

# React Router Exercises

## functions as props

### challenge

<https://codesandbox.io/s/lifting-state-up-challenge-ci1u64>

### solution

<https://codesandbox.io/s/lifting-state-up-solution-6xpvsr>

### understanding

- You can create stateless components. Advantage of creating stateless components is that you can re-use that component with different data.
- Events listeners like onClick can only be applied to standard HTML elements like 'button'. With this method, you can name your event handler the way you want for each component.
- You can reduce the number of state variables. Managing state variable errors becomes easy.

## react router

### Setup the router

#### understanding

- Create a Browser Router in our root file. (index.js) It enables client side routing for web apps.
- The routing elements are provided by the react-router-dom. So you need to add react-router-dom as dependency first.

#### challenge

Add React Router to the App and create two routes. One for Home page and one for product detail page.

<https://codesandbox.io/s/react-router-install-7rr23w>

solution

<https://codesandbox.io/s/react-router-install-solution-8pjvmy>

## Adding links to navigate

understanding

- Browsers maintain their own history stack as the user navigates around. That's how the back and forward buttons can work. We can also achieve that with react router.

challenge

Add four routes for different pages, and then use links in the top to navigate.

<https://codesandbox.io/s/react-router-links-question-vcn927>

Test: You will see that browser Back and Forward button works.  
History is being preserved.

COPY

solution

<https://codesandbox.io/s/react-router-links-solution-j581js>

## url params to create unique links for each product

understanding

- If a path segment starts with `:` then it becomes a "dynamic segment".
- `products/:productId` - We call the `productId` in the URL "URL Params", or just "params" for short.

challenge

In a category listing page, on click of view details for each product, navigate to a page which shows the details of that particular product only. The URL should be unique for each product.

<https://codesandbox.io/s/react-router-url-params-challenge-y3jubb>

solution

<https://codesandbox.io/s/react-router-url-params-solution-q0mbqg>

## NavLink and active Class

understanding

- A `<NavLink>` is a special kind of `<Link>` that knows whether or not it is "active" or not.
- This is useful when building a navigation menu, such as a breadcrumb or a set of tabs where you'd like to show which of them is currently selected.

challenge

Make the nav link on the top stand out when they're active i.e when that current page is active.

[Home](#) | [Category](#) | [Cart](#) | [WishList](#)

# This is Home

# iPhones

iPhone 11 64 GB

Price: 51000 INR

[View Details](#)

iPhone XR 64 GB

Price: 41000 INR

[View Details](#)

iPhone 12 128 GB

solution

<https://codesandbox.io/s/react-router-navlink-solution-o9k0ew>

Path Matching exercise

understanding

`/invoices/:id` VS `/invoices/sent` which one will be selected?

understanding

<https://codesandbox.io/s/react-router-understanding-precendence-vz8b4>