

UD 3 - JavaScript: tipos y objeto document

2º DAW, Desarrollo web en entorno cliente

Contexto y prerequisites

En este guión de prácticas vas a trabajar sobre los conceptos vistos en la UD 3. Puedes tomar como referencia las secciones [Tipos de dato](#) y [Modelo de objetos del documento \(DOM\)](#) de [JavaScript.info](#).

En clase usamos como entorno de desarrollo Visual Studio Code y las extensiones [ESLint](#), [Prettier](#) y [JavaScript \(ES6\) code snippets](#). Te recomiendo usar el mismo IDE para homogeneizar la experiencia como desarrollador y tener menos dificultades a la hora de buscar ayuda si la necesitas.

Adjunto a este guión, tienes un fichero comprimido con el código HTML de cada ejercicio, de forma que sólo tengas que centrarte en implementar el código de JavaScript.

Ejercicios

En este guión de prácticas será recomendable que incluyas el código JavaScript desde ficheros externos. Esto es así porque seguramente habrá funciones que puedas reutilizar en varios ejercicios. De este modo podrías incluir un par de scripts, un *common.js* con funciones que puedas reutilizar y un *exercise_1.js* para el primer ejercicio por ejemplo.

Algo así es lo que tendrías en un hipotético *exercise_1.html*.

```
<!DOCTYPE html>
<html>
  <head>
    <script src="common.js"></script>
    <script src="exercise_1.js"></script>
  </head>
  <body>
    <p>Hi! I just wanna say hello.</p>
  </body>
</html>
```

Adjunto a la tarea tendrás el esqueleto necesario para afrontar el primer ejercicio, así como un par de funciones dentro de *common.js* que pueden serte de utilidad.

Ejercicio 1: convertir decimal a binario, octal y hexadecimal (1 pt)

En este ejercicio se pretende que dado un valor introducido por el usuario, validando que sea un entero mayor o igual a cero, se pueda convertir a binario, octal o hexadecimal. Para

ello, podrás usar un campo de tipo *input* y tres botones, cada uno de ellos implementando la función para convertir a cada una de las bases. El resultado será mostrado por pantalla en un elemento dentro del código HTML.

Aquí tienes una captura de pantalla de cómo funciona la página al introducir un valor y pulsar en convertir a binario:

Conversor de base numérica

Introduce un número entero en mayor o igual que cero en base decimal

El resultado de convertir '1247' a binario es '10011011111'

En caso de introducir un valor no procesable, en el mismo lugar, escribirías el mensaje de error correspondiente en color rojo:

Conversor de base numérica

Introduce un número entero en mayor o igual que cero en base decimal

Error: no puedo procesar el valor '1247fdgfdg'

Ejercicio 2: transformador de cadenas (2,5 pt)

En este caso, vamos a transformar el texto de una caja a otra. Para ello, crearás un grupo de radio button con las siguientes opciones:

- **Mayúsculas:** Este texto es un ejemplo >> ESTE TEXTO ES UN EJEMPLO
- **Minúsculas:** Este texto es un ejemplo >> este texto es un ejemplo
- **Capitalizar:** Este texto es un ejemplo >> Este Texto Es Un Ejemplo
- **CamelCase:** Este texto es un ejemplo >> EsteTextoEsUnEjemplo
- **Reshulón:** Este texto es un ejemplo >> EsTE tEXtO eS un EjEmPLO



- **Mimimi:** Este texto es un ejemplo >> Isti tixti is in ijimpli

El texto se transformará cada vez que se modifique el cuadro de arriba y se convertirá según la función que esté habilitada en el *radio button*. Aquí tienes una idea:

Transformador de cadenas

Se mata a un hombre, pero no se puede matar una idea

☐ Mayúsculas ☐ Minúsculas ☐ Capitalizar ☐ Camel case ☐ Mi mi mi ☒ Reshulón

SE maTA A Un h0mBrE, PeR0 n0 SE PuEde MATaR uNA IdEA

Ejercicio 3: cálculos con fecha (1,5 pt)

Este ejercicio es algo más fácil. Utilizando un desplegable con las opciones “ahora”, “hace 5 minutos” y “próxima semana”. Cada vez que se cambie de opción en el desplegable, quiero que saques por pantalla la siguiente información:

- **Ahora:** Hoy es domingo, 11 de diciembre de 2022 y son las 22:47:15
- **Hace 5 minutos:** 2022/12/11 22:42:15.157
- **Próxima semana:** 2022-50

Con esta captura puedes hacerte una pequeña idea de lo que se requiere:

Utilidades de fecha

Selecciona una fecha para consultar:

Hoy es lunes, 12 de diciembre de 2022 y son las 1:12:30

Ejercicio 4: validar email (1,5 pt)

Crea un campo de texto para introducir una dirección de correo electrónico. Al ir escribiendo en dicho campo una dirección de correo electrónico, mostrar un texto al lado del campo que indique si es correcto o no. Colorea el texto informativo en verde si es correcto y rojo en caso contrario. Debes usar una expresión regular para ello.

Estas dos capturas pueden servirte de orientación:

Validador de correo electrónico

NO VÁLIDO

Validador de correo electrónico

VÁLIDO

Ejercicio 5: lanzar dados (3,5 pt)

Este último ejercicio es un poco más complejo. En él pretendo que puedas lanzar uno o dos dados de seis caras. Puedes implementar el lanzamiento como quieras, siéntete libre (un par de botones, un *radio button* más un botón, un desplegable más un botón etc.). En el código de ejemplo que te propongo, he creado dos párrafos que contienen uno y dos emojis de dados. Dependiendo de en cuál haga click, tiro uno o dos, respectivamente:

Lanzador de dados




☐ Mostrar log en pantalla



Habrás observado que también hay un *checkbox* que debes implementar y que debe estar activo por defecto. Servirá para mostrar por pantalla un histórico de los lanzamientos realizados, de esta forma:



El histórico se está mostrando en un *textarea* no editable. Independientemente de que se muestre por pantalla o no, quiero que siempre saques por consola el resultado de cada tirada de dados.

Finalización y entrega

Ahora que ya has terminado los ejercicios, no te olvides de subir una copia descargada en .zip de tu repositorio, o la carpeta que tengas con los ejercicios de esta unidad, a la tarea que será creada en Google Classroom antes de la fecha de cierre.

Recuerda redactar tu código de la forma más limpia posible y usar nombres inteligibles para tus variables y funciones o métodos. Con la extensión de *ESLint* de Visual Studio Code deberías tener suficiente para validar que el formato y la sintaxis de tu código es la correcta.

Ánimo.