


# React Practice

## App Component

```
export default function App() {  
  async function getAdvice() {  
    const response = await fetch("https://api.adviceslip.com/advice");  
    const data = await response.json();  
    console.log(data);  
  }  
  
  return (  
    <div>  
      <h1>Hello World!</h1>  
      <button onClick={getAdvice}>Get Advice!</button>  
    </div>  
  );  
}
```



---

```
import { useEffect, useState } from "react";
```

```
export default function App() {  
  /* useState is a function that returns an array in react  
  and here we are destructuring that array 1st position of the  
  array is value of the state, the 2nd value is a setter function that  
  we can use to update the piece of state */  
  const [advice, setAdvice] = useState("");  
  const [count, setCount] = useState(0);  
  
  async function getAdvice() {  
    const res = await fetch("https://api.adviceslip.com/advice");  
    const data = await res.json();  
    setAdvice(data.slip.advice);  
    /* Now in this getAdvice function we can use setAdvice function to  
    update the stat, whenever the piece of state is updated  
    user interface will also be updated */  
    // take current count add 1 and that will become the new count  
    setCount((c) => c + 1);  
  }  
  
  // Generate very first piece of advice when loaded  
  /* useEffect takes two arguments 1st a function that  
  we want to execute at the beginning when the component loads and 2nd  
  argument is dependency array */
```

```

useEffect(function () {
  getAdvice();
}, []);
return (
  <div>
    <h1>{advice}</h1>
    <button onClick={getAdvice}>Get Advice!</button>
    {/* include this component like it is another html element */}
    {/* Pass count as a prop to Message. Props are like parameters
    to function. We call the prop count and pass in the prop
    value */}
    <Message count={count} />
  </div>
);
}

```

```

// In react we divide user interfaces into components
// components are reusable pieces of code
// components are used to render UI
// Name of all components should start with capital letter(convention)
// Now accept the props object as parameter. In this props object count is now
a property
function Message(props) {
  // update count dynamically
  return (
    <p>
      You have read <strong>{props.count}</strong> pieces of advice
    </p>
  );
}

```