

## React JS를 사용하기 위한 준비

```
<script src="https://unpkg.com/react@17/umd/react.production.min.js"></script>  
<script src="https://unpkg.com/react-dom@17/umd/react-dom.production.min.js"></script>  
<script src="https://unpkg.com/babel-standalone@6.15.0/babel.min.js"></script>
```

### 01 JSX : script안에서 작성(html비슷하지만 html은 아님, 단독태그 끝에 /넣어야함)

```
<div>  
  
  <p className= "s1"></p>  
  
  <br />  
  
</div>
```

### 02 ReactDOM.render() : JSX요소나 컴포넌트를 마운트요소에 출력

```
ReactDOM.render(  
  JSX요소 또는 <컴포넌트 />, 마운트요소  
);
```

### 03 컴포넌트

```
class 컴포넌트명(첫글자대문자) extends React.Component {  
  
  render(){ return( <div>JSX작성</div> ) }  
  
}
```

## 04 스테이트/함수/생명주기

```
class 컴포넌트명(첫글자대문자) extends React.Component {  
  
    constructor(props){  
  
        super(props);  
  
        this.state = {   스테이트명:0   };  //스테이트는 변수는 아닌데 변수비슷한거  
  
        this.함수 = this.함수.bind(this);  
  
    }  
  
    함수명(){  
  
        this.setState({  
  
            스테이트: this.state.스테이트 + 1  
  
        });  
  
    }  
  
    componentDidMount(){  
  
        alert("컴포넌트가 마운트된후 실행");  
  
    }  
  
    componentDidUpdate(){  
  
        alert("컴포넌트가 업데이트된후 실행");  
  
    }  
  
    render(){ return(<div>JSX작성</div>) }  
  
}
```

## 05 속성(props)

```
class 컴포넌트명(첫글자대문자) extends React.Component {
```

```
  render(){ return(
```

```
    <div>
```

```
      <p>{ this.props.속성 }</p>
```

```
    </div>
```

```
  ) }
```

```
}
```

```
ReactDOM.render(
```

```
  <컴포넌트 속성="안녕" />, 마운트요소
```

```
);
```

## 06 이벤트

```
<JSX태그 on이벤트={ this.함수명 }>
```

```
<JSX태그 on이벤트={ function(){할일} }>
```

```
<JSX태그 on이벤트={ ()=>{할일} }>
```

**onClick** // 클릭했을때

**onMouseOver** // 오버했을때

**onMouseOut** // 아웃했을때