

🚩 Current Skill Common Mistakes

Returning multiple nodes:

Since the component is only but a JavaScript function, and we know that JavaScript functions can only return a single object, value or item, therefore, the component can only return one JSX element.

Granted, it all seems a bit overcomplicated. Here is the right way to do it.

We wrap all the selected elements inside a single element div or section or any other wrapper of our choice. Take a look at the example below.

```
const MyComponent = () => {  
  // return multiple root node  
  // WRONG!  
  return (  
    <Comp1 />  
    <Comp2 />  
  )  
}  
  
// this code will trigger an error  
// Parse Error: Adjacent JSX elements must be wrapped in an enclosing tag
```

The correct way to do it:

```
return (  
  <div>  
    <Comp1 />  
    <Comp2 />  
  </div>  
);
```

React fragment

Naturally, the **div** or **section** or any other HTML container inherently have default CSS properties that could affect our design.

To sidestep these default properties, React gives us the **React.Fragment** which is a wrapper that has no effect on the CSS. A React fragment cannot be styled.

Take a look at its syntax in the provided example:

```
return (  
  <React.Fragment>  
    <div />  
    <div />  
  </React.Fragment>  
);  
//this also is valid  
return (  
  <>  
    <div />  
    <div />  
  </>  
);
```



Using absolute paths

Using an absolute path can be a little bit tricky and it can lead to a lot of problems when trying to reuse the component.

=> So, we advise you to use the relative paths instead.

```
//relative path  
import MyFirstComponent from './MyFirstComponent';  
  
//absolute path  
import MyFirstComponent from '/src/component/MyFirstComponent'
```



[< Previous](#)

[next >](#)



