



# Primer programa en JavaScript



# Consigna del día de hoy:

- Programaremos en JavaScript un sitio que nos pida nuestro nombre, y luego nos muestre un mensaje de Bienvenida.



# Archivos necesarios

- Crearemos un archivo con el nombre index.html
- Para esto, podemos usar la terminal de nuestro sistema operativo (Shell o el cmd de Windows)
- En Windows:      **copy con index.html**
- En Linux o Mac: **touch index.html**



# ¿En donde programamos?

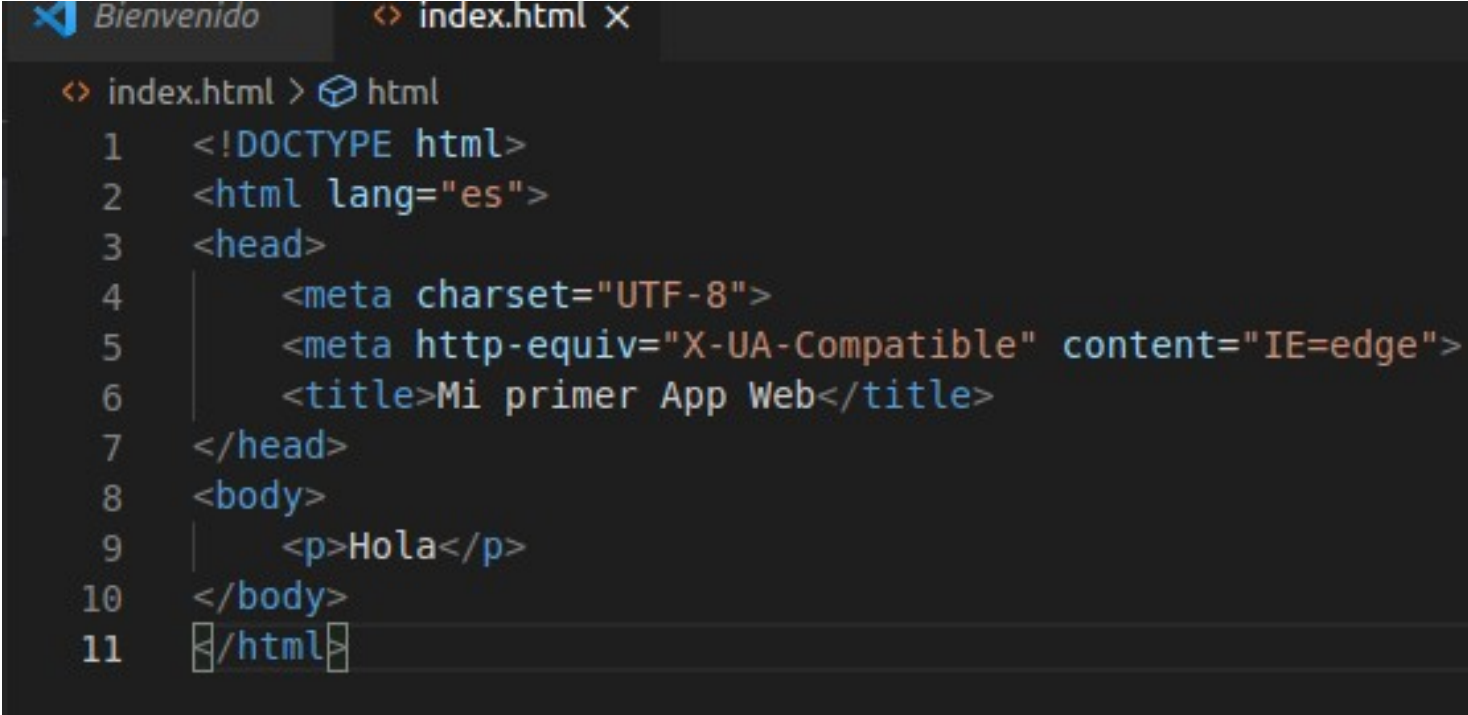
- Podemos usar simplemente el editor o block de notas.
- Podemos usar editores de código profesionales:
- Algunos de ellos son:
  - Visual Studio Code (recomendado)
  - Sublime Text 3
  - Atom
  - Notepad++
  - Otra alternativa es usar editores online como StackBlitz



# Instalación de Visual Studio Code (Opcional)

- Desde su sitio web:
- <https://code.visualstudio.com/download>
- O desde la tienda de aplicaciones (en Linux o Mac)
- La instalación es muy sencilla, y no necesita una mayor explicación...

# index.html - estructura básica:



A screenshot of a code editor window. The title bar shows 'Bienvenido' and 'index.html'. The editor displays the following HTML code:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Mi primer App Web</title>
</head>
<body>
  <p>Hola</p>
</body>
</html>
```

# Incluir código JavaScript en HTML

- Podemos hacerlo de dos formas: (usando la etiqueta script)
- Importar un archivo js externo, con el atributo “src”
- Escribir código javascript dentro del html, dentro de la etiqueta script.

```
<body>
  <p>Hola</p>

  <script>
    alert("Hola");
  </script>

  <script src="funciones.js"></script>
</body>
```



# **windows.prompt()**

- Prompt es un método de la clase windows
- La clase Windows tiene varias propiedades para interactuar con la ventana, la página, o el navegador.
- Mas adelante definiremos el concepto de clase...

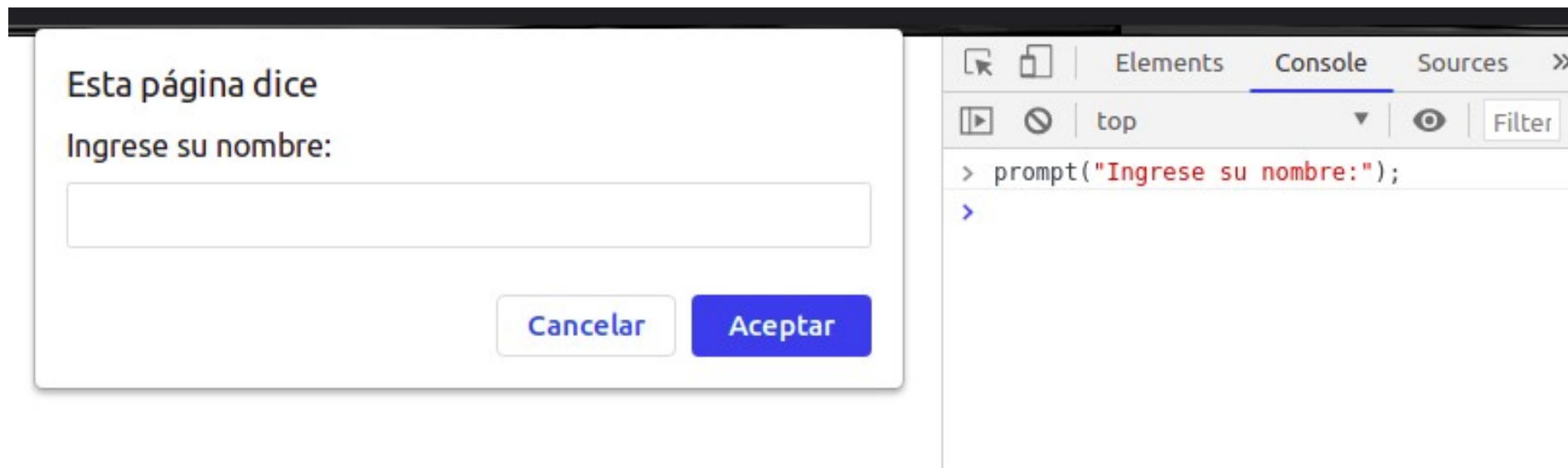




# **windows.prompt()**

- A diferencia del método `alert()` visto la clase pasada, con `prompt()` obtenemos un valor de retorno.
- `Prompt` nos retorna un string, obtenido del texto ingresado en la ventana emergente que hacemos visible al llamar al método `prompt()`
- A su vez, `prompt` puede recibir un string como parámetro, indicando el texto a visualizar en la ventana

# prompt()



# ¿Cómo podemos guardar el valor obtenido de llamar a `prompt()` ?

- Como mencionamos anteriormente, la función `prompt` nos retorna un texto (string)
- Este valor que nos devuelve, una vez que finalice la ejecución de la función `prompt`, se perderá
- Entonces, debemos guardarlo en memoria para usarlo posteriormente



# Variables

- Definición básica:
- Una variable es un espacio reservado de memoria.
- Con las variables podemos guardar datos (de cualquier tipo, en tiempo de ejecución.)
- En JavaScript, las variables se componen de un nombre, y un valor.



# Variables

- Declaración de una variable:
- `var` miVariable;
- Lo que estoy haciendo es declarar una variable (espacio de memoria...), con el nombre miVariable.
- Usamos la palabra reservada `var`
- Por defecto, su valor será undefined

# Variables

- Para asignar un valor a la variable, lo hago con el operador =
- `miVariable = "Holaa";` // le asigno la cadena Holaa
- `miVariable = 10;` // le asigno el valor numérico 10
- `miVariable = true;` // le asigno el valor booleano true
- Mas adelante veremos los tipos de datos en JavaScript

# Variables

- Puedo declarar y asignar un valor, en la misma linea:
- `var nombre = "Joaquin";`
- `var edad = 30;`

# Variables

- Vamos a guardar el valor de retorno de prompt en una variable:
- `var nombre = prompt("Ingrese su nombre");`
- Luego, vamos a mostrarlo en una alerta, pasándole al método alert la variable nombre como parámetro:
- `alert(nombre);`





# Otras formas de declarar variables

- Var es poco usada actualmente, por problemas que explicaremos mas adelante.
- Podemos utilizar las sentencias `let` y `const` para declarar variables.
- Con respecto a `const`, las variables de este tipo no se pueden setear su valor una vez asignado el mismo (son constantes)

# Concatenar texto

- Para mostrar el mensaje de bienvenida que queríamos hacer, debemos unir el texto “Bienvenid@ ” con el valor de la variable nombre.

Usamos el operador +

- “Bienvenid@ ” + nombre
- Esta nueva cadena, se la pasamos como parámetro al metodo alert:
- `alert(“Bienvenid@ ” + nombre);`



# Algunas explicaciones:

- En la programación, se le suele llamar método a todas las propiedades de una clase, o a bloques de código
- Los metodos pueden ser funciones o procedimientos
- Las Funciones son los metodos que retornan un valor.
- Los procedimientos son solo un bloque de código que hace cierta tarea, sin retornar nada.

# Idea intuitiva de métodos y procedimientos

- En javascript es irónico lo mencionado anterior, ya que todos los metodos nos retornan algo (Por defecto nos retornan “undefined”)
- Para declarar tanto funciones como procedimientos usamos la sentencia function:

```
function nombreDeFuncion(parametro1, parametroN) {  
    // Bloque de código de la funcion
```

- }

# Ejemplos:

```
function hacerAlgo() {  
    alert("UwU");  
}
```

```
function incrementaEnDiez(numero) {  
    return (numero + 10);  
}
```

---



# Resumen de hoy:

- Aprendimos a programar JavaScript dentro del HTML con la etiqueta script.
- Aprendimos a usar la funcion primitiva prompt()
- Aprendimos a declarar variables, y asignarles valores.
- Vimos una idea intuitiva de los metodos, procedimientos, y funciones.
- Vimos una idea de como declarar funciones.



# Próxima clase:

- Vamos a seguir con las variables
- Veremos los tipos de datos en JavaScript
- Trabajaremos con los operadores
- Trabajaremos con expresiones

# Tarea para hacer de deberes:

- Modifique el programa de hoy, agregue otro input que pida el apellido del usuario.
- Concatenar las tres cadenas y mostrarlas con alert()
- **Nota:** El operador de concatenación ( + ) lo podemos usar para concatenar todas las cadenas que queramos
- Ejemplo:  
`var cadenas = "Cadena 1 " + "cadena 2 " + "cadena n";`