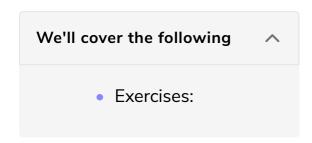
## Imperative React

Learn about the use of useRef hook for imperative programming in React!



In a React function component, React's useRef Hook is used mostly for imperative programming. Throughout React's history, the *ref* and its usage had a few versions:

- String Refs (deprecated)
- Callback Refs
- createRef Refs (exclusive for Class Components)
- useRef Hook Refs (exclusive for Function Components)

Recently, the React team introduced **React's createRef** as the latest equivalent to a function component's useRef hook:

```
class InputWithLabel extends React.Component {
  constructor(props) {
    super(props);
    this.inputRef = React.createRef();
}

componentDidMount() {
    if (this.props.isFocused) {
        this.inputRef.current.focus();
    }
}

render() {
    ...
    <input
        ref={this.inputRef}
        id={id}
        type={type}</pre>
```

```
value={value}
    onChange={onInputChange}
    />
    </>
    );
}
```

src/App.js

With the helper function, the ref is created in the class' constructor, applied in the JSX for the ref attributed, and here used in a lifecycle method. The ref can also be used elsewhere, like focusing the input field on a button click.

Where createRef is used in React's class components, React's useRef Hook is used in React function components. As React shifts towards function components, today its common practice to use the useRef hook exclusively to manage refs and apply imperative programming principles.

Let's see the live execution of legacy version:

```
°□ n□
                                           \squarePNG
                               5□
                                 @@
 IHDR
                D¤□Æ □APLTE
                          """""""""""""""""""""2RZVºÖ ÔôU·Ñ=r□$()'25]ÎíC□□0LS<o}X
 IHDR
           □·□ì □:PLTE
        @ 🗆 🗆
¢ßqÇ8Ù□´□mK˱mƶmÛü·yi!è□ΪYÏuë ÀÏ_Àï?i÷□ý+ò□□ÄA□|□ù{□□´?¿□_En□).□JËD¤<□
@-¢Z\Ts@R*□(□ ¯@□J□□□□u□X/□4J□9□¡5·DEμ4kÇ4□&i¥V4Ú□;®Đ□□¯□vsf:àg,□¢èBC»î$¶□ºÍùî□□á□@□ô□I
-ê>Û□º«¢XÕ¢î}ß"ëÛÑ;□ÃöN´□ØvÅý□Î,ÿ1 □ë×ÄO@&v/Äþ_□ö\ô□Ç\í.□□½+0□□;□□□!□fÊ□¦´Ó% JY·O□Â□'/Å]
```

Let's now observe the differences in legacy version and the modern version.

```
°□ n□
                                    \squarePNG
       ã□
                  )□
                        9□
                          5□
                            @@
              IHDR
          IHDR
          D¤□Æ □APLTE
                       @ 🗆 🗆
          □·□ì □:PLTE
¢ßqÇ8Ù□´□mK˱mƶmÛü·yi!è□ΪYÏuë ÀÏ_Àï?i÷□ý+ò□□ÄA□|□ù{□□´?¿□_En□).□JËD¤<□
       ¯©□J□□□□u□X/□4J□9□¡5·DEμ4kÇ4□&i¥V4Ú□;®Đ□□¯□vsf:àg,□¢èBC»î$¶□ºÍùî□□á□@□ô□I
-ê>Û□º«¢XÕ¢î}ߨëÛÑ;□ÃöN´□ØvÅý□Î,ÿ1 ☐ë×ÄO@&v/Äþ_□ö\ô□Ç\í.□□½+0□□;□□□!□fÊ□¦´Ó% JY·O□Â□'/Å]_□
```

## Exercises:

EXCICISCS!

• Read more about the different ref techniques in React.