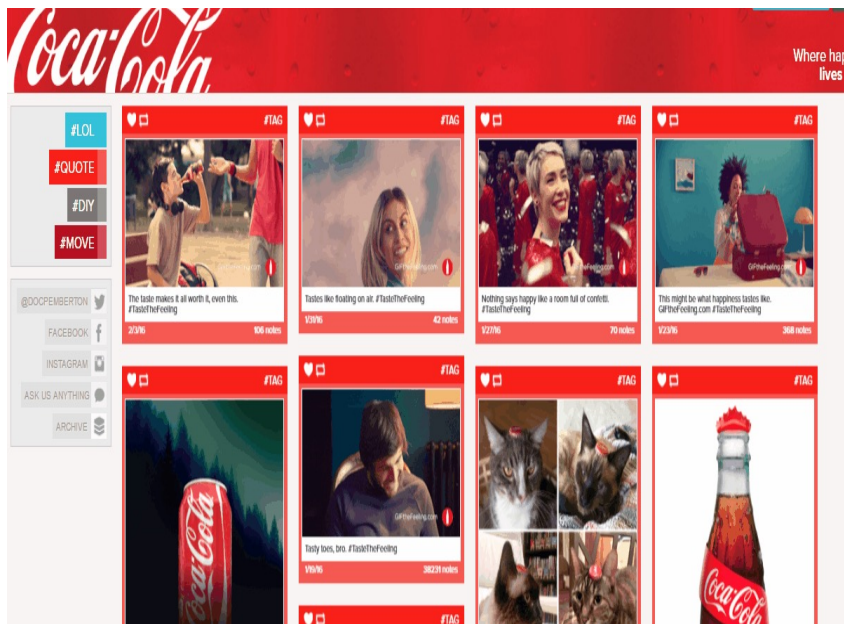


7

## Side effect with useEffect

# Side Effect



Xử lý HTTP request, response



Local Storage, Session Storage.



Xử lý timer (setTimeout, setInterval)



Subscribes / unsubscribes các dịch vụ bên ngoài.



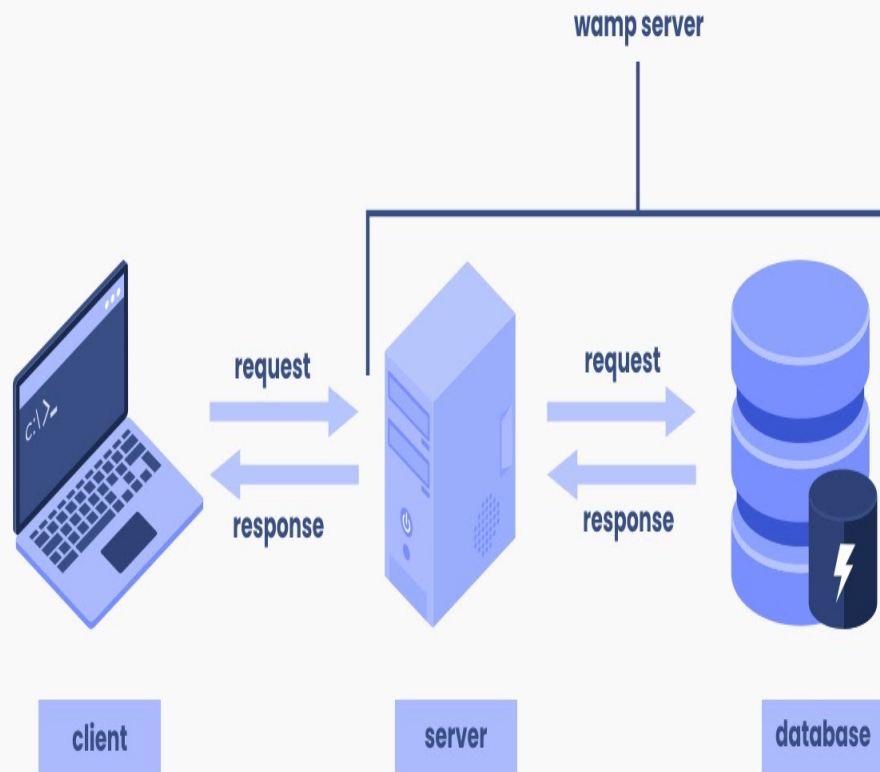
Tương tác trực tiếp với DOM

## Side Effect

Người dùng tiến hành đăng nhập.  
Khi click vào button “Login”, tiến hành gửi thông tin user lên server.

Nếu người dùng nhập chính xác,  
tiến hành điều hướng về trang chủ.

Nếu người dùng nhập không chính xác,  
tiến hành hiển thị lỗi yêu cầu người dùng nhập lại.



# Hooks

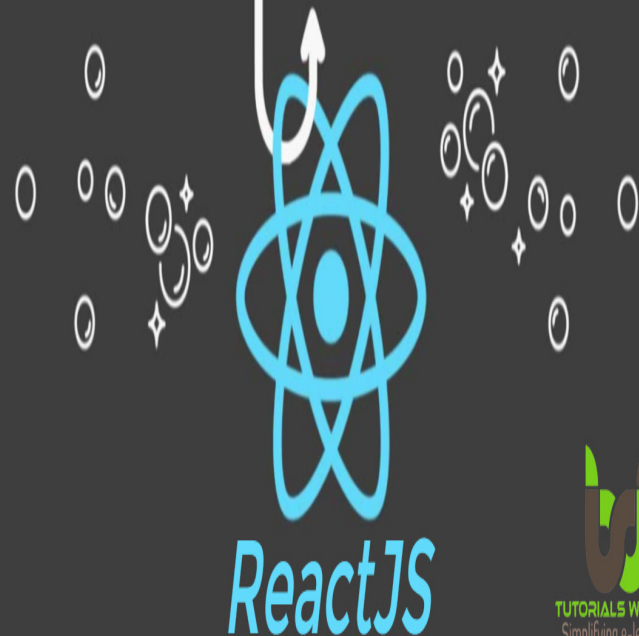
Hooks là các function làm việc với State và Lifecycle

Hooks thay thế cho ES6 Class

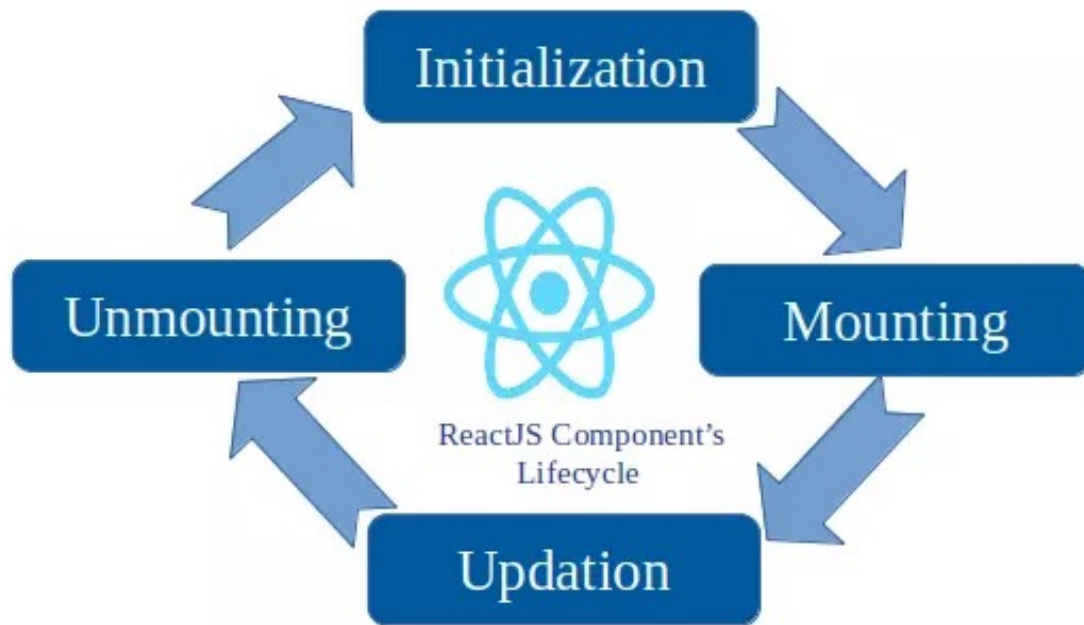


Vậy **Lifecycle** và **ES6 Class** là gì ?

useEffect Hooks



## Lifecycle và ES6 Class



## ES6 Class Simple

### Mount

**constructor**



**render**



**componentDidMount**

### Update

props  
changes



setState()



forceUpdate()



**render**



**componentDidUpdate**

### Unmount

**componentWillUnmount**

## Compare

ES6 Class



Hooks

### Class Components

state + setState(...)

componentDidMount()

componentWillUnmount()

componentDidMount()

### Functional Component

useState(...)

useEffect(...)

useEffect(...)

useEffect(...)

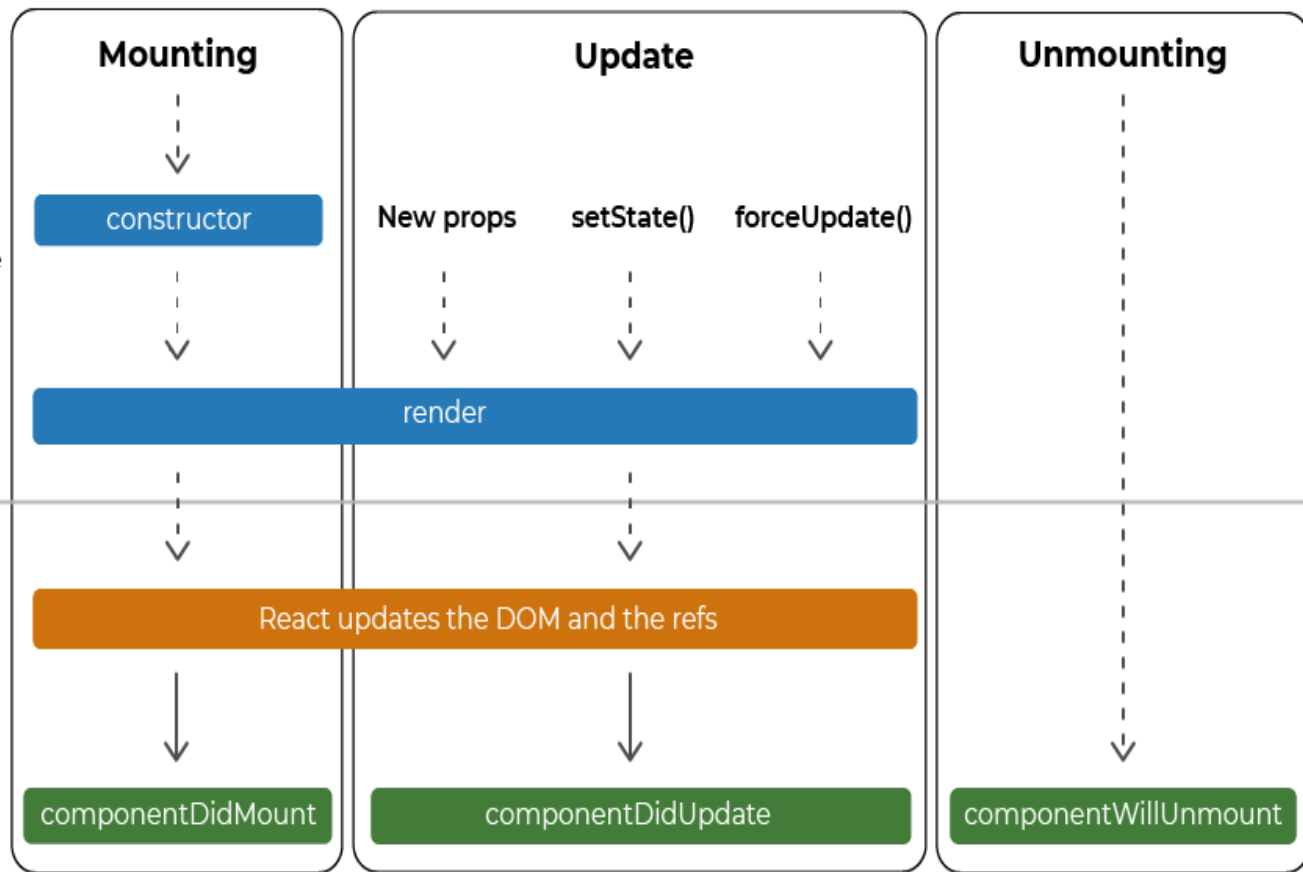
## Detail

### “Render phase”

Pure methods, without side effects. Can be interrupted, cancelled or restarted by React.

### “Commit phase”

Can operate on the DOM, generate side effects and program updates.



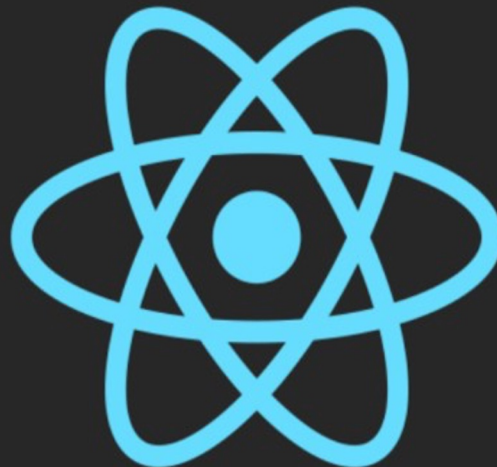


# Các Hook trong React

## REACT HOOKS

### GETTING STARTING WITH HOOKS

- useState
- useEffect
- useContext
- useCallback
- useReducer
- useMemo
- useRef
- useEffect



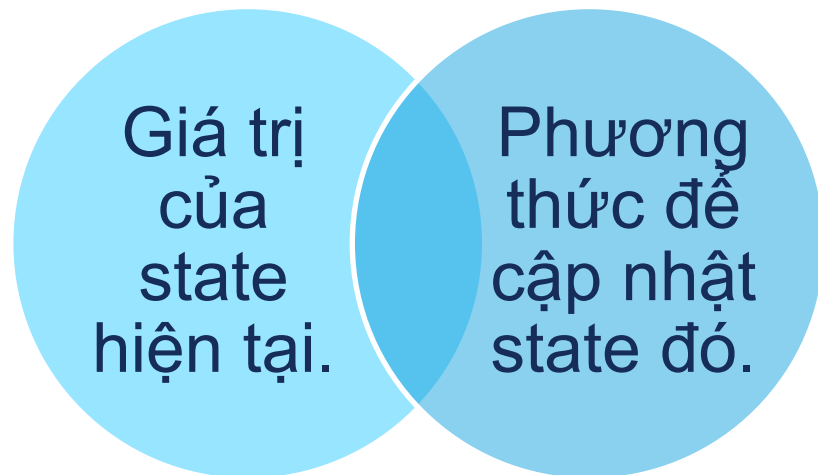
Sử dụng thế nào ?



## Khởi tạo State bằng Hook (useState)

- Chúng ta sử dụng hook có tên là `useState()` để khai báo và cập nhật giá trị State.  
`useState()` trả về 1 cặp, gồm:

```
const [count, setCount] = useState(0);  
  
const handleChangeCount = () => {  
  setCount(count + 1);  
}
```



## Side Effect cơ bản với useEffect

- Để xử lý các logic bên ngoài component, React cung cấp một function hook đặc biệt là `useEffect`. Cú pháp cơ bản của `useEffect` như sau:

Cú pháp:

```
useEffect(() => { /* effect here */ })
```

# When Does useEffect Run?

## Running

5

Trường hợp  
sử dụng  
useEffect

### • once

• similar to componentDidMount

```
useEffect(() => {  
  // put 'run once' code here  
}, [])
```

• ————— empty array = run once

### • on props change

• similar to componentDidUpdate

```
function YourComponent({ someProp }) {  
  useEffect(() => {  
    // code to run when someProp changes  
  }, [someProp]);  
}
```

• ————— include all monitored props

### • after every render

• similar to componentDidUpdate

```
useEffect(() => {  
  // put 'every update' code here  
})
```

• ————— no second argument = run every render

### • on state change

• similar to componentDidUpdate

```
function YourComponent() {  
  const [state, setState] = useState()  
  useEffect(() => {  
    // code to run when state changes  
  }, [state])  
}
```

• ————— include all state vars to watch

### • on unmount

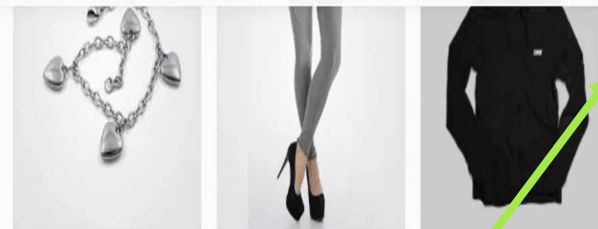
• similar to componentWillUnmount

```
useEffect(() => {  
  return () => {  
    // put unmount code here  
  }  
})
```

• ————— return the cleanup function

## Ứng dụng

- ✓ Khởi tạo Component: Initial Data.
- ✓ State thay đổi: VD - Validate form
- ✓ State thay đổi: Bộ lọc danh sách/ bảng
- ✓ State thay đổi: Trigger animation
- ✓ Props thay đổi : Update lại Component
- ✓ Props thay đổi : Update theo phụ thuộc



Anchor Bracelet

Accessories

\$150.00 - \$180.00

Black Brown Red

☆☆☆☆☆

SELECT OPTIONS

Basic Gray Jeans

Women

\$150.00

☆☆☆☆☆

ADD TO CART

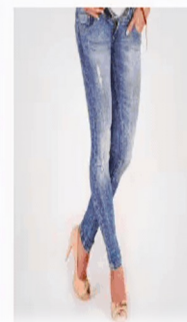
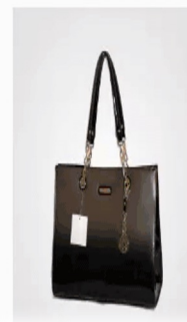
Black Hoodie

Men

\$150.00

☆☆☆☆☆

ADD TO CART



Sale!

## Clean-up Function

Clean up subscriptions

Clean up modals

Remove event listeners

Clear timeouts

Cú pháp:

```
useEffect(() => {  
  // Your effect  
  
  return () => {  
    // Cleanup  
  };  
}, []);
```

## Return

```
useEffect( () => {  
  
  // perform a side effect  
  
  return function () { /* clean up side effect */ };  
}, [ ] );
```

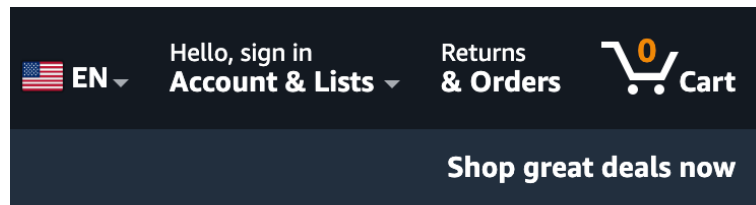
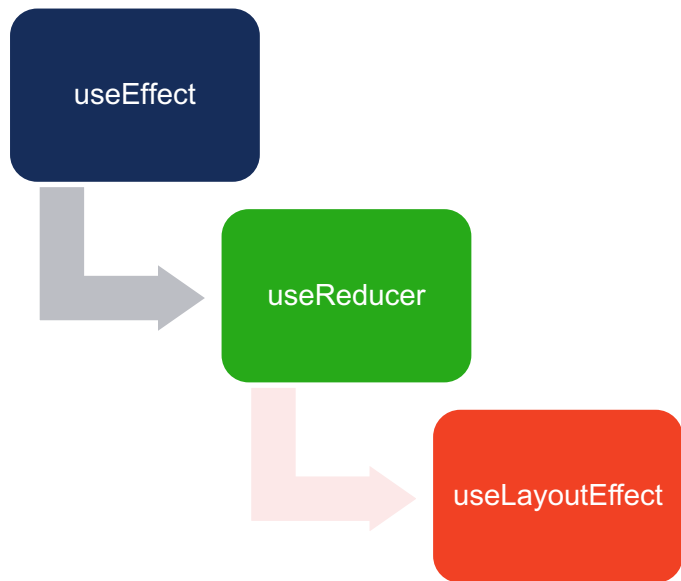


### **clean-up function:**

Return a function to clean up after the effect. E.g. unsubscribe, stop timers, remove listeners, etc.

# Thực hành

- Sử dụng các Hook cơ bản sau để hoàn thành tính năng **mua hàng** trên 1 trang web



Sort by: Featured ▼

rice Organizations (SAGE Sourcebooks for

cs) | by Ralph Brody and Murali D Nair | Nov 21,