

K-Digital Training 웹 개발자 입문 2기

React

WITH 팀 리쳐드



state와 useState

state란?

- React에서 앱의 유동적인 데이터를 다루기 위한 개체
- 왜 사용할까?
 - State가 변경될 시 자동으로 재렌더링 되기 때문
 - 이 점이 변수와 다른 점

클래스형 컴포넌트 state

- 기존 형태

```
import React, { Component } from "react";

class ClassState extends Component {
  constructor(props) {
    super(props);

    this.state = {
      number: 0,
    };
  }
  render() {
    const { number } = this.state;
    return (
      <div> <h3>Number : {number}</h3> </div>
    );
  }
}

export default ClassState;
```

클래스형 컴포넌트 state

- 현재 버전

```
import React, { Component } from "react";

class ClassState extends Component {
  state = {
    number: 0,
  };
  render() {
    const { number } = this.state;
    return (
      <div> <h3>Number : {number}</h3> </div>
    );
  }
}

export default ClassState;
```

this.setState()

- state의 값을 바꾸는 함수

```
<button onClick={() => { this.setState({number: 1}); }}> 1로 설정 </button>  
<button onClick={() => { this.setState({number: 2}); }}> 2로 설정 </button>  
iv>
```

- 바로 동작하는 것이 아닌, 저장해놓고 나중에 실행되는 것이기 때문에 setState를 2번 쓴다 해서 2번 동작하는 것은 아니다.

this.setState()

```
<div>
  <div> <h3>Number : {number}</h3> </div>
  <button onClick={() => {
    this.setState({number: number + 1 });
    this.setState({number: this.state.number + 1});
  }}> 설정 </button>
</div>
```

Number : 1

설정

this.setState()

- setState의 인자로 함수에 넣어줄 때는 아래와 같이 사용한다.

```
<div>
  <div> <h3>Number : {number}</h3> </div>
  <button onClick={() => {
    this.setState(prevState => {
      return {
        number: prevState.number + 1
      };
    });

    this.setState(prevState => ({
      number: prevState.number + 1
    }));
  }}> 설정 </button>
</div>
```

Number : 2

설정

this.setState()

```
<div>
  <div> <h3>Number : {number}</h3> </div>
  <button onClick={() => {
    this.setState(prevState => {
      return {
        number: prevState.number + 1
      };
    });

    this.setState(prevState => ({
      number: prevState.number + 1
    }));
  }}> 설정 </button>
</div>
```

- prevState는 기존 상태
- setState의 인자로 사용된 함수는 이전 state 값을 전달받으며 순서대로 실행된다.

함수형 컴포넌트 useState

- 함수형 컴포넌트는 state 기능이 원래 없었다.
- React 16.8 버전 이후부터 useState라는 함수가 생겼고, 이를 통해 함수형 컴포넌트에서도 state를 사용하게 되었다.

함수형 컴포넌트 useState

```
import React, { useState } from 'react';

const SayFunction = () => {
  const [ message, setMessage ] = useState("");
  const onClickEnter = () => { setMessage("안녕하세요~"); };
  const onClickLeave = () => { setMessage("안녕히가세요."); };

  return (
    <div>
      <button onClick={onClickEnter}>입장</button>
      <button onClick={onClickLeave}>퇴장</button>
      <h1>{message}</h1>
    </div>
  );
};

export default SayFunction;
```

함수형 컴포넌트 useState

```
import React, { useState } from 'react';

const SayFunction = () => {
  const [ message, setMessage ] = useState("");
  const onClickEnter = () => { setMessage("안녕하세요~"); };
  const onClickLeave = () => { setMessage("안녕히가세요."); };

  return (
    <div>
      <button onClick={onClickEnter}>입장</button>
      <button onClick={onClickLeave}>퇴장</button>
      <h1>{message}</h1>
    </div>
  );
};

export default SayFunction;
```

- useState 함수의 인자에는 상태의 초기값이 들어간다.
- useState의 초기값은 숫자일 수도, 문자일 수도, 배열일 수도 있다. 즉, 값의 형태가 자유로움
- 배열의 첫번째 원소는 현재 상태, 두번째 원소는 상태를 바꿔주는 setter 함수

함수형 컴포넌트 useState


```
import React, { useState } from 'react';
function Counter() {
  const [ number, setNumber ] = useState(0);
  const onClickEnter = () => setNumber(number+1);

  return (
    <div>
      <h1>{number}</h1>
      <button onClick={onClickEnter}>+1</button>
    </div>
  );
}

export default Counter;
```

실습. React state 실습

- <https://deeplearningon.com/student/index/k-digital-2>
- 접속 후 “React > state와 useState 실습” 진행

state와 useState	
state와 useState 실습	
state와 useState 실습 해답	