

## PRÁCTICA FINAL ECMAScript

---

### NORMAS GENERALES

A.- Esta práctica final de la asignatura de ECMAScript consiste en implementar código de JS sobre los documentos HTML de su proyecto de práctica de la asignatura de “Desarrollo de páginas web estáticas (HTML y CSS)”. Para ello, deberá seguir las indicaciones de los enunciados de los diferentes ejercicios.

B.- Además de este documento, se le proporcionará un archivo “json.js”, que contiene un JSON del resultado de ejecutar el método http GET sobre una api pública.

C.- Durante la realización de la práctica queda totalmente prohibido copiar código de algún compañero o de alguna fuente externa. **El incumplimiento de este apartado supondrá la calificación más baja (cero) en la presente práctica.**

D.- El código de JS estará en un archivo independiente, añadido de forma externa a la página web. El incumplimiento de este apartado supondrá la calificación más baja (cero) en la presente práctica.

E.- **Se recuerda que son de aplicación el Currículo del Curso y la Guía Docente de la asignatura.**

F.- Se valorará **la guía de estilo** y el uso de comentarios en el código.

G.- **Entre paréntesis se indica la valoración de cada ejercicio. Se recuerda que la nota final de la práctica es el 50% de la nota final de la asignatura de ECMAScript.**

H.- Instrucciones de Entrega:

- **Antes de realizar el examen teórico de la asignatura de ECMAScript, deberá haber entregado esta práctica final. El incumplimiento de este apartado supondrá la calificación más baja (cero) en la presente práctica.**

- Cree un archivo zip con su nombre que contenga todo el contenido de su proyecto.

- Entregue directamente dicho archivo al profesor o envíeselo vía mensaje directo por privado en Sandstorm.

\*\*\*\*\*

### FINAL DE LAS NORMAS GENERALES

\*\*\*\*\*

## PRÁCTICA FINAL

### ECMAScript

---

Cuando tenga que mostrar algún resultado por consola, deberá aparecer el siguiente texto:

"Ejercicio NumEjercicio ==> " y a continuación el resultado, Ejemplo: **Ejercicio 1 ==>**  
...

1. **(0.5 puntos)** Uso de la guía de estilo y comentarios.
  2. **(0.25 puntos)** Almacene en el almacenamiento local del navegador una propiedad que se llame "practica" con el valor "Práctica Final ECMAScript 2025". Asegúrese de que esta se elimine cuando el navegador se cierre.
  3. **(0.5 puntos)** Implemente una expresión de función anónima que devuelva la hora del sistema en formato **HH:MM:SS**
  4. **(0.5 puntos)** En el encabezado de la página principal (en el lugar que considere), debe aparecer la hora del sistema, pero debe simular un reloj digital. **(Apóyese en la expresión de función anónima del ejercicio 3, para ello inserte en el documento html el elemento que estime oportuno).**
  5. **(0.25 puntos)** Modifique el texto del "enunciado principal / título" que aparezca en el encabezado de la página principal de su proyecto, de tal manera que cuando el puntero del ratón se sitúe encima, dicho texto debe aparecer con un color amarillo "#ffff00".  
  
Y cuando el puntero salga del elemento, el color del texto debe cambiar al original que tenía.
- (Si en el encabezado de su página principal no tiene ningún título, insértelo para realizar este ejercicio)**
6. **(0.5 puntos)** Modifique una imagen que aparezca en cualquier documento html de su proyecto, de tal manera que cada vez que haga click sobre ella, la sustituya por otra imagen de su elección, es decir, la primera vez, cuando haga click, debe cambiar a "imagen2", la siguiente vez que haga click, a la anterior "imagen1" y así sucesivamente... **(Este cambio sólo debe producirse cuando haga click sobre dicha imagen).**
  7. **(0.25 puntos)** Muestre por consola **el nombre** de todos los usuarios del JSON del archivo json.js

## PRÁCTICA FINAL

### ECMAScript

---

8. (0.75 puntos) Cree una clase "Usuario" que contenga, lo siguiente:

- **Propiedades privadas:** "idUser", "nombre", "nombreUser", "email", "empresa", "direccion" y "url".
- la propiedad **empresa**, sólo devolverá el nombre de dicha empresa.
- **direccion será un objeto literal** y deberá contener las siguientes propiedades: "calle", "ciudad", "codigoPostal".
- **Método estático getId(url)**, que pasándole como parámetro el identificador único (URI) del usuario, devuelva su id.

9. (0.25 puntos) Instancie un objeto de la clase Usuario con el identificador "userPrueba". Este objeto debe tener los siguientes valores:

```
nombre = "Prueba Practica Final"
nombreUser = "PruebaPF7"
email = "pruebapf7@hotmail.com"
empresa = "Leroy Merlin"
direccion = {calle: "Gravina 7", ciudad: "Roma", codigoPostal: "41449"}
url = https://prueba.dev/api/users/102/
```

Muestre por consola las propiedades del objeto userPrueba (**incluido el idUser**).

10. (0.5 puntos) Implemente una función que pasándole como parámetro un objeto del json mapee y cree un objeto del tipo Usuario.

11. (0.5 puntos) Implemente una función que recorra el JSON y devuelva un array de objetos del tipo Usuario. (**Apóyese en la función del ejercicio anterior**).

12. (0.25 puntos) Cree una variable global que contenga el resultado de la función del punto anterior. **Muestre por consola SÓLO el nombre del usuario.**

13. (0.5 puntos) Cree las variables que considere necesarias, de tal manera que cada variable contenga los usuarios de la misma ciudad.

Hágalo a partir del array de objetos de usuarios del punto anterior. **Muestre el resultado por consola.**

14. (0.5 puntos) Ordene de forma creciente los arrays anteriores por el valor de la propiedad "nombre". **Muestre el resultado por consola.**

## PRÁCTICA FINAL

### ECMASCRIPT

---

**15. (1 puntos)** Implemente la función **mostrarUsuarios()**.

Inserte en la barra de navegación de la página principal de su proyecto, una opción que se llame **"Usuarios"**, de tal manera, que cuando el usuario haga click sobre ella, se muestre un modal con todos los usuarios ordenados por el valor de la propiedad "nombre".

Diseñe el modal para que aparezcan los siguientes datos de cada usuario: **Nombre, Usuario, Email y Empresa**.

**16. (0.5 puntos)** Implemente la función **filtrarCiudad()**.

En el modal del ejercicio anterior, inserte un elemento Select que muestre como opciones los valores de las ciudades del json.

Cuando el usuario seleccione una ciudad de la lista desplegable, se deberá actualizar la vista mostrando solamente aquellos que sean de la ciudad seleccionada.

**17. (0.5 puntos)** Modifique la función del ejercicio 15, para que cambie el color del texto de los usuarios mostrados en función de la ciudad del usuario:

Gwenborough, color = azul

Wisokyburgh, color = verde

**18. (0.75 puntos)** Implemente una función **calcularDatos()**, que obtenga los siguientes datos de todos los usuarios:

- Menor edad
- Nombre del usuario de menor edad
- Mayor edad
- Nombre del usuario de mayor edad

Esta función se debe invocar cuando se muestren los datos de los usuarios **mostrarUsuarios()**.

**Por lo tanto, deberá insertar en el modal, a continuación del listado de usuarios, los elementos que estime oportuno para mostrar estos datos obtenidos a través de la función **calcularDatos()**.**

Cuando se seleccione una ciudad a través del elemento Select, también se deberán actualizar estos datos, calculados según la ciudad seleccionada.

**19. (0.5 puntos)** Implemente un evento, de tal manera que cuando se pulse la tecla **"p"** (minúscula) ó **"P"** (mayúscula) aparezca una ventana con el texto de la variable global practica (Ejercicio 1), esta ventana se debe cerrar automáticamente pasado 3 segundos.

**PRÁCTICA FINAL**  
**ECMASCRIPT**

---

**20. (0.75 puntos)** Implemente la función **filtrarDireccion()**, de tal manera que cuando el usuario seleccione un usuario en la lista desplegable, aparezca su dirección con el siguiente formato: **C/ nombre de la calle, Ciudad (código postal)**.

Para ello, deberá insertar, al final del modal, un elemento Select que muestre como opciones los nombres de todos los usuarios, y otro elemento html, de su elección, que muestre su dirección con el formato exigido.