



## PROYECTO ENTREGABLE

Llegó el momento de poner a prueba los conocimientos adquiridos, pero no es más que eso. Es una oportunidad para validar su aprendizaje hasta el momento, se trata de una evaluación entregable, tendrán la posibilidad de resolver el trabajo por cuenta propia y entregarlo antes de la fecha límite.

### Recordamos:

La fecha límite de entrega es el viernes 8 de octubre a las 23:59 hs.

## CONSIGNAS

Primero que nada, deberán descargar la **plantilla de archivos .zip** que se encuentran en la misma carpeta que este documento. Luego, es fundamental que **lean atentamente todas las consignas antes de comenzar** a desarrollar sus soluciones. Una idea general del resultado buscado es clave para cumplir los requerimientos. Les dejamos esta pequeña ayuda del funcionamiento buscando en el gif que se llama "demo".



## CONSIDERACIONES

Solo deberán crear código Javascript en el archivo **code.js**, puntualmente en los espacios marcados dentro de las funciones establecidas.

```
JS code.js
<> index.html
JS listeners_NO_TOCAR.js
# styles.css
```

**Bajo ninguna circunstancia modifiquen el resto de los archivos.** Obviamente pueden revisar tanto el **HMTL** como el **CSS**, pero no se debe realizar cambio alguno ya que no impactarán en la corrección del proyecto. Solo se evaluará el código JavaScript por parte de los profesores a la hora de corregir.

### IMPORTANTE 📌

**NO AGREGAR COMENTARIOS EN LÍNEA EN EL code.js, AL MOMENTO DE ENTREGAR CONTROLAR QUE NO HAYA COMENTARIOS AGREGADOS.**

Los comentarios en línea son aquellos utilizando la doble barra al principio del texto, ejemplo:

```
//Esto es un comentario en linea
```



## PUNTO POR PUNTO

### PUNTO 1

Desarrollar la función para consultar al usuario los siguientes datos:

- nombre
- año de nacimiento
- ciudad
- si le interesa Javascript

Los datos obtenidos deberán ser almacenados dentro del objeto **datosPersona**.

#### → Tips

- ◆ La propiedad **edad** debe ser calculada, algo tan simple como restar el **año actual menos el año de nacimiento**.
- ◆ No se preocupen si aún no se cumplió el mes exacto. Tampoco es necesario usar el objeto Date.

### PUNTO 2

Desarrollar la función para renderizar los datos del usuario, a partir de la información recolectada en la función anterior y almacenada en el objeto **datosPersona**. Para ello, utilizar los elementos HTML que se encuentran en el archivo index.html, modificando los mismos con Javascript.

#### → Tips

- ◆ **No está permitido editar el documento index.html**, cualquier cambio debe realizarse utilizando Javascript.
- ◆ Ubica los datos del objeto en el **<span>** que corresponda.



## PUNTO 3

Desarrollar la función que recorra el listado y renderizar una especie de tarjeta con la información de cada materia. **Prestar atención, cada una de las clases CSS son necesarias para conservar el diseño:**

1. Cada tarjeta contenedora deberá tener la clase **'caja'**
2. Dentro de cada contenedor se deberá incluir:
  - a. una imagen con su correspondiente **src** y un **alt** que indique los **lenguajes**.
  - b. un párrafo que tenga las clase **'lenguajes'** y muestre los mismos.
  - c. un párrafo que tenga las clase **'bimestre'** y muestre el bimestre.
3. Al volver a clicar el botón "Obtener materias", no deberían volver a renderizarse las materias.

### → Tips

- ◆ Cada 'caja' se debe inyectar dentro del contenedor que ya se encuentra en el archivo index.html con el id **'fila'**.

## PUNTO 4

Desarrollar la función que permita al botón de alternar tema funcionar correctamente cada vez que se haga click.

1. Utilizar el elemento con el id **'sitio'** como nodo de referencia.
2. Implementar en el elemento la clase **'dark'** ya creada en el CSS.

### → Tips

- ◆ El elemento **'sitio'** es simplemente un contenedor que envuelve todo el resto de los elementos visibles.



## PUNTO 5

Finalmente, agregar un listener en nuestra página que nos permita captar el evento de teclado cuando presionamos la tecla **F**.

1. Captar cuando se presiona una tecla.
2. Si la tecla presionada es la **F**, al texto del párrafo **'sobre-mi'** se le deberá eliminar la clase que lo mantiene oculto,

### → Tips

- ◆ En este caso la tecla no debe alternar la visibilidad, solamente desocultar por única vez.