



**KEEP  
CALM  
AND  
EMPEZAMOS  
EN BREVE**

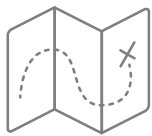


# REACT ROUTER

DigitalHouse >  
Coding School



Renderizando distintos componentes en nuestra App



# ESTRUCTURA DEL ENCUENTRO

USTED  
ESTÁ  
AQUÍ



BIENVENIDA  
E INTRO

INSTALACIÓN Y  
SETUP

USO EN  
COMPONENTES

CIERRE Y  
PREGUNTAS

# Routing

## ¿Qué es y para qué sirve?

Routing o “enrutamiento” añade una capa de abstracción entre la url y los archivos que componen la aplicación.

Por medio de él determinamos cómo una aplicación responde a una petición del cliente que lo hace a través de un URI con un método de solicitud HTTP específico.

Es decir, según la URL, podríamos querer mostrar contenido distinto...

# React Router

<https://reacttraining.com/react-router/>

Es una librería que nos permite gestionar las rutas en nuestra aplicación hecha en React.

Esta librería nos facilita decidir qué componentes renderizar dependiendo de la ruta.



# Instalación

```
npm install react-router-dom
```

Generalmente no instalamos el paquete **react-router** de forma directa. Esta librería ofrece otros paquetes que proporcionan componentes específicos para nuestro entorno de desarrollo (react-router-native para mobile apps, react-router-dom para web apps).



# Browser Router

Lo utilizamos cuando contamos con servidores que manejan peticiones dinámicas.

js

```
<BrowserRouter>
```

```
<App/>
```

```
</BrowserRouter>
```

BrowserRouter recibe un solo componente secundario.

# Browser Router

Lo utilizamos cuando contamos con servidores que manejan peticiones dinámicas.

js

```
import ReactDOM from 'react-dom';
import { BrowserRouter } from 'react-router-dom';

ReactDOM.render(
  <BrowserRouter>
    <App/>
  </BrowserRouter>
  , document.getElementById('root')
);
```



# Link

Componente para gestionar los "enlaces" de la App.

js

```
<Link to='/about'> About </Link>
```

"Enlace" para cambiar la URL de nuestra app

# Link

Componente para gestionar los "enlaces" de la App.

```
import ReactDOM from 'react-dom';
import { Link } from 'react-router-dom';

const Navbar = () => {
  <nav>
    <ul>
      <li> <Link to='/'> Home </Link> </li>
      <li> <Link to='/about'> About Us </Link> </li>
      <li> <Link to='/contact'> Contact </Link> </li>
    </ul>
  </nav>
}

export default Navbar;
```

js

# NavLink

Componente para gestionar los "enlaces" de la App.  
Con algunas props extras.

js

```
<NavLink to='/about'> About </NavLink>
```

"Enlace" para cambiar la  
URL de nuestra app

# NavLink

Componente para gestionar los "enlaces" de la App.  
Con algunas props extras.

```
import ReactDOM from 'react-dom';
import { NavLink } from 'react-router-dom';

const Navbar = () => {
  <nav>
    <ul>
      <li> <NavLink to="/" activeClassName='selected'> Home </NavLink> </li>
      <li> <NavLink to="/about" activeClassName='selected'> About Us </NavLink> </li>
      <li> <NavLink to="/contact" activeClassName='selected'> Contact </NavLink> </li>
    </ul>
  </nav>
}

export default Navbar;
```

js

# Route

Componente para renderizar un componente dependiendo de la ruta.

js

```
<Route path='/about' component={ AboutComp } />
```

Si la URL hace "match" entonces se renderizará el componente "AboutComp"

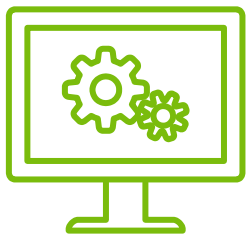
# Route

Componente para renderizar un componente dependiendo de la ruta.

```
import ReactDOM from 'react-dom';
import { Switch, Route } from 'react-router-dom';

const Contenido = () => {
  <Switch>
    <Route path="/" component={ Home } />
    <Route path="/about" component={ About } />
    <Route path="/contact" component={ Contact } />
  </Switch>
}
```

js



¡BASTA DE  
CHARLA!  
ES HORA DE  
**PROGRAMAR**



RECUERDEN  
LA **PRÁCTICA**  
HACE AL  
**MACACO**  
**MAESTRO**







**¡GRACIAS!**

¿Preguntas?

[javier@digitalhouse.com](mailto:javier@digitalhouse.com)