

Cadenas de texto

Atención

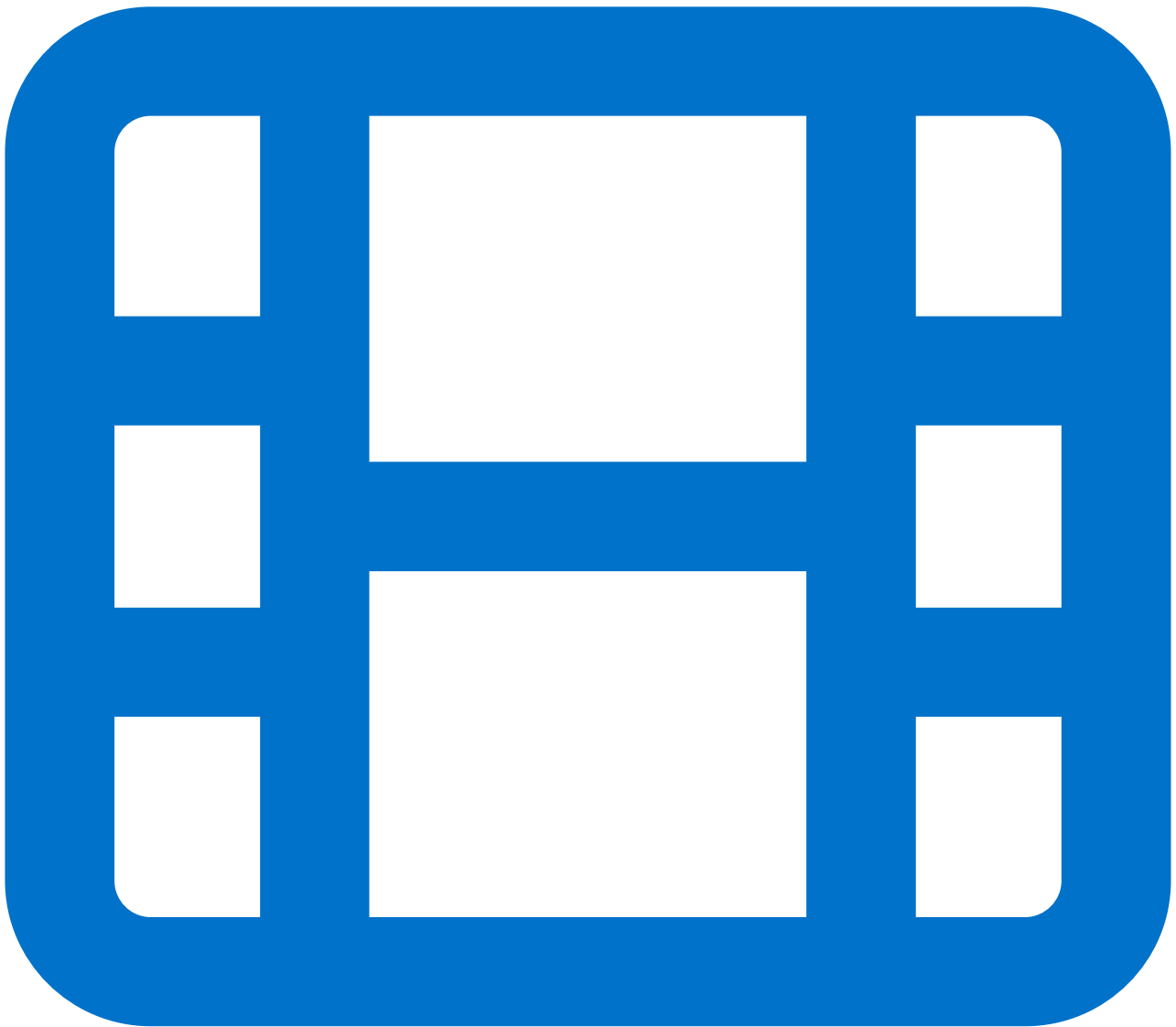
Antes de finalizar su sesión en equipo de hoy, un integrante deberá compartir su pantalla y explicar en detalle cómo resolvió los ejercicios propuestos. Puede ser el facilitador o cualquier compañero que se anime (puedes ser tú 🙋).

Aprovecha esta oportunidad para enriquecer tu propia resolución y participa activamente haciendo preguntas.



Material Audiovisual

Mira el siguiente video para reforzar el contenido teórico y ayudar a resolver las actividades más rápido:



Métodos String | Egg

Actividad: Conteo de caracteres

1. Desde Visual Studio Code, abre la carpeta **example** desarrollada en secciones anteriores.
2. Crea un nuevo script `lengthOf.js`
3. Comenta los otros scripts y conecta `lengthOf.js` con `index.html` del proyecto `example`
4. Define una función `lengthOf` que:
 1. Reciba como parámetro una cadena de texto
 2. Retorne la longitud de la cadena (propiedad `length`)
 3. Si recibe una cadena vacía, devuelva el mensaje "Invalid input".

5. Asegúrate de ejecutar la función y mostrar en la consola los resultados al menos tres veces.
6. Abre con live server el archivo index.html y verifica las salidas programadas.

*console.log(lengthOf("hello")) debe imprimir 5 y
console.log(lengthOf("")) debe imprimir "Invalid input"*

Actividad: Formato de título

Sigue estos pasos para completar la actividad:

1. Desde Visual Studio Code, abre la carpeta **example** desarrollada en secciones anteriores.
2. Crea un nuevo script titleFormat.js
3. Comenta los otros scripts y conecta titleFormat.js con index.html del proyecto example
4. Define una función titleFormat que:
 1. Reciba como parámetro una cadena de texto
 2. Corte la cadena en: primera letra y resto del título (método charAt y slice)
 3. La primera letra la convierta en mayúsculas (método toUpperCase)
 4. El resto en minúsculas (método toLowerCase)
 5. Concatene ambas partes (con +)
 6. Retorne la longitud de la cadena
5. Ejecuta la función y muestra en consola los resultados al menos tres veces.
6. Abre con live server el archivo index.html y verifica las salidas programadas.

*console.log(titleFormat("hello world")), console.log(titleFormat("HELLC
y console.log(titleFormat("HeLLo WOrLd")) deben imprimir "Hello world"*

Actividad: Incluir

1. Desde Visual Studio Code, abre la carpeta **example** desarrollada en secciones anteriores.
2. Crea un nuevo script stringIncludes.js

3. Comenta los otros scripts y conecta stringIncludes.js con index.html del proyecto example
4. Define una función stringIncludes que:
 1. Reciba como parámetro dos cadenas de texto
 2. Verifique si la primera cadena está incluida en la segunda
 3. Si la incluye, devolver true
 4. Si no, devolver false
5. Ejecuta la función y muestra en consola los resultados al menos tres veces.
6. Abre con live server el archivo index.html y verifica las salidas programadas.

console.log(stringIncludes("he", "Hello")) debe imprimir false mientras que console.log(stringIncludes("he", "hello world")) debe imprimir true.



Actividad: Cantidad de caracteres

1. Desde Visual Studio Code, abre la carpeta **example** desarrollada en secciones anteriores.
2. Crea un nuevo script palindrome.js
3. Comenta los otros scripts y conectar palindrome.js con index.html del proyecto example
4. Define una función palindrome que:
 1. Reciba como parámetro una cadena de texto
 2. Rebane la cadena (método split)
 3. Invierta la lista "spliteada" (método reverse)
 4. Una la lista (método join)
 5. Si la cadena de entrada es igual a la cadena de salida, devolver true
 6. Si no, devolver false
5. Ejecuta la función y muestra en consola los resultados al menos tres veces.
6. Abre con live server el archivo index.html y verificar las salidas programadas.

console.log(palindrome("hello")) debe imprimir false mientras que console.log(palindrome("hannah")) debe imprimir true.

Repositorio en GitHub

Puedes subir a un repositorio existente de github la práctica del día de hoy. Los comandos básicos son:

- **git add .** (para preparar todos los archivos creados, modificados o eliminados)
- **git commit -m “nombre del envío”** (para versionar en repositorio local)
- **git push origin main** (o master según corresponda para enviar la nueva versión del proyecto al repositorio remoto)