

⚡ Current Skill Creating Custom Hooks

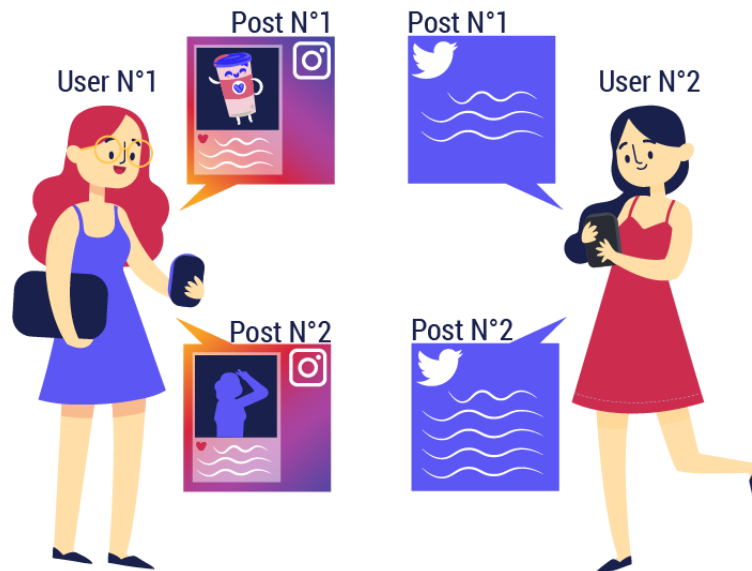
Custom Hooks.

A custom Hook is a JavaScript function whose name begins with "use". It provides flexible logic sharing.

We can call other Hooks in our custom Hooks.

Before we start, We need to be able to build our own Hooks in a personalized way and it will still ensure the functional reusability.

The following example of users and posts will help us better understand custom Hooks' practicality.



Custom hooks

Custom hooks are simply a wrapper function surrounding our existing hooks.

That's all there is to it! The only catch is: in order for React to recognize a function as a custom hook, its name should always start with use so that you can tell at a glance that the rules of Hooks apply to it.

Create a custom Hook

We will create a **useInterval** Hook and then we will use it.

```
function useInterval (timeout, getValue) {  
  const [value, setValue] = useState(getValue);
```

```
useEffect(() => {

  const intervalID = setInterval(

    () => setValue(getValue()),

    timeout

  );

  return function () {

    clearInterval(intervalId);

  }

}, []);

return value;

}

const get CurrentDate = () => new Date();

function App() {

  const date = useInterval(1000, getCurrentDate);

  return <p>Nous sommes le {date.toLocaleString("fr-FR")}</p>;

}
```

Recap

In this video you will find an overview about React Hooks

React Hooks



[< Previous](#)

[next >](#)



