

컴포넌트

## 컴포넌트 (Component)



리액트로 만들어진 앱을 이루는 **최소 단위**

## 컴포넌트

### 기존의 HTML 문서

div

h1

아메리카노

p

3500

div

h1

아메리카노

p

3500

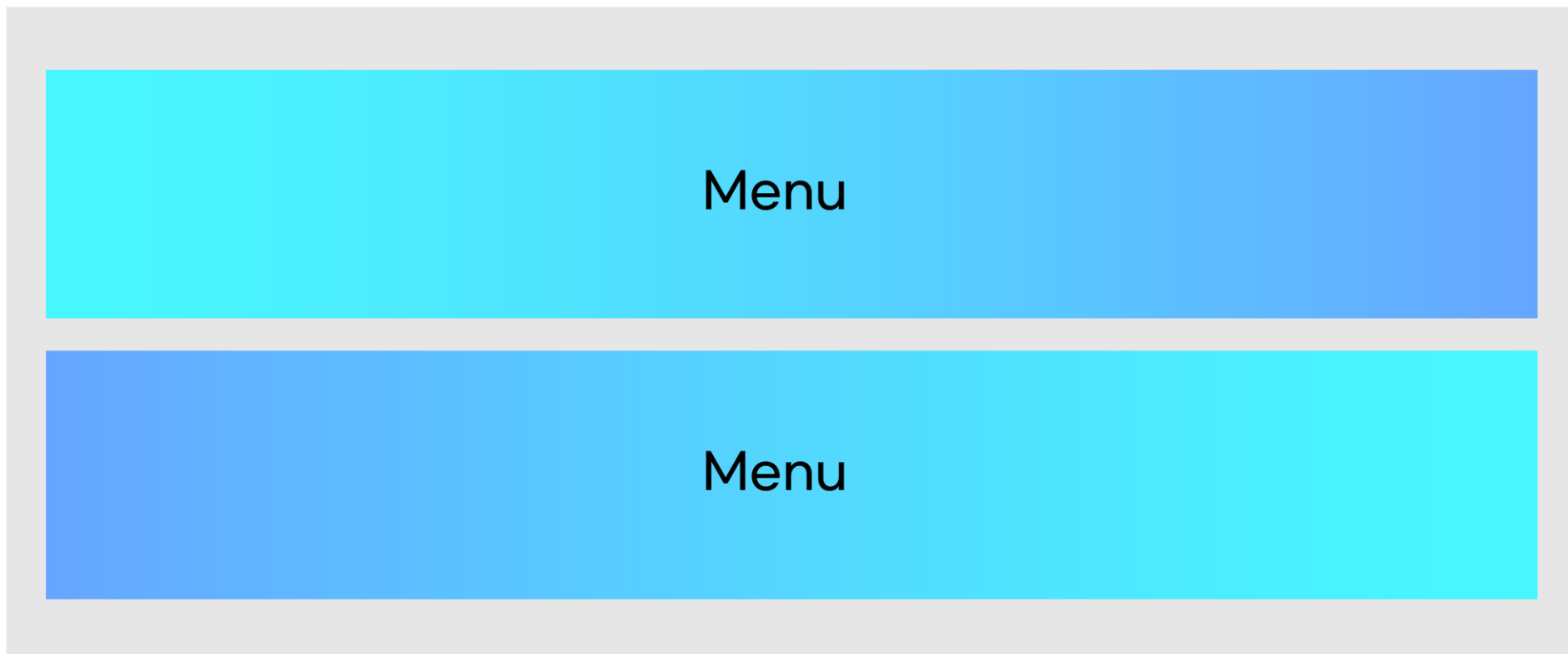
## 컴포넌트

### 기존의 HTML 문서



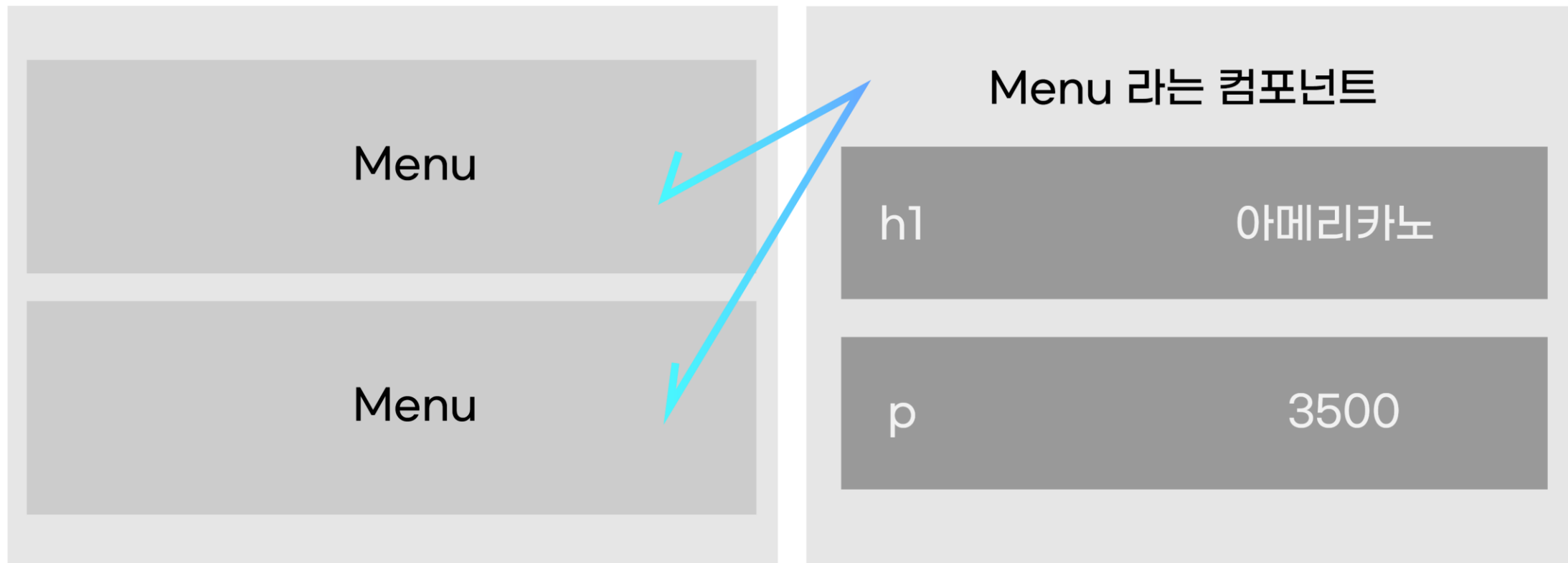
## 컴포넌트

### 기존의 HTML 문서



## 컴포넌트

### 기존의 HTML 문서



## 컴포넌트

### 1. Extension 설치



#### ES7+ React/Redux/React-Native snippets v4.4.3

dsznajder | 5,705,180 | ★★★★★☆ (59)

Extensions for React, React-Native and Redux in JS/TS with ES7+ syntax. Customizable. Built-

Disable



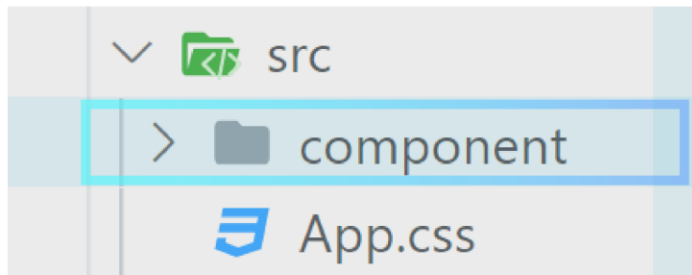
Uninstall



This extension is enabled globally.

## 컴포넌트

### 2. component 폴더 생성 & 파일 생성



component → 파일이름.js 생성 (대문자)

이 때, 주의할 점! 컴포넌트의 이름은 반드시 **대문자**로 시작할 것!

## 컴포넌트

### 3. 컴포넌트 파일 관리



```
import React from 'react'
```

```
✓ const Menubox = () => {  
✓   return (  
    <div>Menubox</div>  
  )  
}
```

```
export default Menubox
```

→ 컴포넌트 만드는 단축키 : rafce

→ 함수형 컴포넌트 생성 완료!

→ 컴포넌트 내보내기



## 컴포넌트

### 4. 내보낸 컴포넌트 불러오기

```
import './App.css';  
import MenuBox from './component/MenuBox'  
  
function App() {  
  return (  
    <>  
    <MenuBox></MenuBox>  
    </>  
  );  
}  
  
export default App;
```

App.js

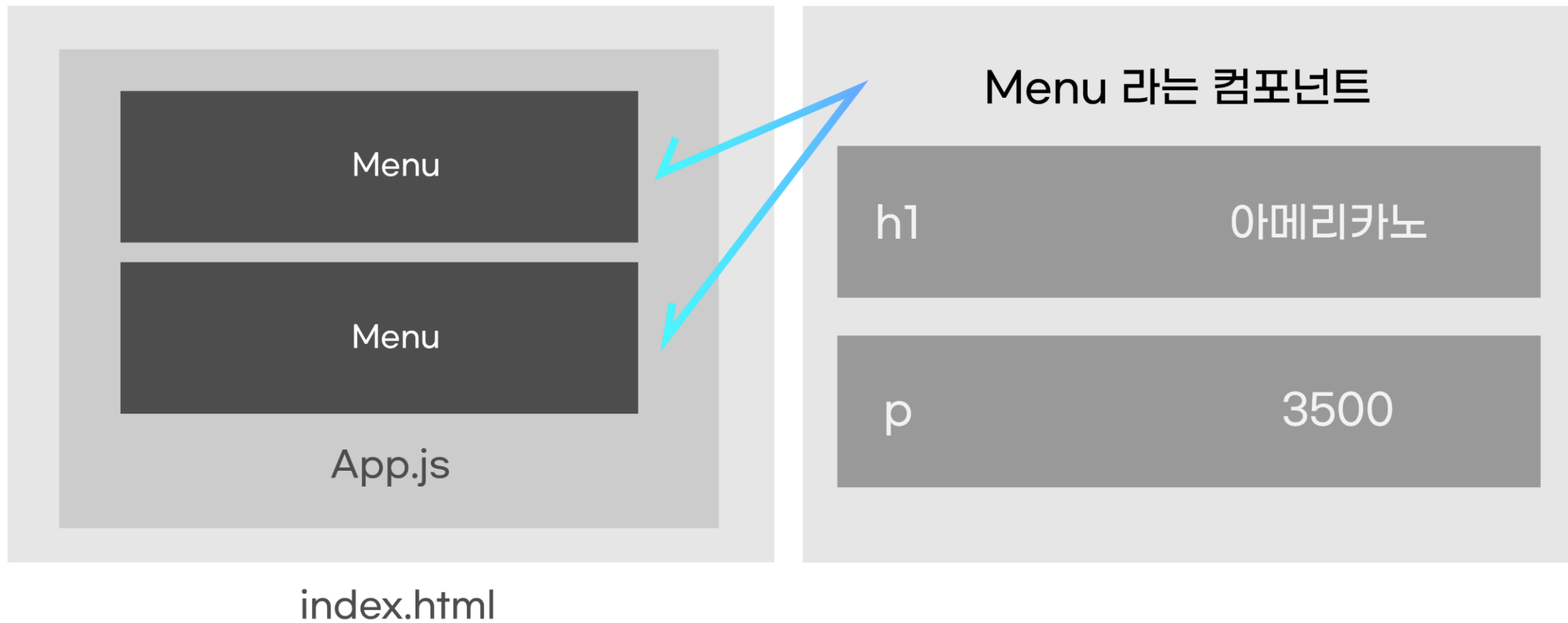


컴포넌트 불러오기



사용하기

## 컴포넌트



컴포넌트

내가 원하는걸 묶어서  
태그화 시킨다!

Menu 라는 컴포넌트

Menu

Menu

App.js

index.html

아메리카노

p

3500

## 컴포넌트 사용 시 주의사항

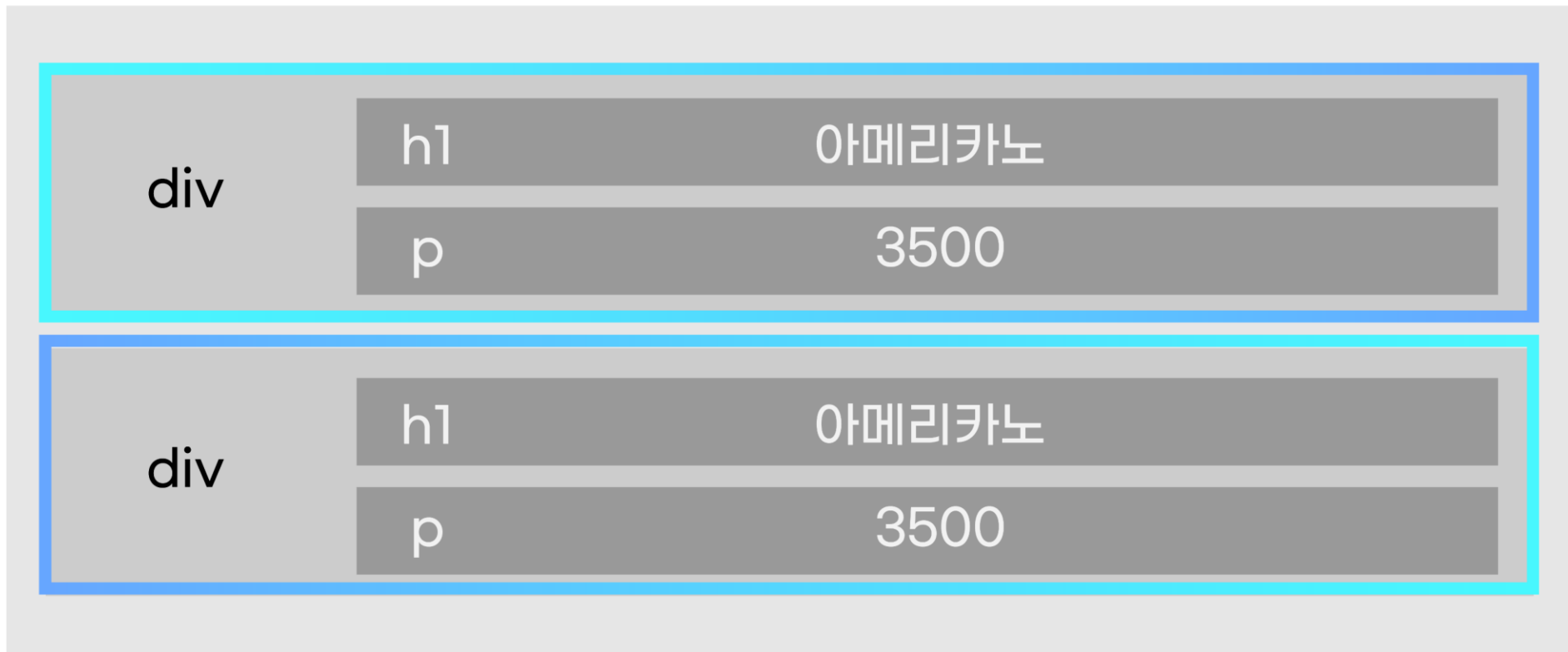


### 컴포넌트의 이름은 반드시 **대문자로** 시작해야한다

리액트가 컴포넌트와 일반 HTML 태그를 구분할 수 있어야한다.  
그렇기 때문에 리액트에서는 일반 HTML 태그는 소문자로,  
컴포넌트는 반드시 대문자로 시작해야한다!

props

## 기존의 HTML 문서



props

## 기존의 HTML 문서

div	h1	아메리카노
	p	3500
div	h1	카페라떼
	p	4000

props

## props (프로퍼티)



상위 컴포넌트가 하위 컴포넌트에 **값을 전달**할 때 사용한다  
상위 컴포넌트에서 값을 입력하고, 하위 컴포넌트에서는 값을 읽어온다

## props 상위 컴포넌트

### props (프로퍼티) 사용방법

```
function App() {  
  return (  
    <>  
    <MenuBox menu="카페라떼"></MenuBox>  
    <MenuBox></MenuBox>  
  </>  
);  
}
```

```
function App() {  
  let coffee = "카페라떼"  
  return (  
    <>  
    <MenuBox menu={coffee}></MenuBox>  
    <MenuBox></MenuBox>  
  </>  
);  
}
```

문자열을 전달할 때는 **큰따옴표** (“”)   
문자열 외의 값을 전달할 때는 **중괄호** ({})를 사용한다



props 하위 컴포넌트

## props (프로퍼티) 사용방법

```
const Menubox = (props) => {  
  return (  
    <div className="menuBox">  
      <p>{props.menu}</p>  
      <span>3500</span>  
    </div>  
  )  
}
```

## 컴포넌트 & props 예제

교육운영부

선영표

전략기획팀

강예진

홍보팀

임보미

기획팀

최영화

컴포넌트와 props를 이용해서  
팀원들을 소개하는 페이지를 만들어보자!

---

state

# State

컴포넌트 내부에서 관리되는 변경이 가능한 데이터

---

state

# State

컴포넌트 내부에서 관리되는 **변경이 가능한 데이터**

변수와의 차이점은?

state

## State와 변수의 차이점

state는 일반 변수와 다르게  
값이 변하면 화면이 렌더링된다

이 때, **setState()** 라는 함수를 이용한다

---

state

```
const [state, setState] = useState(초기값)
```

---

state

```
const [state, setState] = useState(초기값)
```



변수 이름

state

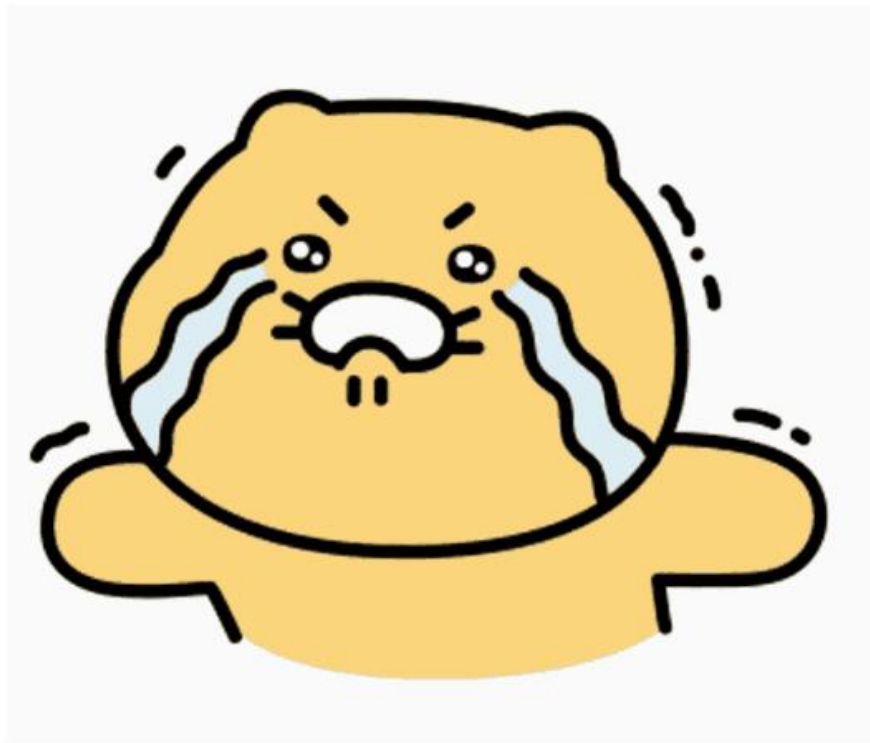
state를 변경시켜주는 함수

const [state, setState] = useState(초기값)

변수 이름



state



❤️ 좋아요 1개

state

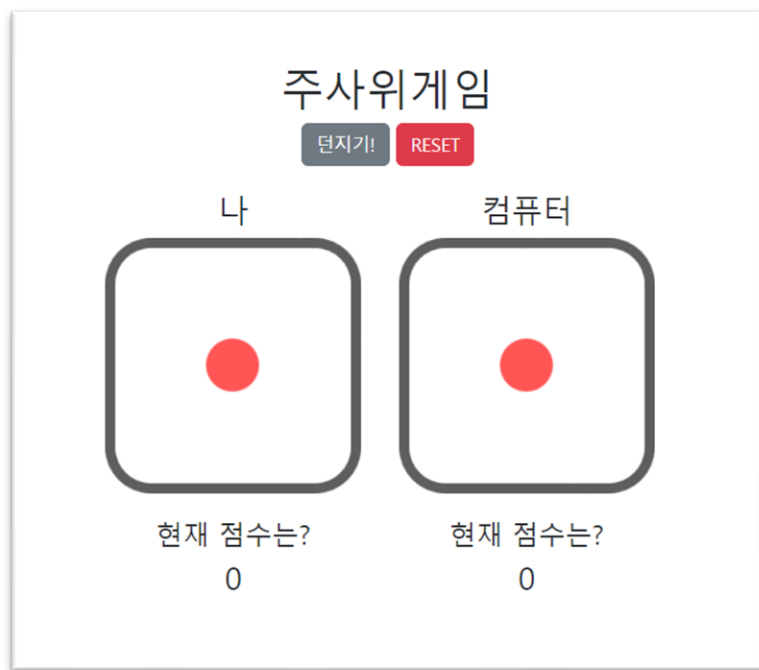


내가 입력한 숫자 : 1

랜덤한 숫자 : 1

정답입니다!

## state



localhost:3000 내용:

게임이 종료되었습니다. 당신의 승리입니다.

확인

localhost:3000 내용:

게임이 종료되었습니다. 당신의 패배입니다.

확인