

## CURSO DE REACT BÁSICO DE FREECODECAMP

**React:** es una biblioteca de JavaScript de código abierto diseñada para crear interfaces de usuario.

**Biblioteca:** conjunto de implementaciones o sub-programas que podemos usar en nuestro código.

Ventajas de React:

- fácil de aprender y usar
- componentes reutilizables
- crear aplicaciones dinámicas
- buen desempeño

**Componente:** parte de la interfaz de usuario que es independiente y reutilizable

componentes  $\left\{ \begin{array}{l} \text{funcionales} \Rightarrow \text{nuevos} \\ \text{de clase} \Rightarrow \text{anteriores} \end{array} \right.$

**Componente funcional:** es una función de JavaScript ecmaScript 6 que retorna un elemento de React (JSX)

```
function Saludo (props) {  
  return (  
    <h1>Hola, {props.nombre}!</h1>  
  )  
}
```

Características de componentes funcionales:

- debe retornar un elemento de React (JSX)
- debe comenzar con una letra mayúscula
- puede recibir valores si es necesario

**Props:** Argumentos que puede recibir un componente de React, los props solo pueden ser enviados de padre a hijo.

**Componente de Clase.**

Clase de ES6 que retorna un elemento JSX

```
class Saludo extends React.component {  
  render() {  
    return <h1>Hola, {this.props.nombre}!</h1>  
  }  
}
```

**Estado:** Representación de JavaScript del conjunto de propiedades de un componente y sus valores actuales.

**Hook:** Función especial que te permite trabajar con estados en componentes funcionales y otros aspectos de React sin escribir un componente de Clase. Esto nos permite escribir código mucho más conciso y fácil de entender.

**Event Listener / Event Handler:** Función en JavaScript que es ejecutada cuando ocurre un evento específico. (por ejemplo un botón)

## Introducción a JSX

**JSX:** Es una extensión de React para la sintaxis de JavaScript, nos permite describir en JavaScript cómo se verán los componentes usando una estructura similar a HTML

Ventajas:

- estructura mas fácil de visualizar
- errores y advertencias mas útiles

**Elementos:** unidades mas pequeñas de React, detienen lo que se ve en la pantalla

“Los componentes de React están hechos de elementos”

**DOM:** es una representación en el navegador de todos los componentes que conforman una pagina o aplicación web.

Elemento = `<button>`

componente = `<button>`

**Atributos:**

	camelCase
class =>	className
for =>	htmlFor
backgroun-image=>	backgroundImage
font-family=>	fontFamily

Const estiloDiv = { color: 'yellow',  BackgroundColor: 'black' }	<code>&lt;div style={estiloDiv}&gt;Hola mundo!&lt;/div&gt;</code>
	<code>&lt;div style={{color:'yellow'}}&gt;Hola mundo!&lt;/div&gt;</code>

Creando una aplicación de React:

- |                                    |                                     |
|------------------------------------|-------------------------------------|
| 1. instalar git                    | (windows,mac,linux)                 |
| 2. instalar nodejs                 | (windows,mac,linux)                 |
| 3. npm i -g create-react-app       | (instala create-react-app)          |
| 4. npx create-react-app hola-mundo | (crea un proyecto de react)         |
| 5. cd hola-mundo                   | (entra en la carpeta del proyecto)  |
| 6. npm start                       | (inicial el servidor de desarrollo) |

Elementos a eliminar:

setupTest.js   reportWebVitals.js   App.test.js  
dentro de index.js => Borrar todo lo de reportWebVitals  
dentro de App.css => Borrar todo

## Desestructuración

{props}	{texto}
props.texto	{texto}

## Operador Ternario

{esBotonClic ? 'boton-clic' : 'boton-reiniciar'}

**Math.js** => npm install mathjs

**React Icons** => npm install react-icons --save

**Fragmentos** => `<div>` => `<></>`

**map:** es un metodo, que va a tomar cada una de las tareas como el argumento de una fucion y va a realizar lo que especifiquemos.

**Uuid:** se utiliza para identificadores unicos

=> npm install uuid

**key:** es una forma que tiene React para identificar un elemento y debe ser único (ordena los elementos)

## PROYECTO TESTIMONIOS

```
├── App.css
├── App.js
├── componentes
│   └── Testimonio.jsx
├── hojas-de-estilo
│   └── Testimonio.css
├── imagenes
│   ├── testimonio-emma.png
│   ├── testimonio-sarah.png
│   └── testimonio-shawn.png
├── index.css
├── index.js
└── logo.svg
```

- `npx create-react-app testimonios`
- `cd testimonios`
- `mkdir componentes`
- `touch componentes/Testimonio.jsx`
- `mkdir hojas-de-estilo`
- `touch hojas-de-estilo/Testimonio.css`
- `mkdir imagenes`
- (copiar imagenes a esta carpeta)
- borrar todo dentro de `App.css`
- borrar `setupTest.js`
- borrar `reportWebVitals.js`
- borrar `AppTest.js`
- borrar dentro de `index.js => reportWebVitals`

### componentes/Testimonio.jsx

```
import React from "react"
import '../hojas-de-estilo/Testimonio.css'

function Testimonio(props){
  return(
    <div className='contenedor-testimonio'>
      <img
        className='imagen-testimonio'
        src={require(`../imagenes/testimonio-${props.imagen}.png`)}
        alt='Foto de Emma'>
      <div className='contenedor-texto-testimonio'>
        <p className="nombre-testimonio"><strong>{props.nombre}</strong> en
{props.pais}</p>
        <p className='cargo-testimonio'>{props.cargo} en
<strong>{props.empresa}</strong></p>
        <p className='texto-testimonio'>"{props.testimonio}"</p>
      </div>
    </div>
  )
}

export default Testimonio
```

## hojas-de-estilo/Testimonio.css

```
.contenedor-testimonio{
  width: 1170px;
  height: 368px;
  margin:10px 10px 50px;
  display:flex;
  align-items:center;
  justify-content:center;
  background-color: #fff;
  box-shadow:0 3px 13px 1px rgb(0 0 0 / 9%);
}

.imagen-testimonio{
  height: 100%;
  width: auto;
}

.contenedor-texto-testimonio{
  text-align: center;
  padding:40px;
  font-family:Lato, sans-serif;
  font-weight:400;
  line-height: 1.5rem;
}

.nombre-testimonio{
  font-size:1.3rem;
}

.cargo-testimonio{
  font-size:1.3rem;
  margin-top:15px;
  padding-bottom:30px;
}

.texto-testimonio{
  font-size:1.1rem;
  text-align: justify;
}

/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (max-width: 600px) {
  .contenedor-testimonio{
    display:flex;
    flex-direction:column;
    height: 100%;
    width: auto;
    margin:auto;
  }
  .contenedor-texto-testimonio{
    padding:auto;
  }
}
```

## App.js

```
import './App.css';
import Testimonio from './componentes/Testimonio';

function App() {
  return (
    <div className="App">
      <div className="contenedor-principal">
        <h1>Esto es lo que dicen nuestros alumnos sobre freeCodeCamp:</h1>
        <Testimonio
          nombre='Shawn Wang'
          pais='Singapur'
          imagen='shawn:'          cargo='Ingeniero de Software'
          empresa='Amazon'
          testimonio='Da miedo cambiar de carrera. Solo gané la confianza de que podía programar
trabajando a través de los cientos de horas de lecciones gratuitas en freeCodeCamp. Dentro de un año
tuve un trabajo de seis cifras como ingeniero de software.freeCodeCamp cambió mi vida.'
        />
        <Testimonio
          nombre='Sarah Chima'
          pais='Nigeria'
          imagen='sarah'
          cargo='Ingeniera de Software'
          empresa='ChatDesk'
          testimonio='freeCodeCamp fue la puerta de entrada a mi carrera como desarrollador de
software. El plan de estudios bien estructurado llevó mis conocimientos de programación de un nivel
de principiante total a un nivel muy seguro. Era todo lo que necesitaba para conseguir mi primer
trabajo de desarrollador en una empresa increíble.'/>

        <Testimonio
          nombre='Emma Bostian'
          pais='Suecia'
          imagen='emma'
          cargo='Ingeniera de Software'
          empresa='Spotify'
          testimonio='Siempre he tenido problemas para aprender JavaScript. He tomado muchos
cursos, pero el curso de freeCodeCamp fue el que se quedó. Estudiar JavaScript, así como estructuras
de datos y algoritmos en freeCodeCamp me dio las habilidades y la confianza que necesitaba para
conseguir el trabajo de mis sueños como ingeniero de software en Spotify.'/>

      </div>
    </div>
  );
}

export default App;
```

## PROYECTO CONTADOR



- npx create-react-app contador
- cd contador
- mkdir componentes
- touch componentes/Boton.jsx
- touch componentes/Contador.jsx
- mkdir hojas-de-estilo
- touch hojas-de-estilo/Boton.css
- touch hojas-de-estilo/Contador.css
- mkdir imagenes
- (copiar imagenes a esta carpeta)
- borrar todo dentro de App.css
- borrar setupTest.js
- borrar reportWebVitals.js
- borrar AppTest.js
- borrar dentro de index.js => reportWebVitals

### componentes/Boton.js

```
import React from "react"
import '../hojas-de-estilo/boton.css'
function Boton({ texto, esBotonDeClic, manejarClic }){

  return(
    <button
      className={esBotonDeClic ? 'boton-clic' : 'boton-reiniciar'}
      onClick={manejarClic}>
        {texto}
      </button>
    )
  }

export default Boton
```

### componentes/Contador.jsx

```
import React from "react"
import '../hojas-de-estilo/Contador.css'

function Contador({ numClics }){
  return(
    <div className='contador'>
      {numClics}
    </div>
  )
}

export default Contador
```

## hojas-de-estilo/Boton.css

```
button{
  width:250px;
  height:80px;
  font-size:35px;
  font-family:Roboto, sans-serif;
  font-weight:bold;
  color:white;
  border:2px solid white;
  border-radius:15px;
  margin:5px;
  cursor:pointer;
}

button:hover{
  background-color:#00471b;
}

.boton-clic{
  background-color:#002ead;
}

.boton-reiniciar{
  background-color:#5a01a7;
}

/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (max-width: 600px) {
  button{
    height: auto;
    width: 50%;
  }
}
```

## hojas-de-estilo/Contador.css

```
.contador{
  min-width: 300px;
  height: 300px;
  font-size:160px;
  padding-bottom:25px;
  color:white;
  display:flex;
  align-items:center;
  justify-content:center;
  border: 8px solid white;
  margin-bottom:15px;
}

/* Small devices (portrait tablets and large phones, 600px and up) */
@media only screen and (max-width: 600px) {
  .contador{
    height: auto;
    width: 10%;
  }
}
```

## App.js

```
import './App.css';
import freeCodeCampLogo from './imagenes/fcc_primary_large.png';
import Boton from './componentes/Boton'
import Contador from './componentes/Contador'
import { useState } from 'react';

function App() {

  const [numClics,setNumClics]= useState(0);

  const manejarClic = () =>{
    setNumClics(numClics+1);
  }

  const reiniciarContador = () =>{
    setNumClics(0);
  }

  return (
    <div className="App">
      <div className='freecodecamp-logo-contenedor'>
        <img
          className='freecodecamp-logo'
          src={freeCodeCampLogo}
          alt='logo de freecodecamp' />
        </div>
        <div className='contenedor-principal'>
          <Contador numClics={numClics}/>
          <Boton
            texto='Clic'
            esBotonDeClic={true}
            manejarClic={manejarClic}/>
          <Boton
            texto='Reiniciar'
            esBotonDeClic={false}
            manejarClic={reiniciarContador}/>
        </div>
      </div>
    );
  }

  export default App;
```



## PROYECTO CALCULADORA

```
├── App.css
├── App.js
├── componentes
│   ├── BotonClear.jsx
│   ├── Boton.jsx
│   ├── LogoFreecodecamp.jsx
│   └── Pantalla.jsx
├── hojas-de-estilo
│   ├── BotonClear.css
│   ├── Boton.css
│   └── Pantalla.css
├── imagenes
│   └── fcc_primary_large.png
├── index.css
├── index.js
└── logo.svg
```

- npx create-react-app calculadora
- cd calculadora
- mkdir componentes
- touch componentes/Botonclear.jsx
- touch componentes/Boton.jsx
- touch componentes/LogoFreecodecamp.jsx
- touch componentes/Pantalla.jsx
- mkdir hojas-de-estilo
- touch hojas-de-estilo/BotonClear.css
- touch hojas-de-estilo/Boton.css
- touch hojas-de-estilo/Pantalla.css
- mkdir imagenes
- (copiar imagenes a esta carpeta)
- borrar todo dentro de App.css
- borrar setupTest.js
- borrar reportWebVitals.js
- borrar AppTest.js
- borrar dentro de index.js => reportWebVitals

### componentes/Boton.jsx

```
import React from "react"
import '../hojas-de-estilo/Boton.css'

function Boton(props){

  const esOperador = valor => {
    return isNaN(valor) && (valor !== '.') && (valor !== '=');
  }

  return(
    <div className={`boton-contenedor ${esOperador(props.children) ? 'operador' : ''}`.trimEnd()}
    onClick={() => props.manejarClic(props.children)}>
      {props.children}
    </div>
  )
}

export default Boton
```

### componentes/BotonClear.jsx

```
import React from "react"
import '../hojas-de-estilo/BotonClear.css'

function BotonClear(props){
  return(
    <div className='boton-clear' onClick={props.manejarClear}>
```

```
    {props.children}
  </div>
)
}
```

```
export default BotonClear
```

### **componenttest/LogoFreecodecamp.jsx**

```
import React from "react"
import logofreecodecamp from '../imagenes/fcc_primary_large.png'

function LogoFreecodecamp(){
  return(
    <div className='freecodecamp-logo-contenedor'>
      <img className='freecodecamp-logo' src={logofreecodecamp} alt='Logo de FreeCodeCamp' />
    </div>
  )
}

export default LogoFreecodecamp
```

### **componenttest/Pantalla.jsx**

```
import React from "react"
import '../hojas-de-estilo/Pantalla.css'

function Pantalla({input}){
  return(
    <div className='input'>
      {input}
    </div>
  )
}

export default Pantalla
```

### **hojas-de-estilo/Boton.css**

```
.boton-contenedor{
  height:83px;
  display:flex;
  flex:1 1;
  justify-content:center;
  align-items:center;
  font-weight:bold;
  font-size:38px;
  background-color:#1b1b32;
  color:white;
  border-radius:25px;
  border: 2px solid white;
  margin:4px;
  cursor:pointer;
  user-select:none;
```

```
}

.boton-contenedor:hover{
    background-color:#5a01a7;
}

.operador{
    background-color:#00471b;
}

.operador:hover{
    background-color:#ff6600;
}
```

### hojas-de-estilo/BotonClear.css

```
.boton-clear{
    height:80px;
    font-size:1.6em;
    display:flex;
    flex:1;
    background-color:#ac0246;
    margin-top:8px;
    justify-content:center;
    align-items: center;
    font-weight: bold;
    color:white;
    border: 2px solid white;
    border-radius:45px;
    cursor:pointer;
    user-select: none;
}

.boton-clear:hover{
    background-color:#ebb000;
}
```

### hojas-de-estilo/Pantalla.css

```
.input{
    height:75px;
    border-radius:50px;
    margin:5px 5px 10px 5px;
    display:flex;
    justify-content:flex-end;
    align-items:center;
    font-weight: bold;
    font-size: 30px;
    background-color:#0a0a23;
    color:white;
    padding:11px 30px 11px 11px;
    border:1px solid #888;
    box-shadow: -1px 4px 1px 1px white;
}
```

## PROYECTO TODOLIST

```
├── App.css
├── App.js
├── componentes
│   ├── ListaDetareas.jsx
│   ├── TareaFormulario.jsx
│   └── Tarea.jsx
├── hojas-de-estilo
│   ├── ListaDeTareas.css
│   ├── Tarea.css
│   └── TareaFormulario.css
├── imagenes
│   └── fcc_primary_large.png
├── index.css
├── index.js
└── logo.svg
```

- npx create-react-app todolist-react
- cd todolist-react
- mkdir componentes
- touch componentes/ListaDetareas.jsx
- touch componentes/TareaFormulario.jsx
- touch componentes/Tarea.jsx
- mkdir hojas-de-estilo
- touch hojas-de-estilo/ListaDeTareas.css
- touch hojas-de-estilo/Tarea.css
- touch hojas-de-estilo/TareaFormulario.css
- mkdir imagenes
- (copiar imagenes a esta carpeta)
- borrar todo dentro de App.css
- borrar setupTest.js
- borrar reportWebVitals.js
- borrar AppTest.js
- borrar dentro de index.js => reportWebVitals

### componentes/Tarea.jsx

```
import React from "react"
import '../hojas-de-estilo/Tarea.css'
import { AiOutlineCloseCircle } from 'react-icons/ai'

function Tarea({ id, texto , completada, completarTarea, eliminarTarea }){
  return(
    <div className={completada ? 'tarea-contenedor completada' : 'tarea-contenedor'}>
      <div
        className='tarea-texto'
        onClick={() => completarTarea(id)}>
        { texto }
      </div>
      <div
        className='tarea-contenedor-iconos'
        onClick={()=> eliminarTarea(id)}>
        <AiOutlineCloseCircle className='tarea-icone' />
      </div>
    </div>
  )
}

export default Tarea
```

### componentes/TareaFormulario.jsx

```
import React, {useState} from "react"
import '../hojas-de-estilo/TareaFormulario.css'
import {v4 as uuidv4} from 'uuid'

function TareaFormulario(props){
  const [input, setInput] = useState("");

  const manejarCambio = e =>{
    setInput(e.target.value);
  }

  const manejarEnvio = e =>{
    e.preventDefault();

    const tareaNueva = {
      id:uuidv4(),
      texto:input,
      completada:false
    }
    props.onSubmit(tareaNueva);
  }

  return(
    <form
      className='tarea-fomulario'
      onSubmit={manejarEnvio}>
      <input
        className='tarea-input'
        type='text'
        placeholder='Escribe una tarea'
        name='texto'
        onChange={manejarCambio}
      />
      <button className='tarea-boton'>
        Agregar Tarea
      </button>
    </form>
  )
}

export default TareaFormulario
```

## componenttest/ListaDeTareas.jsx

```
import React, { useState } from "react"
import TareaFormulario from './TareaFormulario'
import Tarea from './Tarea';
import '../hojas-de-estilo/ListaDeTareas.css'

function ListaDetareas(){
  const [tareas, setTareas] = useState([]);
  const agregarTarea = tarea =>{
    if (tarea.texto.trim()) {
      tarea.texto = tarea.texto.trim();
      const tareasActualizadas = [tarea, ...tareas];
      setTareas(tareasActualizadas);
    }
  }

  const eliminarTarea = id =>{
    const tareasActualizadas = tareas.filter(tarea => tarea.id !== id);
    setTareas(tareasActualizadas);
  }

  const completarTarea = id =>{
    const tareasActualizadas = tareas.map(tarea => {
      if (tarea.id === id) {
        tarea.completada = !tarea.completada;
      }
      return tarea;
    });
    setTareas(tareasActualizadas);
  }

  return(
    <>
    <TareaFormulario onSubmit={agregarTarea} />
    <div className='tareas-lista-contenedor'>
      {
        tareas.map((tarea) =>
          <Tarea
            key={tarea.id}
            id={tarea.id}
            texto={tarea.texto}
            completada={tarea.completada}
            completarTarea={completarTarea}
            eliminarTarea={eliminarTarea}
          />
        )
      }
    </div>
    </>
  )
}

export default ListaDetareas
```

## hojas-de-estilo/Tarea.css

```
.tarea-contenedor{
  width:500px;
  min-height:65px;
  background-color:#1b1b32;
  margin: 5px 0;
  padding:8px 15px 8px 20px;
  display:flex;
  justify-content:space-between;
  align-items:center;
  border-radius:7px;
  border: 1px solid white;
  cursor:pointer;
}

.tarea-contenedor:nth-child(3n +1){
  background-color:#1b1b32;
}

.tarea-contenedor:nth-child(3n +2){
  background-color:#2a2a40;
}

.tarea-contenedor:nth-child(3n +3){
  background-color:#3b3b4f;
}

.tarea-texto{
  width:100%;
  height:100%;
  font-size:20px;
  display:flex;
  align-items:center;
  overflow-wrap:anywhere;
}

.tarea-icone{
  width:25px;
  height:25px;
  margin:5px;
}

.tarea-contenedor.completada{
  background-color:#5a01a7;
  text-decoration: line-through;
}
```

## hojas-de-estilo/TareaFormulario.css

```
.tarea-formulario{
  display:flex;
  flex-wrap:wrap;
  align-items:center;
  justify-content:center;
}

.tarea-input{
  width:350px;
  font-size:18px;
  background-color: white;
  padding:14px 32px 14px 16px;
  border-radius: 4px 0 0 4px;
  border: 2px solid #00471b;
  outline:none;
}

.tarea-boton{
  padding:16px;
  font-size:18px;
  font-family:Roboto,sans-serif;
  border:none;
  border-radius:0 4px 4px 0;
  cursor:pointer;
  outline:none;
  background:#00471b;
  color:white;
  text-transform: capitalize;
}
```

## hojas-de-estilo/ListaDeTareas.css

```
.tareas-lista-contenedor{
  display:flex;
  flex-wrap:wrap;
  justify-content:center;
  margin-top:15px;
}
```



## App.js

```
import './App.css';
import freeCodeCampLogo from './imagenes/fcc_primary_large.png'
import ListaDetareas from './componentes/ListaDetareas'

function App() {
  return (
    <div className="aplicacion-tareas">
      <div className='freecodecamp-logo-contenedor'>
        <img
          src={freeCodeCampLogo}
          className='freecodecamp-logo'
          alt='Logo FreeCodeCamp' />
        </div>
        <div className='tareas-lista-principal'>
          <h1>Mis Tareas</h1>
          <ListaDetareas />
        </div>
      </div>
    );
  }
}

export default App;
```