



# Certified Tech Developer

The Ultimate Degree

## Front End III

# Agregando React a una página existente

En esta primera clase vamos a ver de qué manera podemos agregar React a un proyecto existente.

Para ello, partiremos de un archivo index.html con una estructura básica:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>React en mi HTML</title>
</head>
<body>
  <h1>React en mi HTML</h1>
</body>
</html>
```

El primer paso será agregar una etiqueta <div> vacía para marcar el lugar donde deseemos visualizar algo con React. En nuestro caso, lo haremos debajo del título:

```
<body>
  <h1>React en mi HTML</h1>

  <!-- Agregamos la etiqueta -->
  <div id="button_container"></div>
```



# Certified Tech Developer

The Ultimate Degree

```
</body>
```

Un dato a tener en cuenta es que a este `<div>` le hemos agregado un atributo HTML "id" único. Este punto es clave, ya que dicho atributo nos permitirá luego seleccionar este elemento desde nuestro código JavaScript para poder insertar un componente de React dentro de él.

Ahora que ya tenemos nuestro punto de entrada a React, vamos a agregar tres `<script>` a la página HTML justo antes de cerrar la etiqueta `</body>`:

```
<body>
  <h1>React en mi HTML</h1>

  <div id="button_container"></div>

  <!-- Cargar React. -->
  <!-- Nota: En producción, debemos reemplazar "development.js" con
"production.min.js". -->
  <script
    src="https://unpkg.com/react@18/umd/react.development.js"
    crossorigin>
  </script>
  <script
src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"
    crossorigin>
  </script>
  <!-- Cargamos nuestro componente de React. -->
  <script src="button.js"></script>
</body>
```

Las primeras dos etiquetas cargan React. La tercera, carga el código del componente



# Certified Tech Developer

The Ultimate Degree

que crearemos utilizando la librería.

Ahora que ya tenemos el setup inicial realizado, vamos a crear nuestro primer componente. No hay que preocuparse si todavía no entendemos esta parte, más adelante nos ocuparemos de entender qué es un componente y cómo funciona. Por ahora, podemos crear un archivo "button.js" a la misma altura que el index.html, y dentro del mismo pegar el siguiente código:

```
// Este método nos permite crear elementos dentro de React.
const e = React.createElement;

// ¡Esto es un componente React!
const Button = () => e(
  'button',
  { onClick: () => alert("Has hecho click!") },
  'Haz Click!'
);

// Seleccionamos el div dentro de nuestro index.html
const domContainer = document.querySelector('#button_container');

// Aquí llamamos a un método de React que crea la base
// para poder utilizar la librería dentro de nuestro <div>.
const root = ReactDOM.createRoot(domContainer);

// Con el método render vamos a mostrar el componente Button
root.render(e(Button));
```

Si guardamos nuestros cambios y abrimos el archivo index.html en el navegador, veremos nuestro botón en la pantalla. Además, si hacemos clic, veremos el mensaje de alerta que agregamos:



# Certified Tech Developer

The Ultimate Degree

## React en mi HTML

Haz Click!

¡Felicitaciones! Hemos creado nuestro primer componente en React.