

# **workbook 2023**



# ¿Qué vas a encontrar en este workbook?

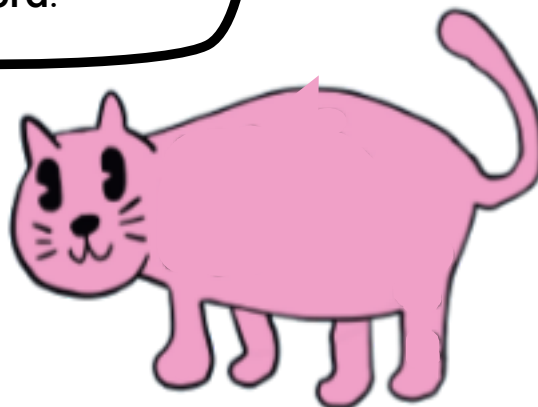
En este Workbook preparamos casi **50 ejercicios de JavaScript** para que practiques todo lo que aprendiste en el curso.

Debajo de cada consigna vas a encontrar **una ayudita** para guiarte en el proceso.

Recordá que **hay distintas maneras de resolver** los ejercicios, lo que nosotros te compartimos es solo un consejo para que sepas por dónde podés empezar.

Al comienzo de cada unidad vas a encontrar **una breve descripción de los temas más importantes**; podés volver a ver los videos y repasarlos antes de comenzar a resolver las consignas.

¡No te olvides de **compartir los ejercicios** que vayas resolviendo **en el canal de Discord!**



Nombre:

Fecha:

# Unidad I

**En esta unidad los temas principales que vas a trabajar son `getElementById`, `value`, `parseInt`, `addEventListener` y `strings`.**

1. Creá un programa que permita al usuario ingresar una lista de números y luego muestre el número más grande y el número más pequeño.

Vas a necesitar un input y un botón. Escuchá el contenido del input con `getElementById` y un `.value`. Gracias a un `addEventListener` en el botón guardá los números ingresados en un array. Para saber cuál es el más grande vas a tener que hacer uso de los bucles.

2. Creá un programa que permita al usuario ingresar una lista de palabras y luego muestre cuántas de ellas comienzan con una letra específica.

Vas a necesitar un input y un botón. Escuchá el contenido del input con `getElementById` y un `.value`. Gracias al `addEventListener` guardá las palabras ingresadas en un array. Para saber con qué letra empiezan, utilizá métodos de `strings`.

3. Creá un programa que permita al usuario ingresar una lista de palabras y luego muestre cuál es la palabra más larga.

Vas a necesitar un input y un botón. Escuchá el contenido del input con `getElementById` y un `.value`. Gracias al `addEventListener` guardá las palabras ingresadas en un array. Para saber cuál es la más larga, utilizá métodos de `strings`.

4. Creá una página web que tenga un botón. Al hacer clic en el botón, se debe mostrar un mensaje de saludo en la página.

Para este programa utilizá un botón con un listener que modifique la propiedad de `display` de un elemento. Así vas a poder pasar de `display none` a `display block` el elemento que necesites.



Nombre:

Fecha:

# Unidad I

5. Creá una página web que tenga dos botones y un cuadro de texto. Al hacer clic en uno de los botones, se debe agregar un texto predeterminado al cuadro. Al hacer clic en el otro botón, se debe borrar el texto.

Para esta web utilizá un botón con un listener que modifique la propiedad de display del elemento que necesites. Con innerHTML agregá y quitá texto.

6. Creá una página web que tenga un botón y una imagen. Al hacer clic en el botón, se debe cambiar el tamaño de la imagen.

Para esta web utilizá un botón con getElementById y repasá los modificadores de propiedades de CSS.

7. Creá una página web que tenga un botón y una imagen. Al hacer clic en el botón, se debe cambiar la imagen por otra imagen.

Para esta web utilizá un botón con getElementById y agregá modificadores de src.

8. Creá una calculadora que calcule el monto de propina que se debe dejar en función del monto de la cuenta y el porcentaje de propina.

Para este ejercicio utilizá parseInt para pasar los campos numéricos a integer. Agregá input text, botones y operaciones aritméticas clásicas en Js para resolver las cuentas.



Nombre:

Fecha:

# Unidad I

9. Creá un juego de adivinanza en el que el usuario debe adivinar un número generado aleatoriamente. El juego debe indicar si el número ingresado por el usuario es mayor o menor que el número generado.

Para este reto utilizá un input, un botón y creá una función que genere un número aleatorio cada vez que se ingresa a la web. Validá la entrada del usuario y parseala como integer.

10. Creá un reloj digital que muestre la hora actual y se actualice automáticamente.

Para este reto vas a tener que investigar un poco cómo utilizar `setInterval` e incluirlo en el ejercicio. ¡Esto va a ser un reto de verdad!

11. Creá un programa que genere contraseñas aleatorias para el usuario. El usuario deberá especificar la longitud de la contraseña y los caracteres que deben incluirse.

Para hacer este reto podrías poner todo el abecedario en un array y, con una función `math.Random`, sacar los caracteres que necesites según la longitud indicada. Para incluir números, creá otro array con números.

12. Creá un formulario que tenga diferentes campos, como nombre, correo electrónico y número de teléfono. Después, validá que los campos estén completados y que el correo electrónico tenga un formato válido.

Para este ejercicio podrías utilizar el atributo `require` de html, pero para que sea más interesante hacelo a mano con Js validando que los input no estén vacíos y que el campo de email incluya un @ y .com



Nombre:

Fecha:

# Unidad I

13. Creá una página web que permita al usuario seleccionar un tema de una lista de opciones. Cada tema deberá cambiar el aspecto visual de la página.

Para este reto utilizá `addClass` y `removeClass` en CSS. Creá estilos diferentes para cada caso y modificá cuál se va a ver en pantalla según la selección del usuario.

14. Creá una app en la que el usuario tenga que ingresar su nombre, apellido y edad. Dependiendo de la edad que tenga, mostrá en pantalla un chiste relacionado con su rango etario. (20 a 30 / 30 a 40 / 40 a 50/ etc.)

Para este reto utilizá un `input` y un botón. Mostrá el chiste con una pantalla emergente. Si querés hacerlo más complejo, podés tener una lista de chistes para cada rango en un `array`.

15. Creá un menú desplegable que se active al hacer clic en un botón. El menú deberá contener varias opciones que lleven a diferentes secciones de la página.

Para este reto utilizá la modificación de propiedades de CSS para mostrar y ocultar el `div` que contenga las opciones.



Nombre:

Fecha:

## Unidad II

**El objetivo de esta unidad es que mejores tus habilidades para manipular el DOM desde JavaScript.**

1. Creá una página con una galería de imágenes. Cuando se hace clic en una imagen, se debe mostrar una vista ampliada de la misma en una pantalla emergente. El usuario debe poder cerrar la pantalla emergente haciendo clic en un botón.

Creá un div que cargue una imagen en mayor tamaño. usá `getElementById`, modificadores de propiedades de CSS, un div flotante que aparezca y desaparezca al hacer click en un botón para cerrar.

2. Creá una página con una barra de navegación. Cuando se hace clic en un elemento de la barra de navegación, se debe mostrar la sección correspondiente de la página y ocultar las demás secciones.

Creá un section para cada una de las pantallas. Todas deben estar en `display none`, luego poné en `display block` la que corresponda. No te olvides de poner en `display none` todas las demás.

3. Creá una página con un formulario de registro. Cuando se hace clic en el botón "Enviar", se debe validar el formulario y mostrar un mensaje de error si hay algún problema. Si el formulario es válido, se debe mostrar un mensaje de confirmación.

Revisá uno por uno los input para chequear si están llenos y completos. Para mostrar el error, utilizá un p oculto que se muestre indicando lo que falta completar. Para que sea más dinámico, utilizá template strings.



Nombre:

Fecha:

## Unidad II

4. Creá una página con una tabla de datos. Cuando se hace clic en una fila de la tabla, se debe mostrar una vista detallada de la información en esa fila. El usuario debe poder cerrar la vista detallada haciendo clic en un botón.

Para este reto utilizá todo lo aprendido sobre cómo cambiar el estado display block a display none de una pantalla

5. Creá una página con un reproductor de video. El usuario debe poder controlar el video haciendo clic en los botones de reproducción y pausa. También debe ser posible mostrar y ocultar la barra de control haciendo clic en un botón.

Para este reto tenés que usar el tag video e investigar sus propiedades para poder manejarlas de JavaScript.

6. Creá una aplicación de generador de horarios en la que el usuario pueda ingresar los detalles de sus clases y la aplicación genere una lista de tareas semanales.

Poné un input en pantalla con un select que contenga los días de la semana para que el usuario pueda ir agregando tareas. Cada día de la semana va en un array que puedas recorrer y mostrar en pantalla.

7. Creá una aplicación de generación de tarjetas de presentación en la que el usuario pueda ingresar su información personal y la aplicación genere una tarjeta de presentación con un diseño seleccionado.

Para este reto utilizá toda tu capacidad creativa para generar una tarjeta con HTML y CSS. Después ingresá los campos con la info requerida y rellena la tarjeta antes de mostrarla en pantalla.





Nombre:

Fecha:

## Unidad II

8. Creá una aplicación de juego de ahorcado en la que el usuario deba adivinar una palabra y la aplicación dé pistas hasta que el usuario adivine correctamente.

Usá un array que contenga palabras random guardadas en un objeto junto con pistas para darle al usuario. Con `math.Random` sacá una palabra y con el método `split` convertí las letras en un nuevo array.

9. Creá una web de juego de aventuras de texto en la que el usuario deba tomar decisiones y avanzar en una historia interactuando con el texto y tomando decisiones.

Utilizá `display none`, `display block`, modificación de `display` con CSS y condicionales lógicos para resolver este ejercicio.

10. Creá una aplicación de generador de nombres de bandas en la que la aplicación combine diferentes palabras y estilos de música para crear un nombre de banda único.

Creá tres arrays con distintos tipos de palabras. Agregá un botón para generar un random que saque una palabra de cada array. Juntalas y mostralas en pantalla con `innerHTML` en un `h1`.

11. Creá una aplicación de calendario en la que el usuario pueda agregar eventos y ver un calendario mensual y semanal.

Retomá la app de tareas que ya creamos, agregale un combo con los meses y guardá en un array todas las tareas asignadas por mes. Podés ayudarte creando un objeto que contenga un grupo de tareas dentro de cada día de la semana.



Nombre:

Fecha:

## Unidad II

12. Creá una aplicación de juego de piedra, papel o tijera en la que el usuario juegue contra la computadora.

Usá todo lo que aprendiste sobre condicionales lógicos. Con un buen uso de if y else podés solucionarlo este reto. Para la parte de la computadora, no te olvides de crear tu IA con un math random.

13. Creá una aplicación de galería de arte en la que el usuario pueda cargar imágenes de internet y verlas en una galería con descripciones.

Utilizá distintos tipos de datos para agregar a la descripción: título, descripción, autor, etc.



Nombre:

Fecha:

## Unidad III

**En esta unidad vas a encontrar principalmente ejercicios sobre arrays map, reduce y filter.**

1. Creá una galería de imágenes en la que el usuario pueda filtrar las imágenes por categoría. Utilizá filter para mostrar solo las imágenes correspondientes a la categoría seleccionada.

Mostrá todas las imágenes en pantalla; agregales una categoría que puedas filtrar con el método filter y volvé a renderizar la vista con las imágenes que correspondan con el uso de map.

2. Creá una página de carrito de compras que muestre todos los productos agregados al carrito y su precio total. Utilizá reduce para calcular el precio total.

Utilizá un array de productos y un array vacío al que vayas agregando los productos a medida que se apriete el botón de agregar. Para la suma utilizá el método reduce.

3. Creá un buscador de películas que filtre las películas por género y año de estreno. Utilizar map y filter para mostrar solo las películas que cumplan con las especificaciones.

Agregá películas en un array, recorrelas con map para incorporar las películas en pantalla. Utilizá un list que tenga las diferentes categorías. Con el método onChange modificá el array.



Nombre:

Fecha:

## Unidad III

4. Creá una página de registro de usuarios que verifique si el usuario ya existe en una lista de usuarios registrados. 5. Creá una página de clasificación de estudiantes que muestre la lista de estudiantes ordenados por promedio de calificaciones.

Usá filter para buscar el nombre de usuario en un array que contenga una lista de usuarios random.

5. Creá una página de resultados que muestre cuántas respuestas correctas seleccionó un participante.

Creá un array que guarde los objetos de cada pregunta y sus respuestas. Usá los checkbox para marcar la respuesta correcta.

6. Creá una página de búsqueda de recetas que filtre las recetas por ingrediente y tiempo de preparación.

Creá una colección de recetas dentro de un array. Recorré el array con un map y muestra las recetas en pantalla. Agregá un select que al cambiar su valor en el método onChange filtre los parámetros con filter y vuelva a ejecutar el map.

7. Creá un juego de preguntas y respuestas en el que el usuario tenga que responder preguntas sobre un tema específico. Utilizá un array de objetos para almacenar las preguntas y las respuestas.

Mostrá 3 preguntas en pantalla. Agregá un campo con nombre y un botón para enviar las respuesta. Cuando se presione el boton, agregá en un array toda la data. Mostrá el resultado con innerHTML en un h1.



Nombre:

Fecha:

## Unidad III

8. Creá una página de clasificación de estudiantes que muestre la lista de estudiantes ordenados por promedio de calificaciones.

Creá un array con un objeto que contenga las propiedades de cada alumno, incluido su promedio. Luego, ordenalos con filter y utilizá map para mostrar el resultado en pantalla.

9. Creá una página de comparación de precios que muestre el precio de un producto en diferentes tiendas en línea. Utilizá reduce para encontrar el precio más bajo.

Creá un array con un objeto que represente el producto para cada una de las tiendas. Utilizá reduce para saber cuál es el más barato y ponelo de manera deestaca haciendo uso del addClass de Js.

10. Creá una página de generador de memes que permita al usuario seleccionar una imagen y agregar texto. Utilizar objetos dentro de arrays para almacenar las imágenes y sus propiedades.

Usá objetos dentro de arrays que contengan la url de la imagen y su tamaño. Gracias a display flex y sus atributos, mostrá la imagen y hacé que el texto quede alineado en la parte inferior al centro.

11. Creá un juego de memoria en el que el usuario tenga que recordar la secuencia de colores que se muestra en la pantalla. Utilizá un array para almacenar la secuencia de colores y Math.random() para generar la secuencia aleatoria.

Generá un array con los colores y un número random. Ejecutá el math.Random 6 veces generando una combiancion de colores diferente en cada jugada. Poné en pantalla una botonera de colores que tengan que respetar el orden del array al ser tocados secuencialmente.



Nombre:

Fecha:

## Unidad III

12. Creá una aplicación de generador de citas inspiradoras en la que el usuario pueda ver una lista de citas y obtener una cita aleatoria cada vez que se actualiza la página.

Creá una base de datos de citas en un array y con el `math.Random` seleccioná una al azar. Mostrala en pantalla con `innerHTML`.

13. Creá una aplicación de lista de canciones favoritas en la que el usuario pueda agregar y eliminar canciones.

¡No hay pistas para este reto!



Nombre:

Fecha:

## Unidad IV

Con esta última unidad incorporamos los conceptos de callback, promesa, async await, fetch y API.

1. Creá una página que muestre la información del clima en tiempo real para una ciudad determinada utilizando una API de clima.

Utilizá API Fetch para consultas unas de las tantas APIs del clima disponibles y consultá los diferentes datos.

2. Creá una página que permita al usuario buscar y mostrar información sobre una película utilizando una API de películas.

Utilizá de API fetch para buscar una película con un slug de búsqueda. Pasalo por la url para que la API entregue los resultados esperados.

3. Creá una página que muestre las últimas noticias de un tema determinado utilizando una API de noticias. La página debe mostrar los titulares de las noticias y permitir al usuario hacer clic en cada uno para leer la historia completa.

Te reto a buscar la forma de robar información de un portal de noticias existente revisando el curso de Flython e incorporando esta tecnología con Js.

4. Creá una página que muestre información sobre el precio actual de una criptomoneda utilizando una API de criptomonedas.

Te reto a que incluyas botones para las 5 criptos más importantes y que, al hacer click en cada, se muestre la info detallada de cada moneda en pantalla.



Nombre:

Fecha:

## Unidad IV

5. Creá una página que muestre información sobre el tráfico en una ciudad determinada utilizando una API de tráfico.

Utilizá API Fetch y recorré el objeto que te entrega la API para encontrar los datos más relevantes y mostralos en pantalla de manera ordenada.

Podés encontrar muchas APIs de uso libre para realizar los ejercicios en la web <https://rapidapi.com/>

