

React: Rutas con parámetros

Como vimos anteriormente, si queremos que nuestra aplicación tenga navegación entre componentes tendremos que instalar el paquete **react-router-dom** e importar los módulos **BrowserRouter**, **Route** y **Link**.

Utilizamos `<BrowserRouter></BrowserRouter>` para envolver todos nuestros componentes. Dentro de cada etiqueta `<Route></Route>` pondremos el componente que queremos que se muestre en función de la ruta a la que estemos navegando:

```
App.js
<BrowserRouter>
  <Routes>
    <Route path='/saludo' element={<Saludo />}/>
  </Routes>
</BrowserRouter>
```

En esa ruta nos aparecerá el componente Saludo cuando vayamos a la ruta `"/saludo"`.

Igual que en express podíamos decirle a nuestra aplicación que estuviese pendiente de los parámetros que pudiesen llegar en la ruta mediante `/ruta/:parametro`, en React haremos algo parecido. En la etiqueta `<Route>` en el atributo **path** podemos definir un parámetro de la misma manera que hacíamos en express:

```
App.js
<Route path='/saludo/:nombre' element={<Saludo />}>
```

Ahora, en el componente Saludo tendremos que decirle que recoja esos parámetros para poder utilizarlos. Para eso importamos el módulo **useParams** del paquete **react-router-dom**:

```
Saludo.js
import {useParams} from 'react-router-dom';
```

Dentro de este componente, usaremos ese módulo para capturar los parámetros que nos llegan en la ruta.

```
Saludo.js
let params = useParams();
```

En la variable **params** tendremos un objeto con los parámetros que hemos definido en la etiqueta `<Route>` en el componente **App**. Le habíamos llamado **nombre**, por lo que ahora en el componente **Saludo** recogeremos ese parámetro como **nombre**:

```
App.js
```

```
<Route path='/saludo/:nombre' element={<Saludo />}>
```

Saludo.js

```
let nombre = params.nombre;
```

Ahora ya podemos utilizar como queramos el valor que nos llega al componente **Saludo**.

```
import {useParams} from 'react-router-dom';
```

```
function Saludo(){  
  let params = useParams();  
  let nombre = params.nombre;  
  
  return (  
    <p>{nombre}</p>  
  )  
}
```