

# PJT1. 카운터 앱 만들기

☰ 다중 선택

Project

```
// src/App.jsx

import "./App.css";
import Viewer from "./components/Viewer";
import Controller from "./components/Controller";
import { useState } from "react";

function App() {
  const [count, setCount] = useState(0);

  const onClickButton = (value) => {
    setCount(count + value);
  };

  return (
    <div className="App">
      <h1>Simple Counter</h1>
      <section>
        <Viewer count={count} />
      </section>
      <section>
        <Controller onClickButton={onClickButton} />
      </section>
    </div>
  );
}

export default App;
```

```
// src/App.css

body {
  padding: 20px;
```

```

}

.App {
  margin: 0 auto;
  width: 400px;
}

.App > section {
  background-color: rgb(245, 245, 245);
  border: 1px solid rgb(240, 240, 240);
  border-radius: 5px;
  padding: 20px;
  margin-bottom: 10px;
}

```

```

// src/components/Viewer.jsx

import React from "react";

const Viewer = ({ count }) => {
  return (
    <div>
      <div>현재 카운트 : </div>
      <h1>{count}</h1>
    </div>
  );
};

export default Viewer;

```

```

// src/components/Controller.jsx

import React from "react";

const Controller = ({ onClickButton }) => {
  return (
    <div>

```

```

<button
  onClick={() => {
    onClickButton(-1);
  }}
>
  -1
</button>
<button
  onClick={() => {
    onClickButton(-10);
  }}
>
  -10
</button>
<button
  onClick={() => {
    onClickButton(-100);
  }}
>
  -100
</button>
<button
  onClick={() => {
    onClickButton(100);
  }}
>
  +100
</button>
<button
  onClick={() => {
    onClickButton(10);
  }}
>
  +10
</button>
<button
  onClick={() => {
    onClickButton(1);
  }}

```

```

    }}
  >
    +1
  </button>
</div>
);
};

export default Controller;

```

## Simple Counter

현재 카운트 :

**0**

-1 -10 -100 +100 +10 +1

## 주의사항

1. react의 화면을 구성할 때 여러개의 컴포넌트들이 부모-자식 관계를 이룸(계층 구조 형성)
2. 특정 컴포넌트가 다른 컴포넌트에게 데이터를 전달하려면 부모-자식 관계여야 함
3. 하나의 상태를 여러 컴포넌트에서 관리하게 될 경우 이 상태는 반드시 여러 컴포넌트의 공통 부모가 되는 곳에 만들어야 함(스테이트 리프팅, 스테이트 끌어올리기)