4. 리액트의 작동 원리 (실습)

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

Division of Computer Engineering

한성대학교 HANSUNG UNIVERSITY

학습 목표: 4. 리액트의 작동 원리

- React element 생성과 ReactDOM 렌더링
  - 엘리먼트 생성: React.createElement()
    - 단일 엘리먼트
    - 엘리먼트 트리
    - map() 함수 활용
  - ReactDOM 렌더링
    - ReactDOM.render() -- react@16, react@18
    - 함수형 컴포넌트 -- react@16, react@18
      - 전역 데이터 활용
      - 매개변수 전달
    - 클래스형 컴포넌트 -- react@16, react@18

실습준비: 빈 페이지 생성

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Page setup: ex05-00 from ch04-01-01-page-setup.html</title>
</head>
<body>

<!-- 타겟 컨테이너 -->
<div id="react-container"></div>

<!-- React@18
    <script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
    <script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
-->
<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>

/* ex05-00.html, 순수 리액트와 자바스크립트 코드 */

</script>

</body>
</html>
```

실습1: 리액트 엘리먼트 생성하여 렌더링 (react@16)

- 문제: 엘리먼트 하위에 자식 노드로 <h1> 엘리먼트 생성

https://ko.reactjs.org/docs/react-api.html#createelement
https://ko.reactjs.org/docs/react-dom.html#render

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

/* ex05-01-1.html */

// create addr element with "h1"
```

```
ReactDOM.render(
    addr,
    document.getElementById('root')
);

console.log('addr', addr);
```

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

- 인자로 주어지는 타입에 따라 새로운 **리액트 엘리먼트를 생성하여 반환**

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

- 인자로 주어지는 렌더링 할 **리액트 엘리먼트**를
  제공된 컨테이너의 **DOM** (렌더링이 일어날 대상 DOM)**에 렌더링**,
- 구성요소에 대한 참조를 반환

실습2: 리액트 엘리먼트 (react@18)

- 문제: 엘리먼트 하위에 자식 노드로 <h1> 엘리먼트 생성

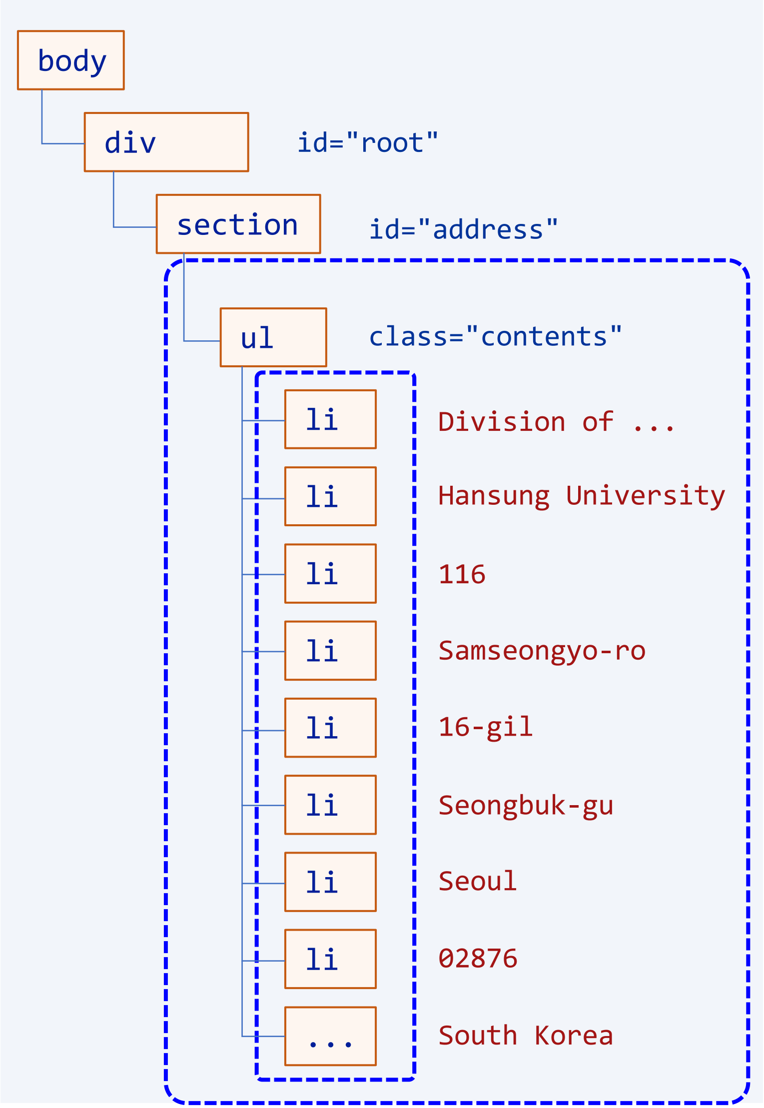
```
<!-- Target Container -->
<div id="root"></div>

<script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
<script>

/* ex05-01-2.html */
const addr = React.createElement("h1", {id: "address"}, "Address");

// create root & render root at react@18


console.log('root:', root);
console.log('addr:', addr);

</script>
```

https://ko.reactjs.org/docs/react-dom-client.html

```
ReactDOM.createRoot(container[, options]);
```

```
const root = createRoot(container);
root.render(element)
```

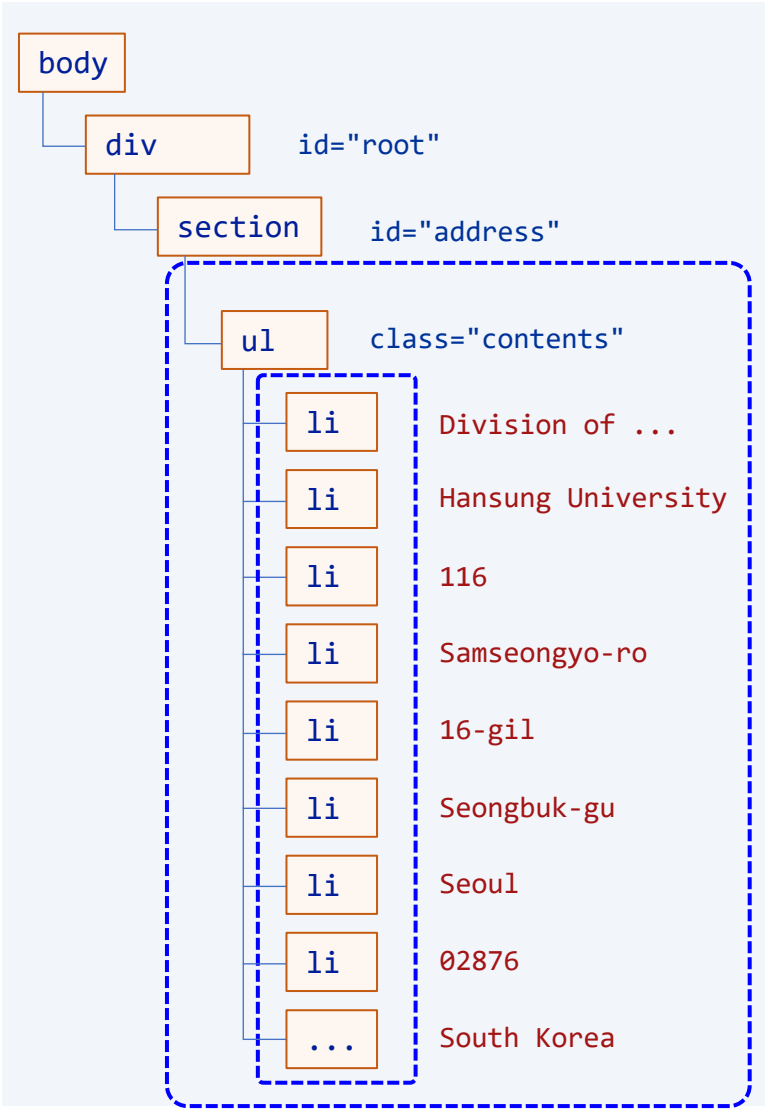
실습3: 리액트 엘리먼트 – 엘리먼트 계층




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

/* ex05-01-3.html */
const addr = React.createElement(

);

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

```
[
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

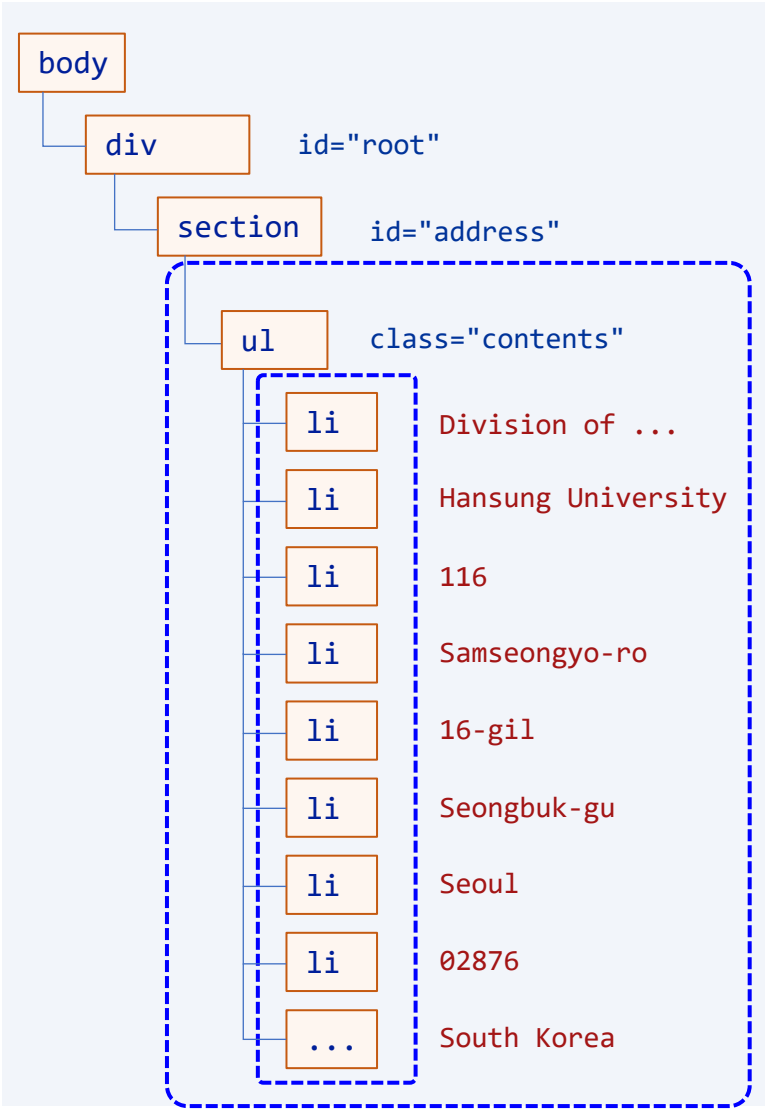
실습4: 리액트 엘리먼트 – 배열 활용




```
/* ex05-01-4.html */
const address = [ ... ];

const addr = React.createElement(
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
                                                    
);

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

```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

실습5~6: 리액트 엘리먼트 – map() 활용




```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

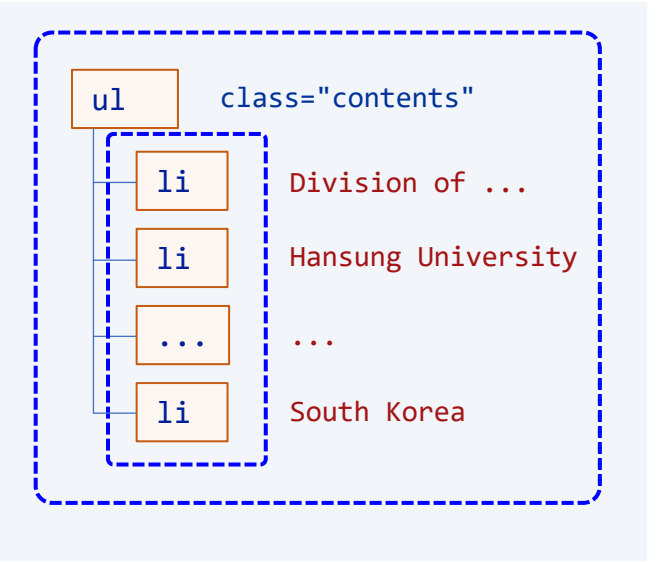
```
/* ex05-01-5.html */
const address = [ ... ];

const addr = React.createElement(
  "section",
  {id: "address"},
  React.createElement(
    "ul",
    {className: "contents"},
    
  )
);

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

```
/* ex05-01-6.html */
...
const addr = React.createElement(
    ...
    
  )
);
...
```

실습7~8: 함수형 컴포넌트 (react@16, react@18)

```
/* ex05-02-1.html */
const Address = function(){

}

ReactDOM.render(
    React.createElement(Address, null, null),
    document.getElementById("root")
);
```

```
/* ex05-02-2.html */
...
const Address = function(){

}

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

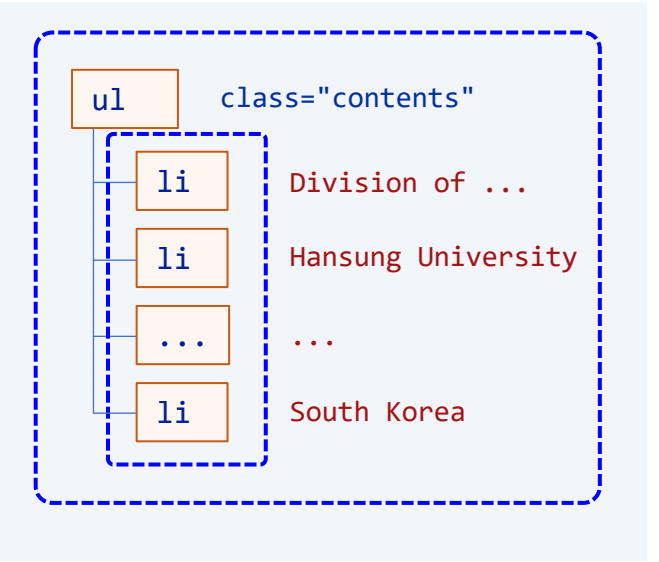



```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

실습9~10: 함수형 컴포넌트에서 매개변수 전달 (react@16, react@18)

```
/* ex05-02-3.html */

  return React.createElement(
    "ul",
    { className: "address" },
    props.data.map( (addr, i) => React.createElement("li", {key: i}, addr) )
  );
}

ReactDOM.render(

    document.getElementById("root")
);
```

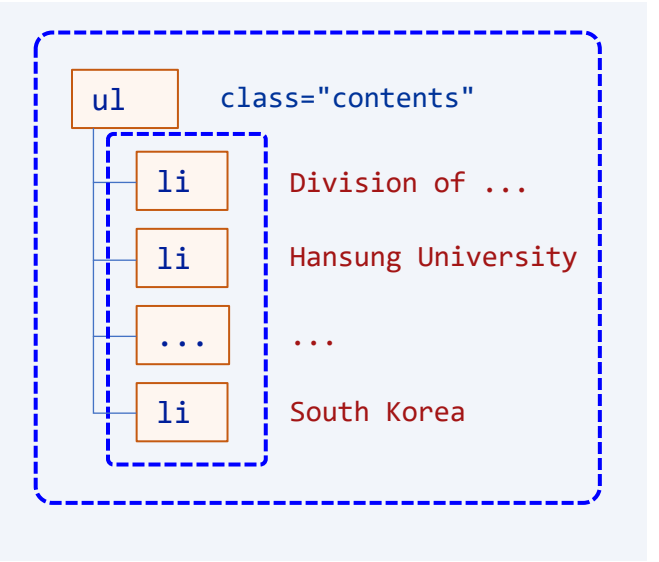
```
/* ex05-02-4.html */
...

  return React.createElement(...);
}

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




```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

실습11~12: 함수형 컴포넌트에서 매개변수 전달 (react@16, react@18)

```
/* ex05-02-5.html */
const Address =
  return React.createElement(
    "ul",
    { className: "address" },
  );
}

ReactDOM.render(
    ,
    document.getElementById("root")
);
```

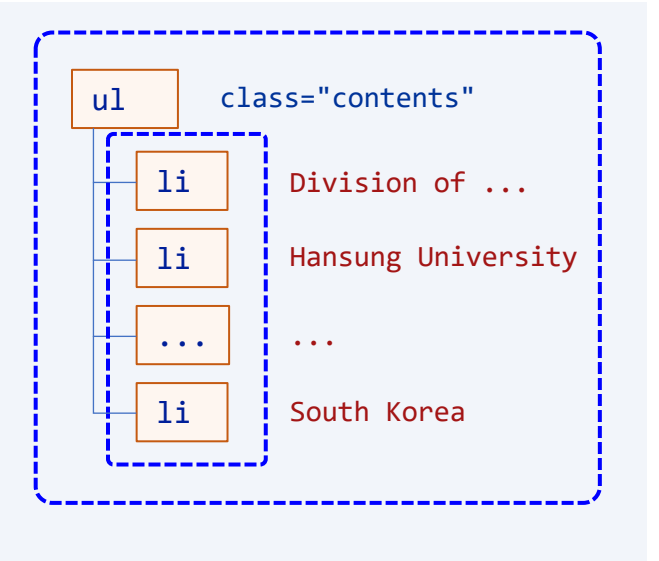



```
/* ex05-02-6.html */
...
const Address =
  return React.createElement(...);
}

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

```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

참고13~14: 함수형 컴포넌트에서 JSX 활용 (react@18)

```
/* ex05-02-7.html */
const Address = function(){
  return React.createElement(
    "ul",
    { className: "address" },
    address.map( (addr, i) => React.createElement("li", {key: i}, addr) )
  );
}

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

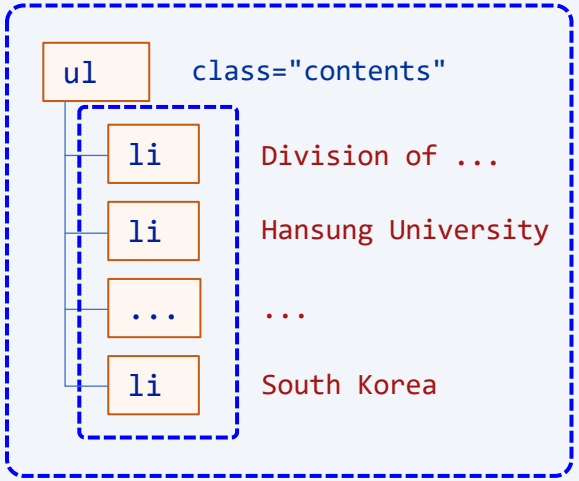



```
/* ex05-02-8.html */
const Address = function(props){
  return React.createElement(
    "ul",
    { className: "address" },
    props.data.map( (addr, i) => React.createElement("li", {key: i}, addr) )
  );
}

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

```
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];
```

실습15~16: 클래스형 컴포넌트 (react@16, react@18)

```
/* ex05-03-1.html */
class Address extends React.Component {
    
}

ReactDOM.render(
    React.createElement(Address, { address }, null),
    document.getElementById("root")
);
```




```
/* ex05-02-2.html */
class Address extends React.Component {
    
}

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

```
const address = [
  "Division of Computer Engineering",
  "Hansung University",
  "116",
  "Samseongyo-ro",
  "16-gil",
  "Seongbuk-gu",
  "Seoul",
  "02876",
  "South Korea"
];
```