

5. JSX를 사용하는 리액트 (실습)

Prof. Seunghyun Park (sp@hansung.ac.kr)

Division of Computer Engineering

학습 목표: 5. JSX를 사용하는 리액트

- 실습1: 리액트 엘리먼트를 생성하여 렌더링

- 1) 순수 리액트
- 2) JSX 사용

- 실습2~3: 데이터 (배열, 객체) 활용

- 1) 배열, 객체를 리액트 엘리먼트로 생성하여 렌더링
- 2) 배열에 map(), 객체에 Object.key() 적용
- 3) 배열, 객체를 리액트 컴포넌트로 생성하여 렌더링
- 4) 배열에 map(), 객체에 Object.key() 적용

- 실습4: 데이터 확장1

- 1~2) 객체를 리액트 엘리먼트로 생성하여 렌더링
- 3) 객체를 리액트 컴포넌트로 생성하여 렌더링
- 4~5) 객체를 리액트 컴포넌트에 매개변수 포함
- 6) 리액트 컴포넌트 분리

- 실습5: 데이터 확장

- 실습6: create-react-app으로 프로젝트 생성

- 1) 기본 프로젝트
- 2) famous-location 소스 적용

실습준비: 빈 페이지 생성

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Page setup</title>
</head>
<body>

<!-- Root element -->
<div id="root"></div>

<!-- React Library & React DOM-->
<script src="https://unpkg.com/react@16/umd/react.development.js"></script>
<script src="https://unpkg.com/react-dom@16/umd/react-dom.development.js"></script>
<script src="https://unpkg.com/babel-standalone@6.15.0/babel.min.js"></script>

<script type="text/babel">

  /* ch05-04-babel.html from ch04-01-01-page-setup.html */
  // JavaScript 파일의 링크를 script 태그에 포함
  // JSX 코드

</script>

</body>
</html>
```

실습1-1: 리액트 엘리먼트를 생성하여 렌더링

- 문제: <div> 엘리먼트 하위에 자식 노드로 <h1> 엘리먼트 생성

```
<!-- Target Container -->
<div id="root"></div>

/* ex06-01-1.html */
// create an element with "h1" and getUser() function
const user = {
  firstname: "Seunghyun",
  lastname: "Park"
}
const getUser = user => `${user.lastname}, ${user.firstname}`;

const element = React.createElement(
  "h1",
  null,
  "Hello, ", getUser(user), "!"
);

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(element);
```

```
React.createElement(
  type, [props], [...children]
)
```

- 인자로 주어지는 타입에 따라
새로운 리액트 엘리먼트를 생성하여 반환

```
ReactDOM.render(
  element, container[, ...callback]
)
```

- 인자로 주어지는 렌더링 할 리액트 엘리먼트를
제공된 컨테이너의 DOM에 렌더링,
- 구성요소에 대한 참조를 반환

```
root = ReactDOM.createRoot(root)
root.render(element)
```

실습1-2: JSX를 사용한 리액트 엘리먼트를 생성하여 렌더링

- 문제: <div> 엘리먼트 하위에 자식 노드로 <h1> 엘리먼트 생성

```
<!-- Target Container -->
<div id="root"></div>

/* ex06-01-2.html */
// create an element with "h1" and getUser() function using JSX
const user = {
  firstname: "Seunghyun",
  lastname: "Park"
}
const getUser = user => `${user.lastname}, ${user.firstname}`;

const element1 = (
  <h1>
    Hello, {getUser(user)}!
  </h1>
);

const element2 = <h1>Hello, {getUser(user)}!</h1>;

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(element1);
```

```
<script type="text/babel">

const element1 = (
  <h1>
    Hello, {getUser(user)}!
  </h1>
);
```

Babel

```
<script type="module">

const element1 = React.createElement(
  "h1",
  null,
  "Hello, ",
  getUser(user),
  "!"
);
```

실습2-1: 데이터 (배열) 활용

- 문제: 데이터 (배열)을 사용한 리액트 엘리먼트 구성 및 렌더링

```
<!-- Target Container -->
<div id="root"></div>
```

```
/* ex06-02-1.html */
const data = [...];
```

```
const majors = (
  <ul>
    <li>{data[0]}</li>
    <li>{data[1]}</li>
    <li>{data[2]}</li>
    <li>{data[3]}</li>
  </ul>
);
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(majors);
```

```
const data = [
  "web",
  "mobile software",
  "bigdata",
  "ditital contents"
]
```

- web
- mobile software
- bigdata
- ditital contents

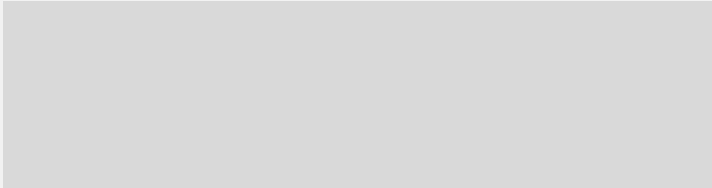
```
<div id="root">
  <ul == $0>
    <li>
      ::marker
      "web"
    </li>
    <li>...</li>
    <li>...</li>
    <li>...</li>
  </ul>
</div>
```

실습2-2: 데이터 (배열) 활용 - map() 함수

- 문제: 데이터 (배열)을 사용한 리액트 엘리먼트 구성 및 렌더링, map() 함수 활용

```
<!-- Target Container -->
<div id="root"></div>
```

```
/* ex06-02-2.html */
const data = [...];
```

```
const majors = (
  <ul>
    
  </ul>
);
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(majors);
```

```
const data = [
  "web",
  "mobile software",
  "bigdata",
  "ditital contents"
]
```

- web
- mobile software
- bigdata
- ditital contents

```
<div id="root">
  <ul == $0>
    <li>
      ::marker
      "web"
    </li>
    <li>...</li>
    <li>...</li>
    <li>...</li>
  </ul>
</div>
```

실습2-3: 데이터 (배열) 활용 - 리액트 컴포넌트

- 문제: 데이터 (배열)을 사용한 리액트 컴포넌트 구성 및 렌더링, map() 함수 활용

```
<!-- Target Container -->
<div id="root"></div>
```

```
/* ex06-02-3.html */
const data = [...];
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render( );
```

```
const data = [
  "web",
  "mobile software",
  "bigdata",
  "ditital contents"
]
```

- web
- mobile software
- bigdata
- ditital contents

```
<div id="root">
  <ul> == $0
    <li>
      ::marker
      "web"
    </li>
    <li>...</li>
    <li>...</li>
    <li>...</li>
  </ul>
</div>
```


실습2-4: 데이터 (배열) 활용 - 리액트 컴포넌트, 매개변수 포함

- 문제: 데이터 (배열)을 사용한 리액트 컴포넌트 구성 및 렌더링, map() 함수 활용, 매개변수 전달

```
<!-- Target Container -->
<div id="root"></div>
```

```
/* ex06-02-4.html */
const data = [...];
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(<MajorList majors={data} />);
```

```
const data = [
  "web",
  "mobile software",
  "bigdata",
  "ditital contents"
]
```

- web
- mobile software
- bigdata
- ditital contents

```
<div id="root">
  <ul> == $0
    <li>
      ::marker
      "web"
    </li>
    <li>...</li>
    <li>...</li>
    <li>...</li>
  </ul>
</div>
```

실습3-1: 데이터 (객체) 활용

- 문제: 데이터 (객체)을 사용한 리액트 엘리먼트 구성 및 렌더링

```
<!-- Target Container -->
```

```
<div id="root"></div>
```

```
/* ex06-03-1.html */
```

```
const data = [...];
```

```
const data = {  
  country: "Korea",  
  city: "Seoul",  
  address: "116 Samseongyo-ro"  
}
```

```
const location = (  
  <ul>  
    <li>country: {data.country}</li>  
    <li>city: {data.city}</li>  
    <li>address: {data.address}</li>  
  </ul>  
)
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(location);
```

- country: Korea
- city: Seoul
- address: 116 Samseongyo-ro

```
..▼ <body> == $0  
  <!-- Root element -->  
  ▼ <div id="root">  
    ▼ <ul>  
      ▼ <li>  
        ::marker  
        "country"  
        ": "  
        "Korea"  
      </li>  
      ▶ <li>...</li>  
      ▶ <li>...</li>  
    </ul>  
  </div>
```

실습3-2: 데이터 (객체) 활용 – Object.key() 함수와 map() 함수

- 문제: 데이터 (객체)을 사용한 리액트 엘리먼트 구성 및 렌더링, Object.key()와 map() 함수

```
<!-- Target Container -->
```

```
<div id="root"></div>
```

```
/* ex06-03-2.html */
```

```
const data = [...];
```

```
const location = (
```

```
<ul>
```

Object.key(obj): 객체의 속성 값을 배열로 반환

```
</ul>
```

```
);
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
```

```
root.render(location);
```

```
const data = {  
  country: "Korea",  
  city: "Seoul",  
  address: "116 Samseongyo-ro"  
}
```

- country: Korea
- city: Seoul
- address: 116 Samseongyo-ro

```
.. <body> == $0
```

```
<!-- Root element -->
```

```
<div id="root">
```

```
<ul>
```

```
<li>
```

```
  ::marker
```

```
  "country"
```

```
  ":"
```

```
  "Korea"
```

```
</li>
```

```
> <li>...</li>
```

```
> <li>...</li>
```

```
</ul>
```

```
</div>
```

실습3-3: 데이터 (객체) 활용 - 리액트 컴포넌트

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트 구성 및 렌더링, `Object.key()`와 `map()` 함수

```
<!-- Target Container -->
```

```
<div id="root"></div>
```

```
/* ex06-03-1.html */
```

```
const data = [...];
```

```
const data = {  
  country: "Korea",  
  city: "Seoul",  
  address: "116 Samseongyo-ro"  
}
```

```
<ul>
```

```
  {Object.keys(data).map((obj_prp, i) => (  
    <li key={i}>{obj_prp}: {data[obj_prp]}</li>  
  ))}
```

```
</ul>
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(  
  ;
```

- country: Korea
- city: Seoul
- address: 116 Samseongyo-ro

```
.. ▾ <body> == $0
```

```
<!-- Root element -->
```

```
▾ <div id="root">
```

```
  ▾ <ul>
```

```
    ▾ <li>
```

```
      ::marker
```

```
      "country"
```

```
      ": "
```

```
      "Korea"
```

```
    </li>
```

```
    ▶ <li>...</li>
```

```
    ▶ <li>...</li>
```

```
  </ul>
```

```
</div>
```

실습3-4: 데이터 (객체) 활용 - 리액트 컴포넌트, 매개변수 포함

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트 구성 및 렌더링, `Object.key()`와 `map()` 함수, 매개변수 포함

```
<!-- Target Container -->
```

```
<div id="root"></div>
```

```
/* ex06-03-1.html */
```

```
const data = [...];
```

```
const data = {  
  country: "Korea",  
  city: "Seoul",  
  address: "116 Samseongyo-ro"  
}
```

```
<ul>
```

```
  {Object.keys(location).map((obj_prp, i) => (  
    <li key={i}>{obj_prp}: {location[obj_prp]}</li>  
  ))}
```

```
</ul>
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(<Location location={data} />);
```

- country: Korea
- city: Seoul
- address: 116 Samseongyo-ro

```
.. ▾ <body> == $0
```

```
<!-- Root element -->
```

```
▾ <div id="root">
```

```
  ▾ <ul>
```

```
    ▾ <li>
```

```
      ::marker
```

```
      "country"
```

```
      ": "
```

```
      "Korea"
```

```
    </li>
```

```
    ▶ <li>...</li>
```

```
    ▶ <li>...</li>
```

```
  </ul>
```

```
</div>
```

실습4-1: 데이터 확장 (1) – 리액트 엘리먼트

- 문제: 데이터 (객체)을 사용한 리액트 엘리먼트 구성 및 렌더링

```
/* ex06-04-1.html */
```

```
const place = (  
  <section id={data.name.toLowerCase().replace(/ /g, "-")}>  
    <h2>{data.name}</h2>  
    <h3>Location</h3>  
    <ul className="locations">  
      <li>{data.locations.country}</li>  
      <li>{data.locations.city}</li>  
      <li>{data.locations.address}</li>  
    </ul>  
    <h3>Majors</h3>  
    <ul className="majors">  
      <li>{data.majors[0]}</li>  
      <li>{data.majors[1]}</li>  
      <li>{data.majors[2]}</li>  
      <li>{data.majors[3]}</li>  
    </ul>  
  </section>  
)  
const root = ReactDOM.createRoot(document.getElementById('root'))  
root.render(place);
```

```
const data = {  
  name: "Hansung University",  
  locations: {  
    country: "Korea",  
    city: "Seoul",  
    address: "116 Samseongyo-ro"  
  },  
  majors: [  
    "Web",  
    "Mobile SW",  
    "Bigdata",  
    "Digital contents"  
  ]  
};
```

Hansung University

Location

- Korea
- Seoul
- 116 Samseongyo-ro

Majors

- Web
- Mobile SW
- Bigdata
- Digital contents

```
<!-- Root element -->  
<div id="root">  
  <section id="hansung-university">  
    <h2>Hansung University</h2>  
    <h3>Location</h3>  
    <ul class="locations">  
      <li>  
        ::marker  
        "Korea"  
      </li>  
      <li>...</li>  
      <li>...</li>  
    </ul>  
    <h3>Majors</h3>  
    <ul class="majors">...</ul>  
  </section>  
</div>
```

실습4-2: 데이터 확장 (1) – 리액트 엘리먼트

- 문제: 데이터 (객체)을 사용한 리액트 엘리먼트 구성 및 렌더링, `Object.key()`와 `map()`

```
/* ex06-04-2.html */
```

```
const place = (  
  <section id={data.name.toLowerCase().replace(/ /g, "-")}>  
    <h2> </h2>  
    <h3>Location</h3>  
    <ul className="locations">  
        
    </ul>  
    <h3>Majors</h3>  
    <ul className="majors">  
        
    </ul>  
  </section>  
)  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(place);
```

```
const data = {  
  name: "Hansung University",  
  locations: {  
    country: "Korea",  
    city: "Seoul",  
    address: "116 Samseongyo-ro"  
  },  
  majors: [  
    "Web",  
    "Mobile SW",  
    "Bigdata",  
    "Digital contents"  
  ]  
};
```

Hansung University

Location

- Korea
- Seoul
- 116 Samseongyo-ro

Majors

- Web
- Mobile SW
- Bigdata
- Digital contents

```
<div id="root">  
  <section id="hansung-university"> ==  
    <h2>Hansung University</h2>  
    <h3>Location</h3>  
    <ul class="locations">  
      <li>  
        ::marker  
        "Korea"  
      </li>  
      <li>...</li>  
      <li>...</li>  
    </ul>  
    <h3>Majors</h3>  
    <ul class="majors">...</ul>  
  </section>  
</div>
```

실습4-3: 데이터 확장 (1) - 리액트 컴포넌트

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트 구성 및 렌더링, Object.key()와 map()

```
/* ex06-04-3.html */
```

```
<section id={data.name.toLowerCase().replace(/ /g, "-")}>
  <h2>{data.name}</h2>
  <h3>Location</h3>
  <ul className="locations">
    {Object.keys(data.locations).map((obj_prp, i) => (
      <li key={i}>{obj_prp}: {data.locations[obj_prp]}</li> ))}
  </ul>
  <h3>Majors</h3>
  <ul className="majors">
    {data.majors.map((major, i) => ( <li key={i}>{major}</li> ))}
  </ul>
</section>
```

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render( );
```

```
const data = {
  name: "Hansung University",
  locations: {
    country: "Korea",
    city: "Seoul",
    address: "116 Samseongyo-ro"
  },
  majors: [
    "Web",
    "Mobile SW",
    "Bigdata",
    "Digital contents"
  ]
};
```

Hansung University

Location

- Korea
- Seoul
- 116 Samseongyo-ro

Majors

- Web
- Mobile SW
- Bigdata
- Digital contents

```
<div id="root">
  <section id="hansung-university"> ==
    <h2>Hansung University</h2>
    <h3>Location</h3>
    <ul class="locations">
      <li>
        ::marker
        "Korea"
      </li>
      <li>...</li>
      <li>...</li>
    </ul>
    <h3>Majors</h3>
    <ul class="majors">...</ul>
  </section>
</div>
```


실습4-4: 데이터 확장 (1) – 리액트 컴포넌트, 매개변수 포함

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트 구성 및 렌더링, Object.key()와 map() 함수, 매개변수 포함

```
/* ex06-04-4.html */
const Place = ({data}) => (
  <section id={data.name.toLowerCase().replace(/ /g, "-")}>
    <h2>{data.name}</h2>
    <h3>Location</h3>
    <ul className="locations">
      {Object.keys(data.locations).map((obj_prp, i) => (
        <li key={i}>{obj_prp}: {data.locations[obj_prp]}</li> ))}
    </ul>
    <h3>Majors</h3>
    <ul className="majors">
      {data.majors.map((major, i) => ( <li key={i}>{major}</li> ))}
    </ul>
  </section>
);

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(<Place famous={data} />);
```

```
const data = {
  name: "Hansung University",
  locations: {
    country: "Korea",
    city: "Seoul",
    address: "116 Samseongyo-ro"
  },
  majors: [
    "Web",
    "Mobile SW",
    "Bigdata",
    "Digital contents"
  ]
};
```

Hansung University

Location

- Korea
- Seoul
- 116 Samseongyo-ro

Majors

- Web
- Mobile SW
- Bigdata

```
<div id="root">
  <section id="hansung-university">
    <h2>Hansung University</h2>
    <h3>Location</h3>
    <ul class="locations">
      <li>
        ::marker
        "Korea"
      </li>
      <li>...</li>
      <li>...</li>
    </ul>
    <h3>Majors</h3>
    <ul class="majors">
      <li>Web</li>
      <li>Mobile SW</li>
      <li>Bigdata</li>
      <li>Digital contents</li>
    </ul>
  </section>
</div>
```

실습4-5: 데이터 확장 (1) – 리액트 컴포넌트, 매개변수 포함

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트 구성 및 렌더링, Object.key()와 map() 함수, 매개변수 포함

```
/* ex06-04-5.html */
const Place = ({ name, locations, majors }) => (
  <section id={name.toLowerCase().replace(/ /g, "-")}>
    <h2>{name}</h2>
    <h3>Location</h3>
    <ul className="locations">
      {Object.keys(locations).map((obj_prp, i) => (
        <li key={i}>{obj_prp}: {locations[obj_prp]}</li> ))}
    </ul>
    <h3>Majors</h3>
    <ul className="majors">
      {majors.map((major, i) => ( <li key={i}>{major}</li> ))}
    </ul>
  </section>
);

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(<Place name={data.name} locations={data.locations}
  majors={data.majors} />);
```

```
const data = {
  name: "Hansung University",
  locations: {
    country: "Korea",
    city: "Seoul",
    address: "116 Samseongyo-ro"
  },
  majors: [
    "Web",
    "Mobile SW",
    "Bigdata",
    "Digital contents"
  ]
};
```

Hansung University

Location

- Korea
- Seoul
- 116 Samseongyo-ro

Majors

- Web
- Mobile SW
- Bigdata

```
<div id="root">
  <section id="hansung-university">
    <h2>Hansung University</h2>
    <h3>Location</h3>
    <ul class="locations">
      <li>
        ::marker
        "Korea"
      </li>
      <li>...</li>
      <li>...</li>
    </ul>
    <h3>Majors</h3>
    <ul class="majors">
      <li>Web</li>
      <li>Mobile SW</li>
      <li>Bigdata</li>
      <li>Digital contents</li>
    </ul>
  </section>
</div>
```

실습4-6: 데이터 확장 (1) – 리액트 컴포넌트 분리

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트를 분리

```
/* ex06-04-6.html */  
  
<div>  
  <header>  
    <h1>{props.title}</h1>  
  </header>  
  <article className="places">  
    <Place  
      name={props.famous.name}  
      locations={props.famous.locations}  
      majors={props.famous.majors}  
    />  
  </article>  
</div>  
);  
  
...  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(<Famous famous={data} title="Famous places" />);
```

```
<section id={name.toLowerCase().replace(/ /g, "-")}>  
  <h2>{name}</h2>  
  <Location locations={locations} />  
  <Majors majors={majors} />  
</section>  
);  
  
<>  
  <h3>Location</h3>  
  <ul className="locations">  
    {Object.keys(locations).map((obj_prp, i) => (  
      <li key={i}>{obj_prp}: {locations[obj_prp]}</li> ))}  
  </ul>  
</>  
);  
  
<>  
  <h3>Majors</h3>  
  <ul className="majors">  
    {majors.map((major, i) => (  
      <li key={i}>{major}</li> ))}  
  </ul>  
</>  
);
```

실습5-1: 데이터 확장 (2) – 리액트 컴포넌트 분리

- 문제: 데이터 (객체)을 사용한 리액트 컴포넌트를 분리

```
/* ex06-05-1.html */
```

```
const Famous = => (
```

```
<div>
```

```
<header>
```

```
<h1>{props.title}</h1>
```

```
</header>
```

```
<article className="places">
```

```
<Place
```

```
key={
```

```
name={
```

```
locations={
```

```
majors={
```

```
/> )}}
```

```
</article>
```

```
</div>
```

```
);
```

```
const root = ReactDOM.createRoot(
```

```
document.getElementById('root')
```

```
);
```

```
root.render(<Famous famous={data} title="Famous places" />);
```

Famous places

Hansung University

Location

- country: Korea
- city: Seoul
- address: 116 Samseonggyo-ro

Majors

- Web
- Mobile SW
- Bigdata
- Digital contents

Musée du Louvre

Location

- country: France
- city: Paris
- address: 99 Rue de Rivoli

Majors

- Mona Lisa
- Le Sacre de Napoléon
- La Liberté guidant le peuple

Griffith Observatory

Location

- country: US
- city: Los Angeles
- address: 2800 East Observatory Rd

Majors

- La La Land
- sunset
- James Dean

```
const data = [
  {
    name: "Hansung University",
    locations: {
      country: "Korea",
      city: "Seoul",
      address: "116 Samseonggyo-ro"
    },
    majors: [ ... ]
  },
  {
    name: "Musée du Louvre",
    locations: {
      country: "France",
      ...
    },
    majors: [ ... ]
  },
  {
    name: "Griffith Observatory",
    locations: {
      country: "US",
      ...
    },
    majors: [ ... ]
  }
];
```

실습6-1: create-react-app 으로 프로젝트 생성

```
> create-react-app famous-location-03  
> cd famous-location-03  
> npm start
```

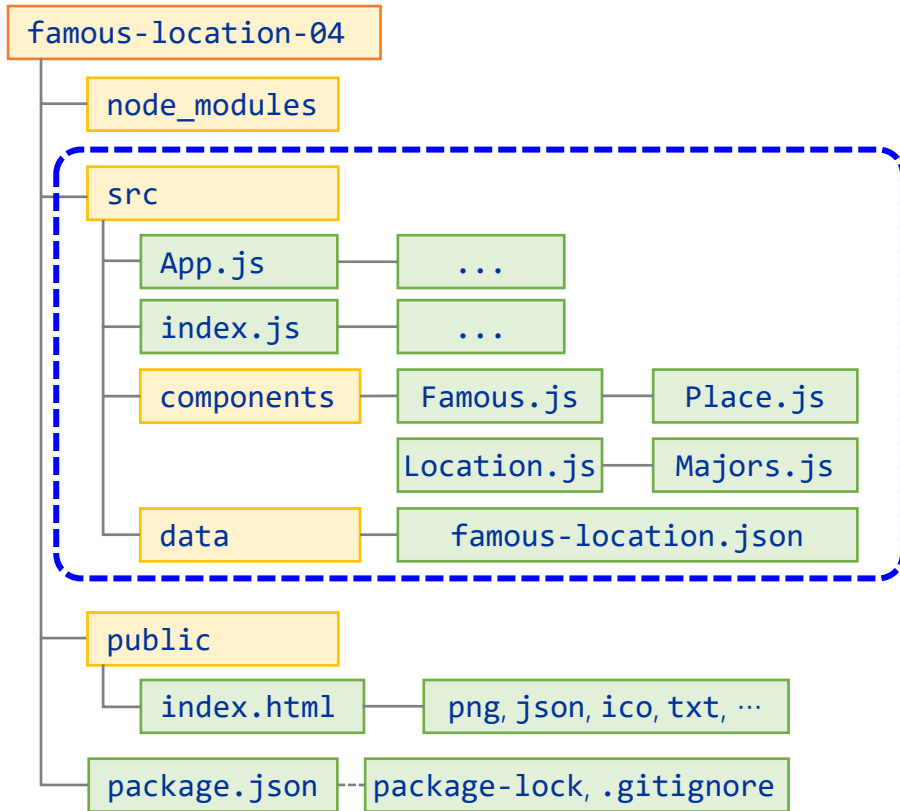


Edit src/App.js and save to reload.

[Learn React](#)

실습6-2: create-react-app 으로 프로젝트 생성하여 famous 앱 빌드

```
> create-react-app famous-location-04
```



```
/* ex06/proj/famous-location-04/src/index.js */
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
```

```
const root = ReactDOM.createRoot(
  document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

```
/* ex06/proj/famous-location-04/src/App.js */
import data from "../data/famous-location.json";
import Famous from "../components/Famous";

function App() {
  return (
    <Famous famous={data} title="Famous places" />
  );
}

export default App;
```

실습6-2: create-react-app 으로 프로젝트 생성하여 famous 앱 빌드

```
/* famous-location-04/src/components/Famous.js */
import Place from "./Place"

const Famous = props => (
  <div>
    ...
    <Place ... />
  </div>
);

export default Famous;
```

```
/* famous-location-04/src/components/Location.js */
const Location = ({ locations }) => (
  <div>...</div>
);

export default Location;
```

```
/* famous-location-04/src/components/Place.js */
import Location from "./Location";
import Majors from "./Majors";

const Place = ({ name, locations, majors }) => (
  <section>
    ...
    <Location ... />
    <Majors ... />
  </section>
);

export default Place;
```

```
/* famous-location-04/src/components/Majors.js */
const Majors = ({ majors }) => (
  <>...</>
);

export default Majors;
```