo html se deberá ver de la siguiente forma:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <script src="main.js"></script>
</body>
</html>
```

## Actividad guiada

### Parte 1

Ahora, en el archivo js, realiza lo siguiente:

1. Crea el objeto juego, el cual tendrá en su interior tres arreglos:
  - a. **opciones**: {0: "✂️", 1: "🖐️", 2: "👊"}
  - b. **nombreJugadores**: {**jugador1**: "jugador1", **jugador2**: "Computadora"}
  - c. **scores**: {**puntosJ1**: 0, **puntosJ2**: 0}
2. Crea una función flecha para solicitar el nombre del usuario, a través de una ventana de mensaje del navegador, y asígnala a la variable **pedirNuevoNombre**.
3. Crea una función expresada que permita asignar el nombre proporcionado por el usuario al `nombre`, nómbrala **cambiarNombre**. Esta función recibirá como parámetro el nuevo nombre proporcionado por el usuario y tendrá que realizar lo siguiente:
  - a. Validar, a través de expresiones regulares, que el nombre contenga letras y números. En caso de que solo contenga números o caracteres especiales, deberá asignar el nombre por defecto declarado en el objeto.
  - b. En cada caso imprime un mensaje de bienvenida:
    - i. En caso de que el nombre cumpla con las condiciones, imprime:  
"Bienvenid@ " + nombre del jugador + "🖐️♂️!"
    - ii. En caso de que el nombre no sea aceptado, imprime:  
"Mejor te llamaré... " + nombre por defecto declarado en el objeto.
4. Crea una función que genere un número aleatorio:
  - a. El valor de este aleatorio deberá estar entre 0 y 2, puedes realizarlo utilizando el método **random** de la función **Math**, redondéalo al entero menor utilizando el método **floor** de la función **Math**.
  - b. Asígnalo a la variable **aleatorio**.
  - c. Regresa el valor de esa variable convertido en cadena.
5. Crea una función expresada llamada **imprimirManoAleatoria**, la cual deberá tomar el número de la variable **aleatorio** y, con base en las opciones del objeto juego, regresará el ícono piedra, papel o tijera, respectivamente, obtenido desde el objeto.

## Receso

Duración: 10 minutos.

```
def __init__(self, **kwargs):
    self.name = kwargs.get("name")
    self.damage = kwargs.get("damage")
    self.armor = kwargs.get("armor")
    self.hit_points = kwargs.get("hp")
    self.current_hit_points = kwargs.get("hp")
    self.level = kwargs.get("level")

def attack(self, enemy: 'Unit') -> int:
    """
    Attack enemy unit. Return number of damage
    """
    damage_top_limit = self.damage + random.randint(0, 10)
    damage_bot_limit = self.damage - random.randint(0, 10)
    calculated_damage = random.randint(damage_bot_limit, damage_top_limit)
    if calculated_damage < 0:
        return 0
    enemy.current_hit_points -= calculated_damage
```

## Actividad guiada

### Parte 2

**Duración: 75 minutos.**

6. Crea una función expresada llamada imprimirResultado. La funcionalidad de esta será **imprimir el resultado** del juego, comparando si gana el jugador 1 o el jugador 2, con base en los siguientes criterios:
  - a. Si el jugador 2 tiene más juegos ganados que el jugador 1, imprimir en la consola la cadena: "Fin del juego gana " + nombre del jugador 2 + "Game Over 🤖👊".
  - b. Si el jugador 1 tiene más juegos ganados que el jugador 2, imprimir en la consola la cadena: "Fin del juego gana " + nombre del jugador 1 + "👊🤖".
7. Crea una función para saber quién gana, tomando como parámetro un 0 para empate, 1 para jugador1 y 2 para jugador2. Asigna la función gana a la variable resultado y programa los siguientes criterios:
  - a. Si gana es igual a empate (valor igual a 0), escribir la cadena:
 

```
"👊" + nombre jugador 1 + ":[\" + puntos del jugador 1 + \"]"
+ nombre jugador 2 + ":[\" + puntos del jugador 2 + \"]👊");
```
  - b. Si gana el jugador 1 (el valor es igual a 1), escribir la cadena:
 

```
"👊" + nombre jugador 1 + ":[\" + puntos del jugador 1 + \"]"
+ nombre del jugador 2 + ":[\" + puntos del jugador 2 + \"]");
```
  - c. En caso contrario, imprimir la siguiente cadena:
 

```
Nombre jugador 1 + ":[\" + puntos jugador 1 + \"]"
+ nombre jugador 2 + ":[\" + puntos jugador 2 + \"]👊");
```

## Actividad guiada

### Parte 2

8. Crea una función expresada para evaluar quién gana, la cual recibirá como parámetros la **elección** del **jugador1** y del **jugador2**. Nombra a esta función evalua. Agrega lo siguiente en el cuerpo de la función:
- a. Crea la variable **gana** y asígnale el valor 0.
  - b. Si la elección del jugador 1 es igual a la del jugador 2, imprime:
    - i. Elección jugador1 + " VS " + Elección jugador2 + " empatan".
  - c. Si el jugador 1 eligió tijeras y el jugador 2 eligió papel, si el jugador 1 eligió papel y el usuario 2 eligió piedra o si el jugador 1 eligió piedra y el usuario 2 tijeras:
    - i. Imprime elección jugador 1 + “gana a “ + elección jugador 2.
    - ii. Suma 1 a los puntos del jugador 1.
    - iii. Asigna 1 a la variable **gana**.
  - d. En caso contrario:
    - i. Imprime elección jugador 2 + “**gana** a “ + elección jugador 1.
    - ii. Suma 1 a los puntos del jugador 2.
    - iii. Asigna 2 a la variable **gana**.
  - e. Con el resultado de la condicional, llama a la función resultado y pásale como argumento la variable gana para imprimir en pantalla quién gana la partida.



## Actividad guiada

### Parte 2

9. Crea en una expresión la función principal para el programa y nómbrala main:
  - a. Crea un bucle while para que se ejecute mientras el valor sea diferente a 3.
  - b. Dentro del bucle, declara la variable **jugador1Hand** y asígnale el valor de la función **imprimirManoAleatoria()**;
  - c. Ahora, declara la variable **jugador2Hand** y asígnale el valor de la función **imprimirManoAleatoria()**.
  - d. Imprime en consola el nombre del **jugador1** y su **elección aleatoria**.
  - e. Imprime en consola el nombre del **jugador2** y su **elección aleatoria**.
  - f. Llama a la función **evalua** y pásale como argumentos las **elecciones aleatorias** de los **jugadores**.
  - g. Como el juego termina cuando alguno de los jugadores gana 3 partidas, evalúa si alguno llegó a 3, en ese caso, llama a la función **imprimirResultado** y pásale como parámetro los **puntos** del jugador 1 y del jugador 2.
  - h. Fuerza a salir de la función utilizando el comando break.
10. A jugar:
  - a. Instancia la función **cambiarNombre**, pasándole como parámetro la función **pedirNuevoNombre()**.
  - b. Imprime en la consola la cadena " 🚩 Que comience el juego 🚩".
  - c. Llama a la función principal **main()**.
11. Ejecuta el archivo html.



# Actividad guiada

## Parte 2

Tu programa debería lucir como la siguiente imagen:

```

❏ Que comience el juego ❏
carlos 🖐️
Computadora ✖️
🖐️ gana a ✖️
🔴 carlos : [1] Computadora : [0]
carlos 🖐️
Computadora ✖️
✖️ gana a 🖐️
carlos : [1] Computadora : [1] 🔴
carlos 🖐️
Computadora 🖐️
🖐️ VS 🖐️ empatan
🔴 carlos : [1] Computadora : [1] 🔴
carlos 🖐️
Computadora ✖️
✖️ gana a 🖐️
carlos : [1] Computadora : [2] 🔴
carlos 🖐️
Computadora 🖐️
🖐️ VS 🖐️ empatan
🔴 carlos : [1] Computadora : [2] 🔴
carlos ✖️
Computadora ✖️
✖️ VS ✖️ empatan
🔴 carlos : [1] Computadora : [2] 🔴
carlos 🖐️
Computadora 🖐️
🖐️ gana a 🖐️
carlos : [1] Computadora : [3] 🔴
Fin del juego gana Game Over Computadora 🤖🖐️

```

## Actividad guiada Parte 2

### Consideraciones

Instructor: Refuerza los conocimientos de aquellos temas que consideres necesarios para que el aprendiz pueda realizar este ejercicio.

La obra presentada es propiedad de ENSEÑANZA E INVESTIGACIÓN SUPERIOR A.C. (UNIVERSIDAD TECMILENIO), protegida por la Ley Federal de Derecho de Autor; la alteración o deformación de una obra, así como su reproducción, exhibición o ejecución pública sin el consentimiento de su autor y titular de los derechos correspondientes es constitutivo de un delito tipificado en la Ley Federal de Derechos de Autor, así como en las Leyes Internacionales de Derecho de Autor.

El uso de imágenes, fragmentos de videos, fragmentos de eventos culturales, programas y demás material que sea objeto de protección de los derechos de autor, es exclusivamente para fines educativos e informativos, y cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por UNIVERSIDAD TECMILENIO.

Queda prohibido copiar, reproducir, distribuir, publicar, transmitir, difundir, o en cualquier modo explotar cualquier parte de esta obra sin la autorización previa por escrito de UNIVERSIDAD TECMILENIO. Sin embargo, usted podrá bajar material a su computadora personal para uso exclusivamente personal o educacional y no comercial limitado a una copia por página. No se podrá remover o alterar de la copia ninguna leyenda de Derechos de Autor o la que manifieste la autoría del material.