2. React

2-1. 리액트 설치

https://ko.reactjs.org/docs/getting-started.html

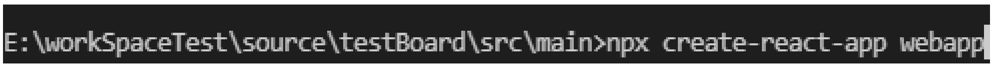
https://github.com/facebook/create-react-app 퀵실행해도 되지만 create react app사용

window+r cmd npm-v(node-v) - nodejs - npm - create react app

npm install -g create-react-app (어디서든 사용하게)

sudo npm install -g create-react-app (권한이 없어서 에러가 뜰때) - 비번

create-react-app -V



```
E:\workSpaceTest\source\testBoard\src\main>npx create-react-app webapp
```

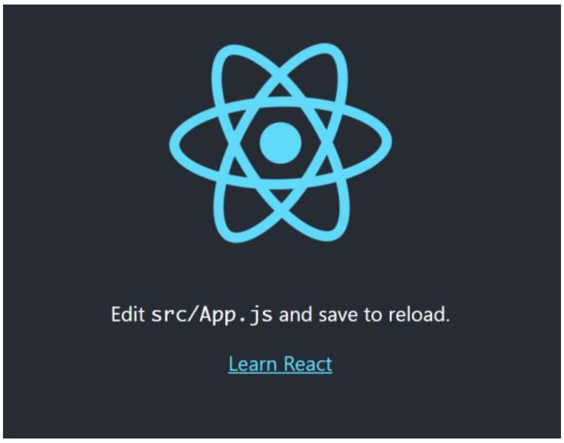
2-2. visual studio code 에디터 설치

view -appreance - terminal -> 터미널 제어 가능

2-3. 기본 서버 실행 및 빌드

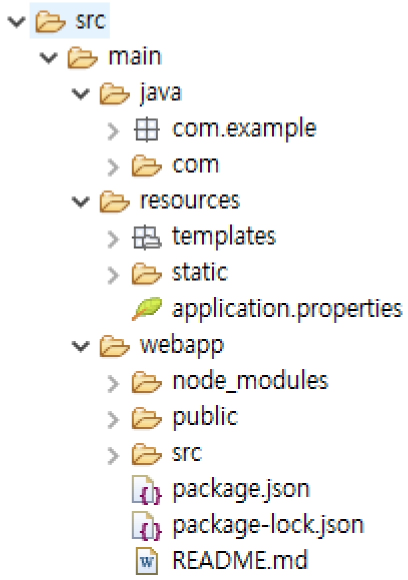
npm run start – 서버 실행

npm run build - 빌드파일 - 이걸 실제 웹서버의 root에 빌드 안쪽 파일에 위치하면 구동됨



Edit src/App.js and save to reload.

2-4. create-react-app 구조 - 5장에서 추가 기술

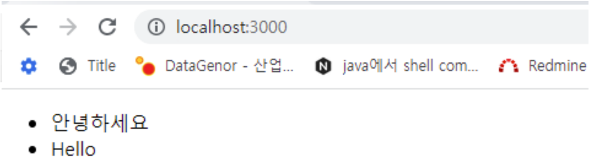


```
src
  main
    java
      com.example
      com
    resources
      templates
      static
      application.properties
    webapp
      node_modules
      public
      src
      package.json
      package-lock.json
      README.md
```

https://kth990303.tistory.com/210

2-5. 리액트가 빠른 이유

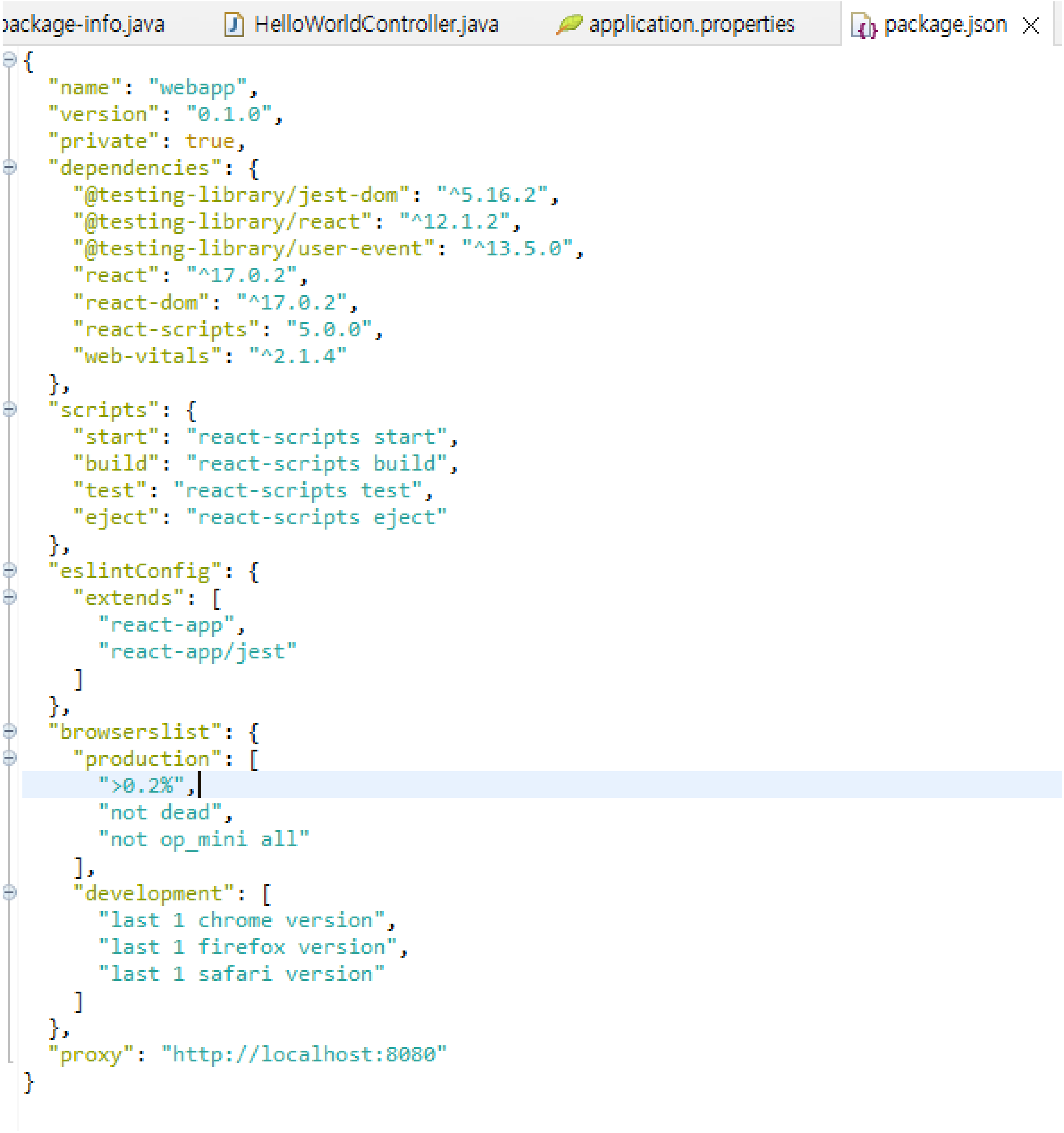
https://ssungkang.tistory.com/entry/React-React-%EC%8B%9C%EC%9E%91%ED%95%98%EA%B8%B0-create-react-app



- 안녕하세요
- Hello

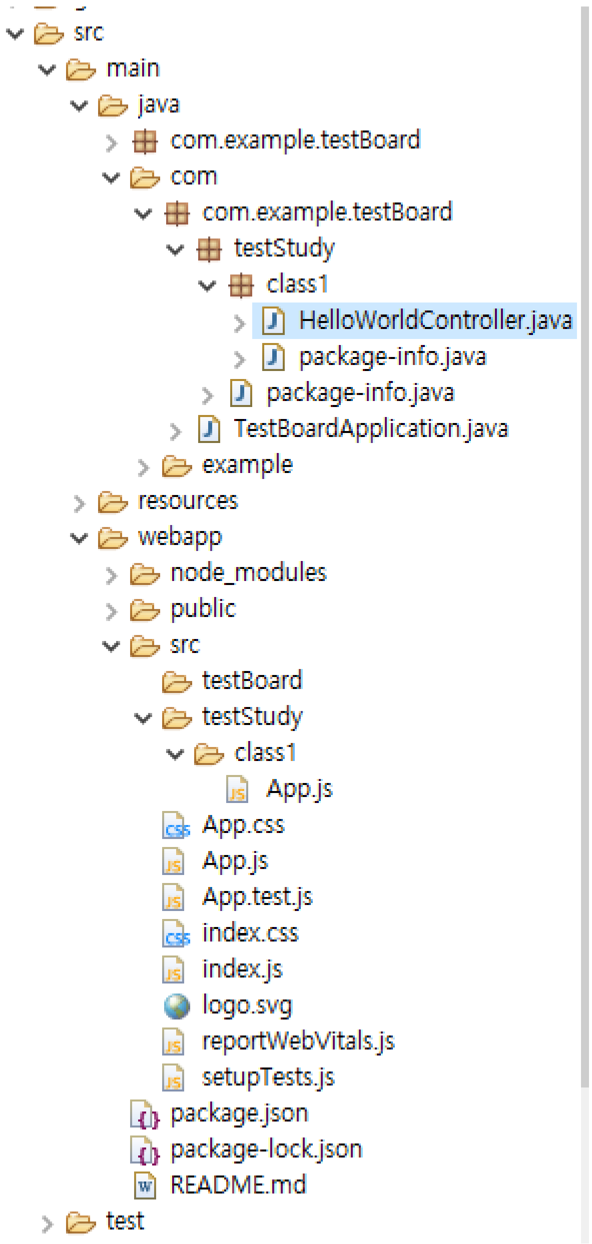
3. React-Spring Boot 연동

3-1. package.json



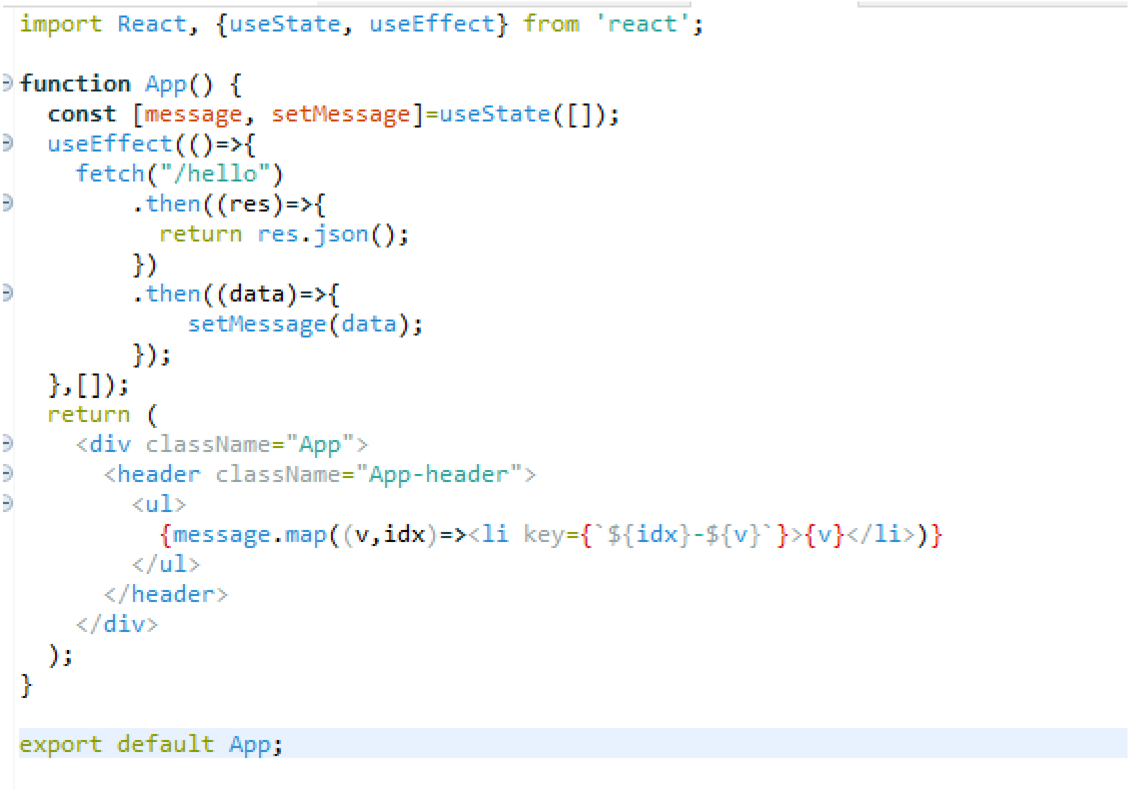
```
{
  "name": "webapp",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@testing-library/jest-dom": "^5.16.2",
    "@testing-library/react": "^12.1.2",
    "@testing-library/user-event": "^13.5.0",
    "react": "^17.0.2",
    "react-dom": "^17.0.2",
    "react-scripts": "5.0.0",
    "web-vitals": "^2.1.4"
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  },
  "proxy": "http://localhost:8080"
}
```

3-2. 현재 패키지 구조

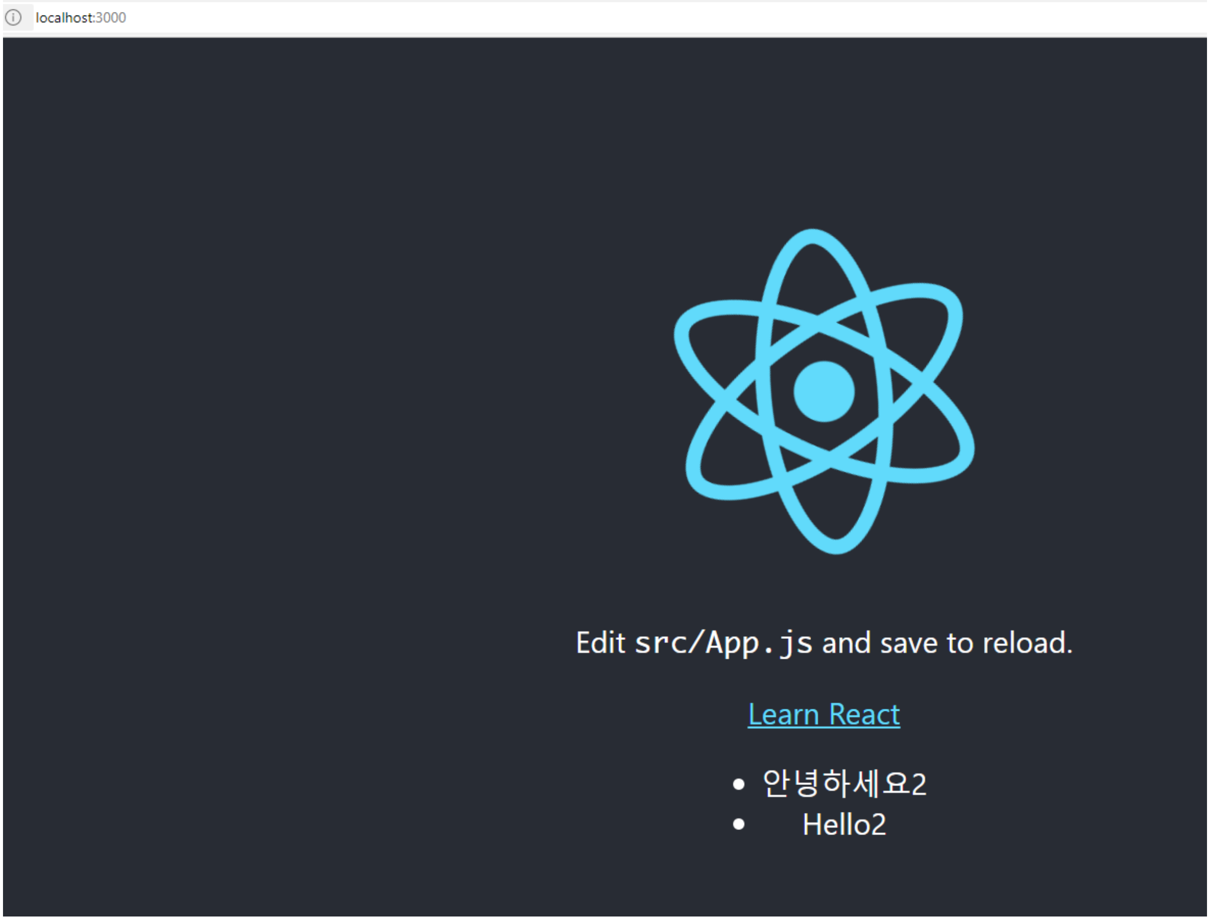


```
src
  main
    java
      com.example.testBoard
      com
        com.example.testBoard
          testStudy
            class1
              HelloWorldController.java
              package-info.java
            package-info.java
          TestBoardApplication.java
        example
    resources
    webapp
      node_modules
      public
      src
        testBoard
        testStudy
          class1
            App.js
        App.css
        App.js
        App.test.js
        index.css
        index.js
        logo.svg
        reportWebVitals.js
        setupTests.js
      package.json
      package-lock.json
      README.md
  test
```

3-3. App.js 그외 소스는 class2참고



```
import React, {useState, useEffect} from 'react';

function App() {
  const [message, setMessage]=useState([]);
  useEffect(()=>{
    fetch("/hello")
        .then((res)=>{
          return res.json();
        })
        .then((data)=>{
            setMessage(data);
        });
  },[]);
  return (
    <div className="App">
      <header className="App-header">
        <ul>
          {message.map((v,idx)=><li key={`${idx}-${v}`}>{v}</li>)}
        </ul>
      </header>
    </div>
  );
}

export default App;
```

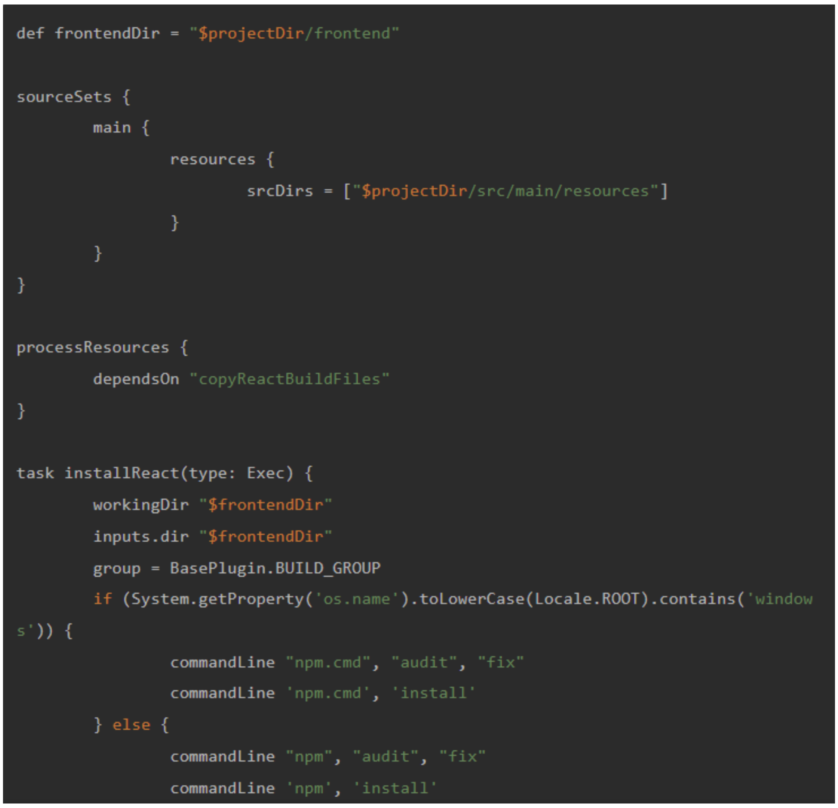


Edit src/App.js and save to reload.

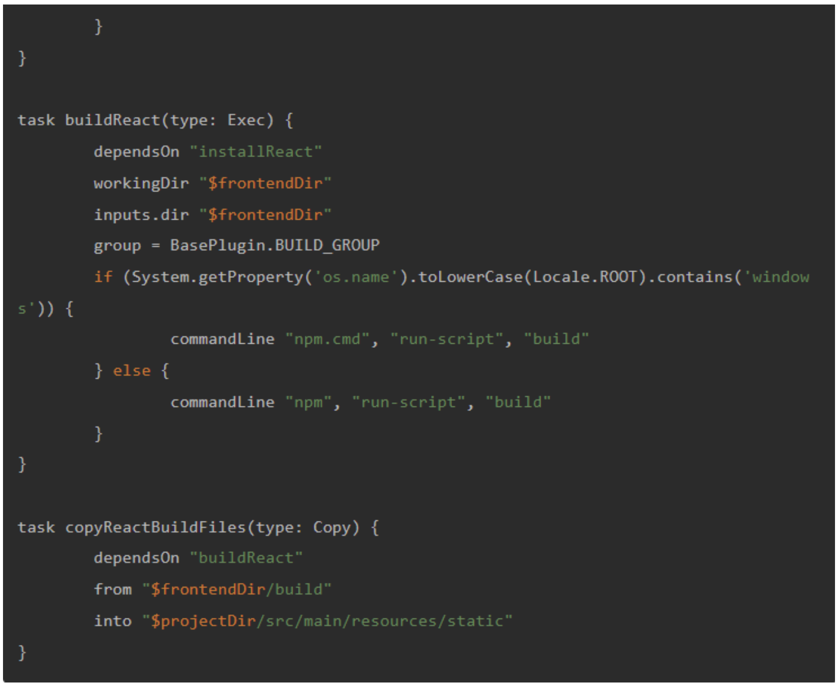
Learn React

- 안녕하세요2
- Hello2

3-4. React-spring build



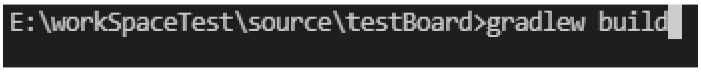
```
def frontendDir = "$projectDir/frontend"

sourceSets {
        main {
                resources {
                        srcDirs = ["$projectDir/src/main/resources"]
                }
        }
}

processResources {
        dependsOn "copyReactBuildFiles"
}

task installReact(type: Exec) {
        workingDir "$frontendDir"
        inputs.dir "$frontendDir"
        group = BasePlugin.BUILD_GROUP
        if (System.getProperty('os.name').toLowerCase(Locale.ROOT).contains('windows')) {
                commandLine "npm.cmd", "audit", "fix"
                commandLine 'npm.cmd', 'install'
        } else {
                commandLine "npm", "audit", "fix"
                commandLine 'npm', 'install'
```



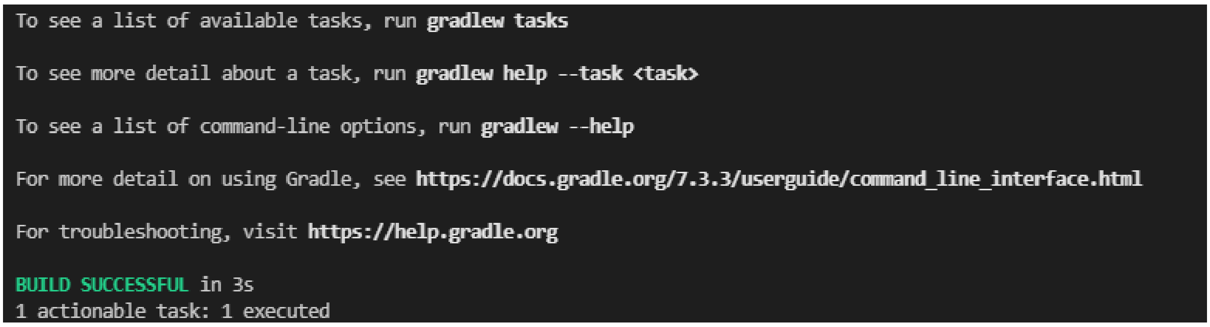
```
        }
}

task buildReact(type: Exec) {
        dependsOn "installReact"
        workingDir "$frontendDir"
        inputs.dir "$frontendDir"
        group = BasePlugin.BUILD_GROUP
        if (System.getProperty('os.name').toLowerCase(Locale.ROOT).contains('windows')) {
                commandLine "npm.cmd", "run-script", "build"
        } else {
                commandLine "npm", "run-script", "build"
        }
}

task copyReactBuildFiles(type: Copy) {
        dependsOn "buildReact"
        from "$frontendDir/build"
        into "$projectDir/src/main/resources/static"
}
```

설정한 build.gradle의 내용은 SpringBoot 프로젝트가 build 될 때 React 프로젝트를 먼저 build 하고 결과물을 SpringBoot 프로젝트 build 결과물에 포함시킨다는 스크립트

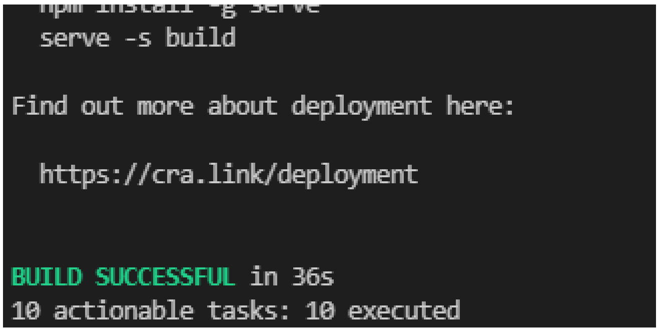
빌드



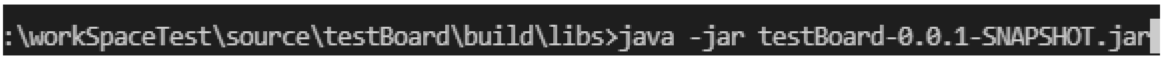
```
E:\workSpaceTest\source\testBoard>gradlew build
```



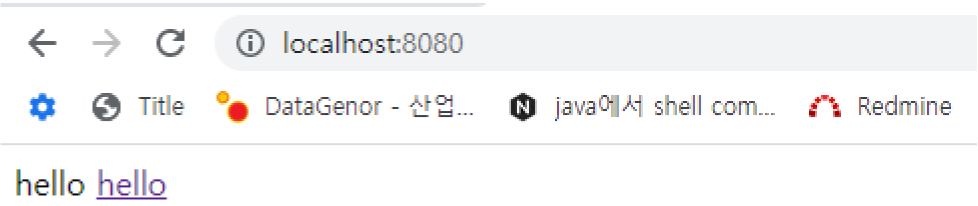
```
To see a list of available tasks, run gradlew tasks

To see more detail about a task, run gradlew help --task <task>

To see a list of command-line options, run gradlew --help

For more detail on using Gradle, see https://docs.gradle.org/7.3.3/userguide/command_line_interface.html

For troubleshooting, visit https://help.gradle.org

BUILD SUCCESSFUL in 3s
1 actionable task: 1 executed
```



```
  npm install -g serve
  serve -s build

Find out more about deployment here:

  https://cra.link/deployment


BUILD SUCCESSFUL in 36s
10 actionable tasks: 10 executed
```



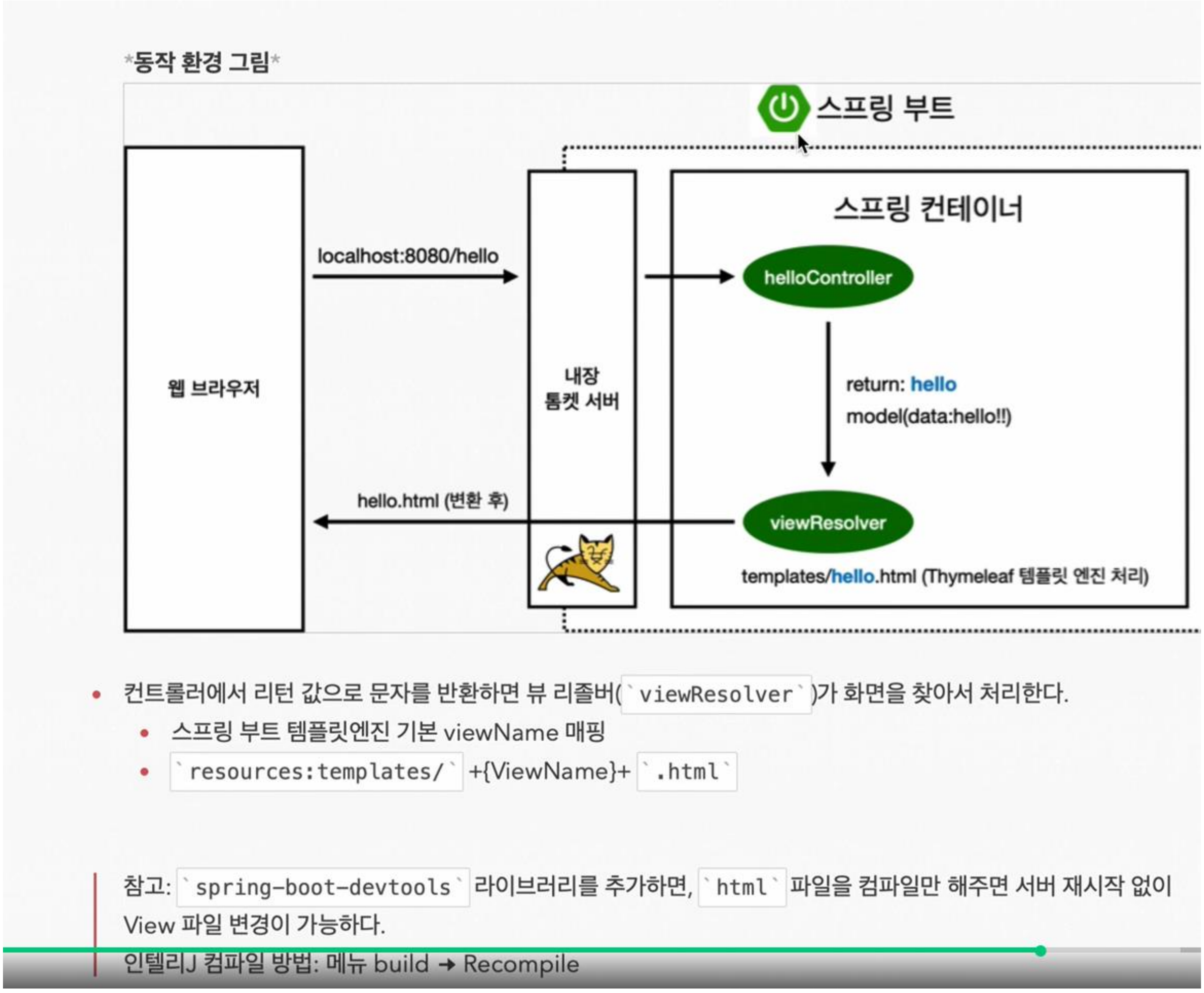
```
:\workSpaceTest\source\testBoard\build\libs>java -jar testBoard-0.0.1-SNAPSHOT.jar
```



hello hello

참고 : https://7942yongdae.tistory.com/136

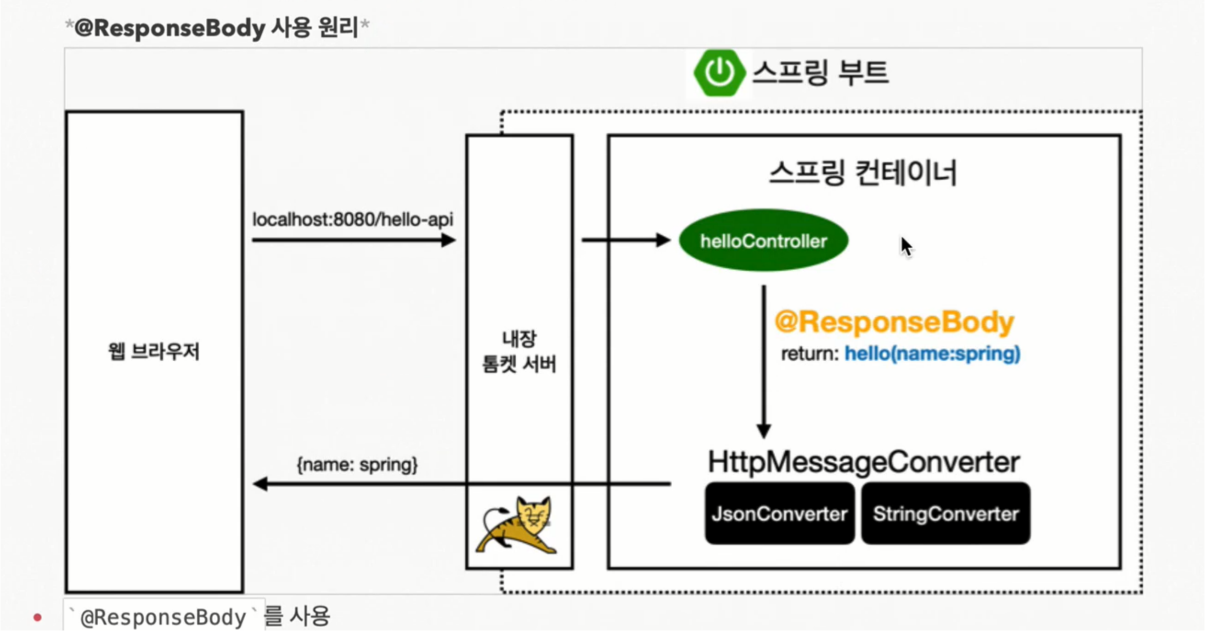
3-5. 동작 프로세스



동작 환경 그림

- 컨트롤러에서 리턴 값으로 문자를 반환하면 뷰 리졸버(`viewResolver`)가 화면을 찾아서 처리한다.
  - 스프링 부트 템플릿엔진 기본 viewName 매핑
  - `resources:templates/` +{ViewName}+ `.html`

> 참고: `spring-boot-devtools` 라이브러리를 추가하면, `html` 파일을 컴파일만 해주면 서버 재시작 없이 View 파일 변경이 가능하다.
> 인텔리J 컴파일 방법: 메뉴 build → Recompile



@ResponseBody 사용 원리

- `@ResponseBody`를 사용

5. React

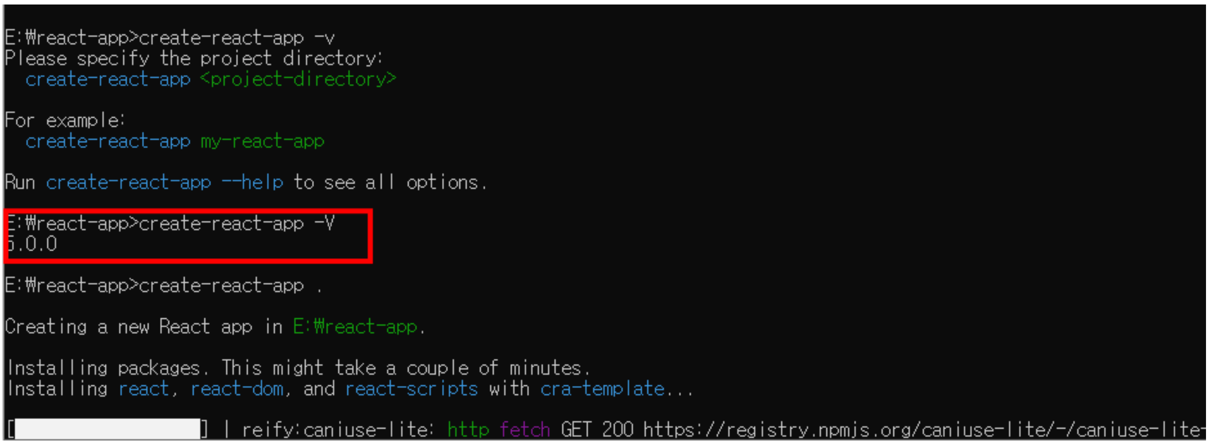
5-1. Creat react app 실행

https://ko.reactjs.org/docs/create-a-new-react-app.html#create-react-app

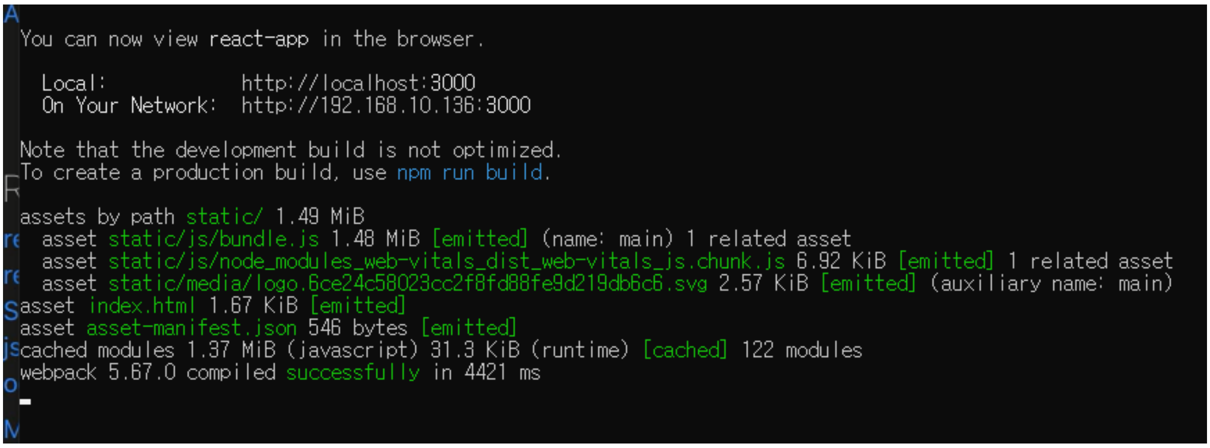
npm install -g create-react-app (어디서든 사용하게)

sudo npm install -g create-react-app (권한이 없어서 에러가 뜰때) - 비번

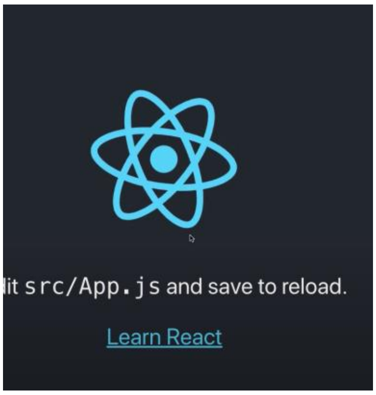
create-react-app -V



```
E:₩react-app>create-react-app -v
Please specify the project directory:
  create-react-app <project-directory>

For example:
  create-react-app my-react-app

Run create-react-app --help to see all options.

E:₩react-app>create-react-app -V
5.0.0

E:₩react-app>create-react-app .

Creating a new React app in E:₩react-app.

 Installing packages. This might take a couple of minutes.
 Installing react, react-dom, and react-scripts with cra-template...

[                  ] | reify:caniuse-lite: http fetch GET 200 https://registry.npmjs.org/caniuse-lite/-/caniuse-lite-
```



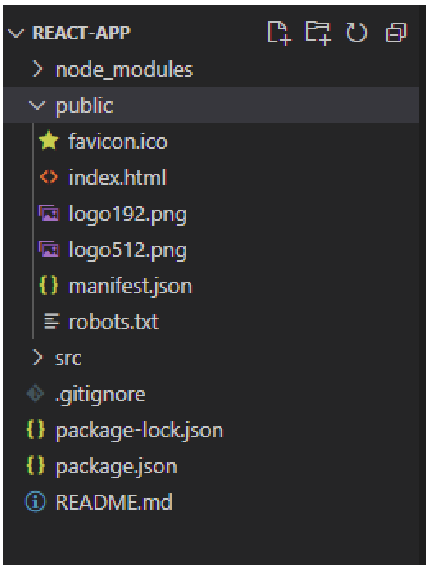
```
 You can now view react-app in the browser.

   Local:            http://localhost:3000
   On Your Network:  http://192.168.10.136:3000

 Note that the development build is not optimized.
 To create a production build, use npm run build.

 assets by path static/ 1.49 MiB
   asset static/js/bundle.js 1.48 MiB [emitted] (name: main) 1 related asset
   asset static/js/node_modules_web-vitals_dist_web-vitals_js.chunk.js 6.92 KiB [emitted] 1 related asset
   asset static/media/logo.6ce24c58023cc2f8fd88fe9d219db6c6.svg 2.57 KiB [emitted] (auxiliary name: main)
 asset index.html 1.67 KiB [emitted]
 asset asset-manifest.json 546 bytes [emitted]
 cached modules 1.37 MiB (javascript) 31.3 KiB (runtime) [cached] 122 modules
 webpack 5.67.0 compiled successfully in 4421 ms
```



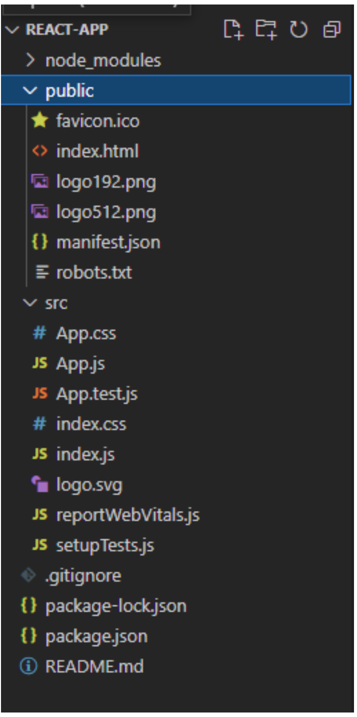
lit src/App.js and save to reload.

Learn React

5-2. 프로젝트 구조



```
REACT-APP
> node_modules
v public
    favicon.ico
    index.html
    logo192.png
    logo512.png
    manifest.json
    robots.txt
> src
  .gitignore
  package-lock.json
  package.json
  README.md
```



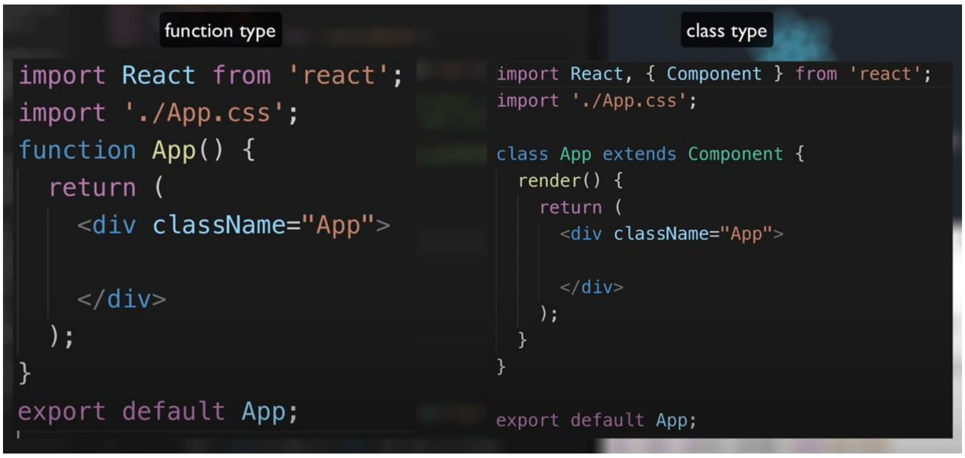
```
REACT-APP
> node_modules
v public
    favicon.ico
    index.html
    logo192.png
    logo512.png
    manifest.json
    robots.txt
v src
    App.css
    App.js
    App.test.js
    index.css
    index.js
    logo.svg
    reportWebVitals.js
    setupTests.js
  .gitignore
  package-lock.json
  package.json
  README.md
```

package.json: 스프링에서의 build.gradle 과 같은 기능

gitignore: 여기에 적힌 파일들은 깃허브에 올리지 않는다.,

readme.md: 프로젝트 설명 (깃허브 리드미)이다.

5-2-1. 리액트 소스 타입



function type

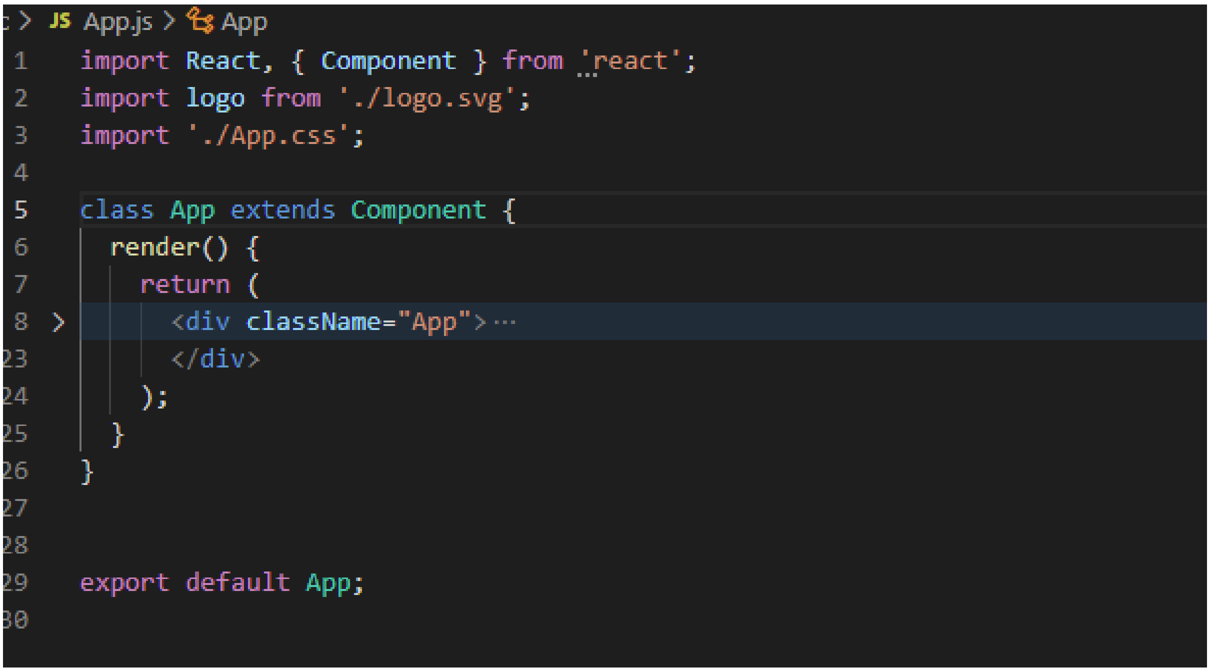
```
import React from 'react';
import './App.css';
function App() {
  return (
    <div className="App">

    </div>
  );
}
export default App;
```

class type

```
import React, { Component } from 'react';
import './App.css';

class App extends Component {
  render() {
    return (
      <div className="App">

      </div>
    );
  }
}

export default App;
```

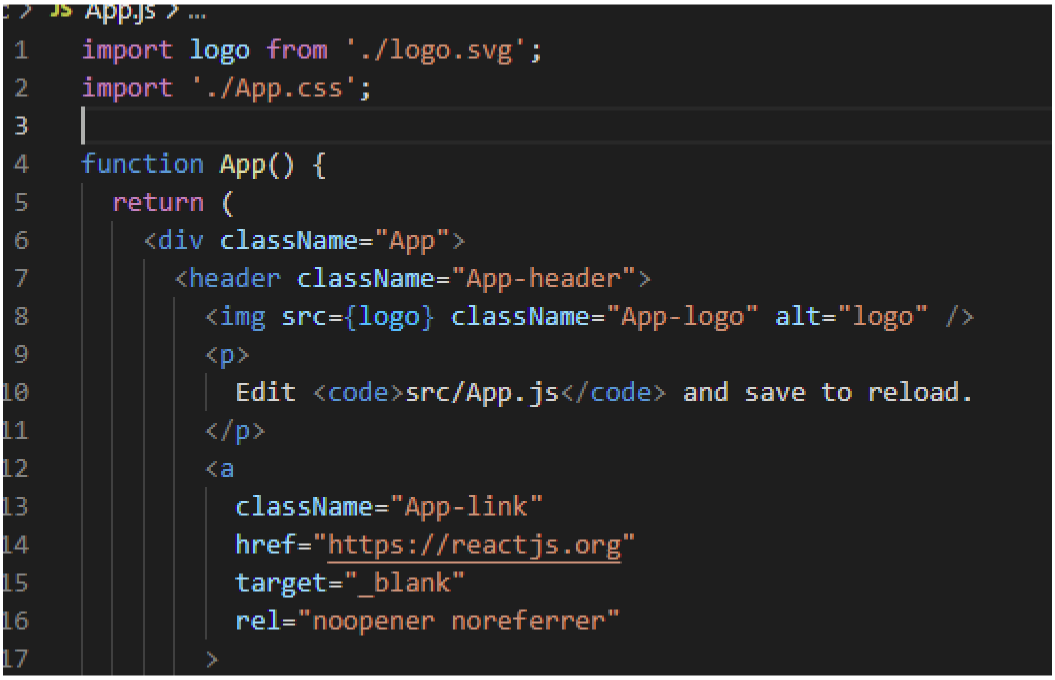
5-3. 기본 시작

Index.js



```
import React, { Component } from 'react';
import logo from './logo.svg';
import './App.css';

class App extends Component {
  render() {
    return (
      <div className="App">⋯
      </div>
    );
  }
}


export default App;

```

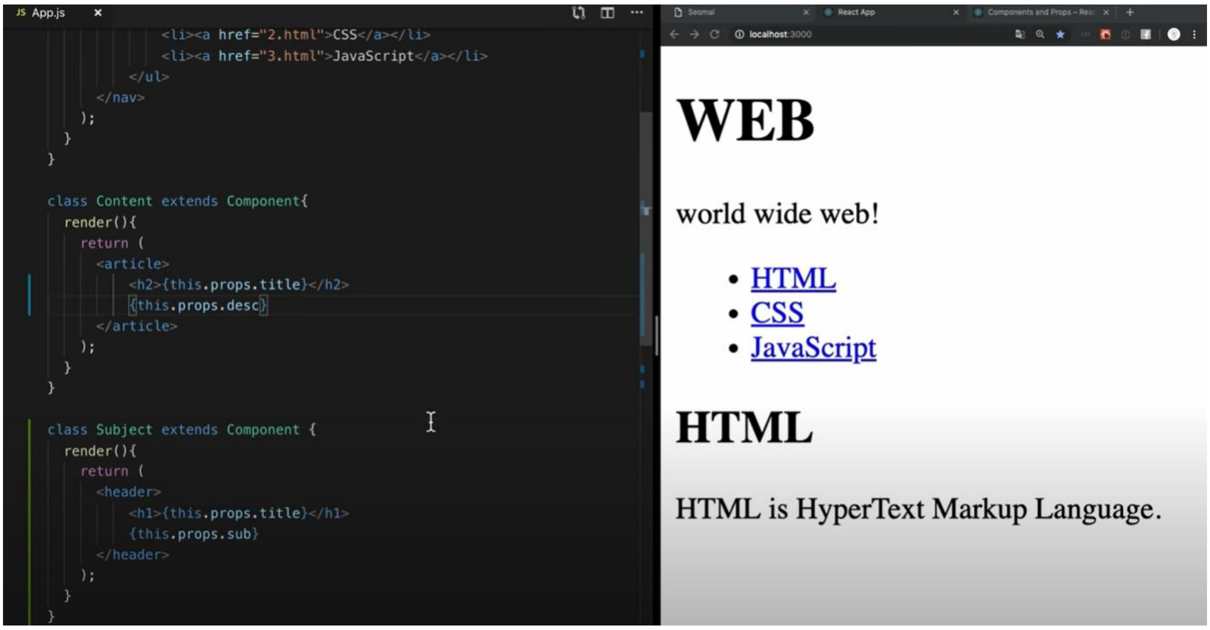
App.js



```
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <p>
          Edit <code>src/App.js</code> and save to reload.
        </p>
        <a
          className="App-link"
          href="https://reactjs.org"
          target="_blank"
          rel="noopener noreferrer"
        >
```

5-4. props vs state

props



App.js

```
            <li><a href="2.html">CSS</a></li>
            <li><a href="3.html">JavaScript</a></li>
          </ul>
        </nav>
      );
    }
  }

  class Content extends Component{
    render(){
      return (
        <article>
            <h2>{this.props.title}</h2>
            {this.props.desc}
        </article>
      );
    }
  }

  class Subject extends Component {
    render(){
      return (
        <header>
            <h1>{this.props.title}</h1>
            {this.props.sub}
        </header>
      );
    }
  }
```

localhost:3000

WEB

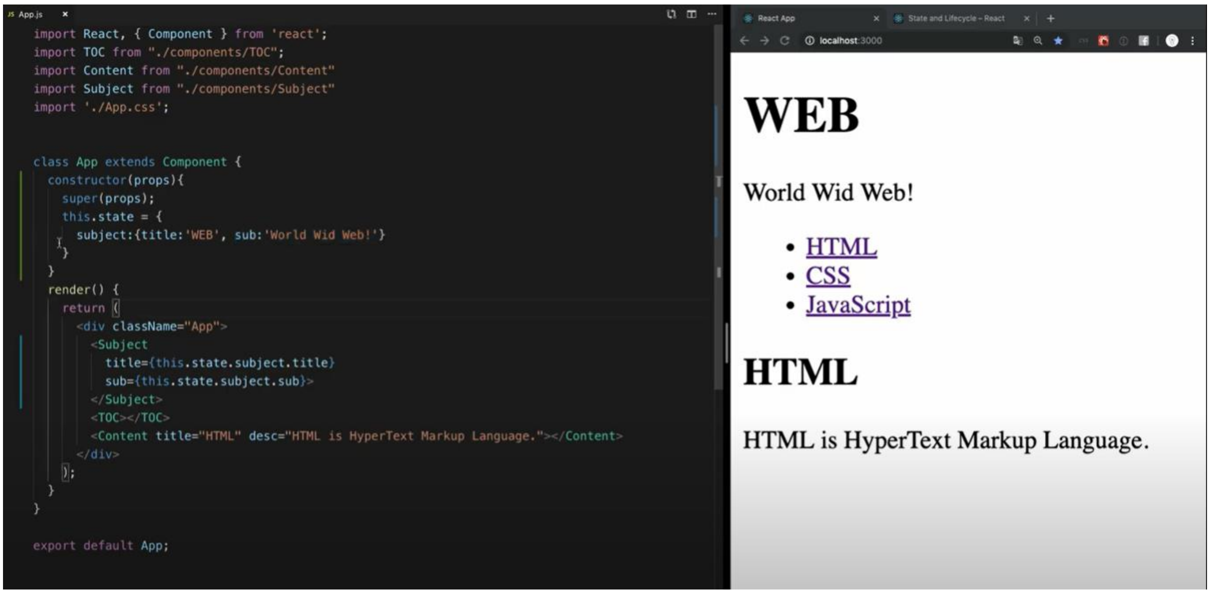
world wide web!

- HTML
- CSS
- JavaScript

HTML

HTML is HyperText Markup Language.

state



App.js

```
import React, { Component } from 'react';
import TOC from "./components/TOC";
import Content from "./components/Content"
import Subject from "./components/Subject"
import './App.css';


class App extends Component {
  constructor(props){
    super(props);
    this.state = {
      subject:{title:'WEB', sub:'World Wid Web!'}
    }
  }
  render() {
    return (
      <div className="App">
        <Subject
          title={this.state.subject.title}
          sub={this.state.subject.sub}>
        </Subject>
        <TOC></TOC>
        <Content title="HTML" desc="HTML is HyperText Markup Language."></Content>
      </div>
    );
  }
}

export default App;
```

localhost:3000

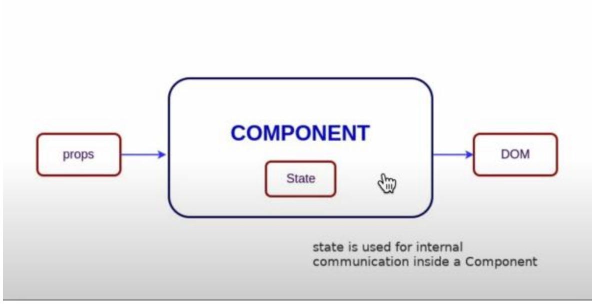
WEB

World Wid Web!

- HTML
- CSS
- JavaScript

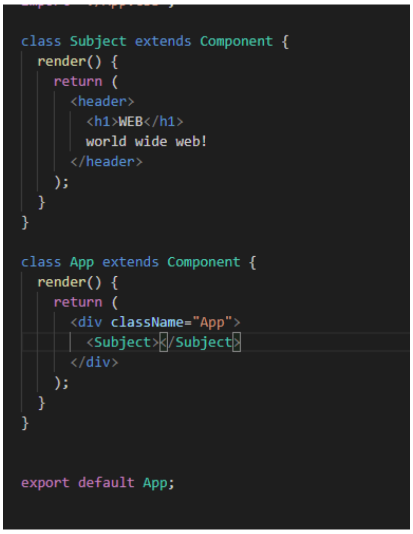
HTML

HTML is HyperText Markup Language.

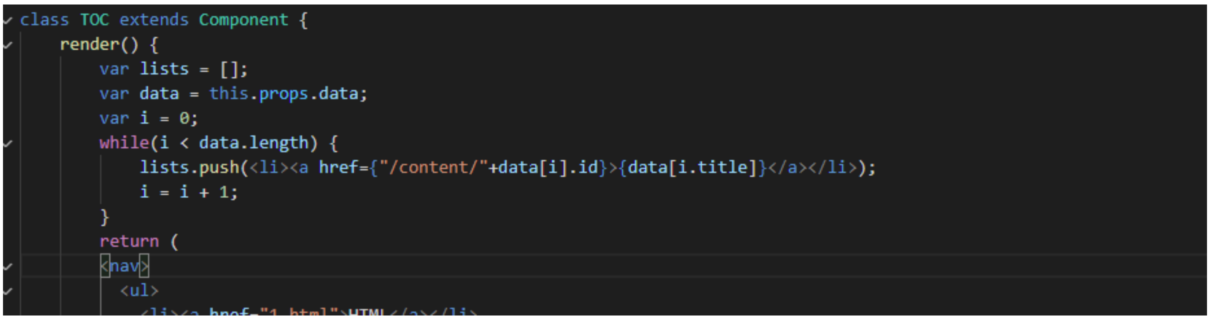


state is used for internal communication inside a Component

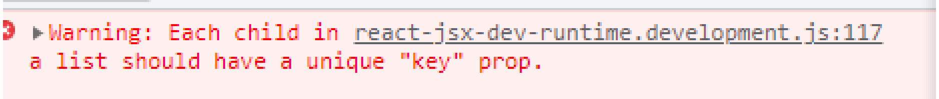
5-5. 컴퍼넌트 생성



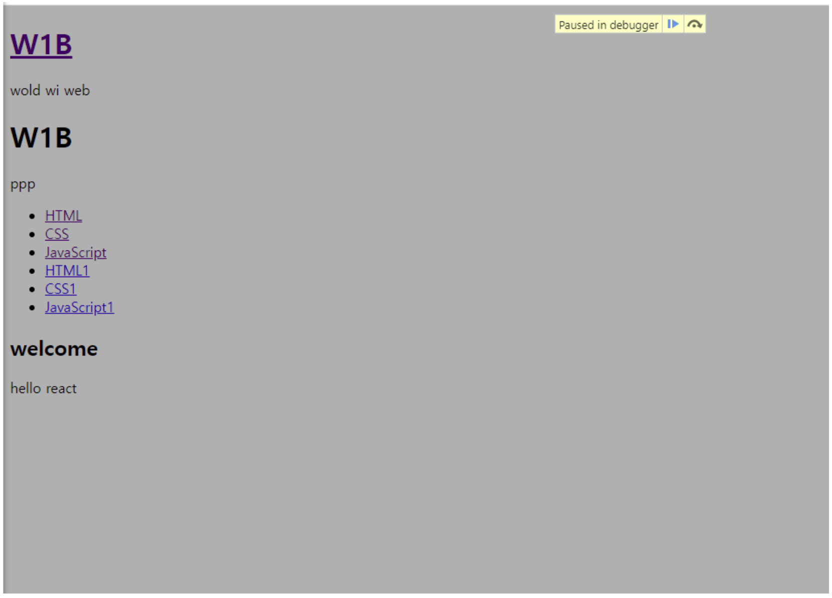
```
class Subject extends Component {
  render() {
    return (
      <header>
        <h1>WEB</h1>
        world wide web!
      </header>
    );
  }
}

class App extends Component {
  render() {
    return (
      <div className="App">
        <Subject></Subject>
      </div>
    );
  }
}


export default App;
```



```
class TOC extends Component {
    render() {
        var lists = [];
        var data = this.props.data;
        var i = 0;
        while(i < data.length) {
            lists.push(<li><a href={"/content/"+data[i].id}>{data[i.title]}</a></li>);
            i = i + 1;
        }
        return (
        <nav>
          <ul>
            <li><a href="1.html">HTML</a></li>
```



```
▶Warning: Each child in react-jsx-dev-runtime.development.js:117
a list should have a unique "key" prop.
```



W1B

wold wi web

W1B

ppp

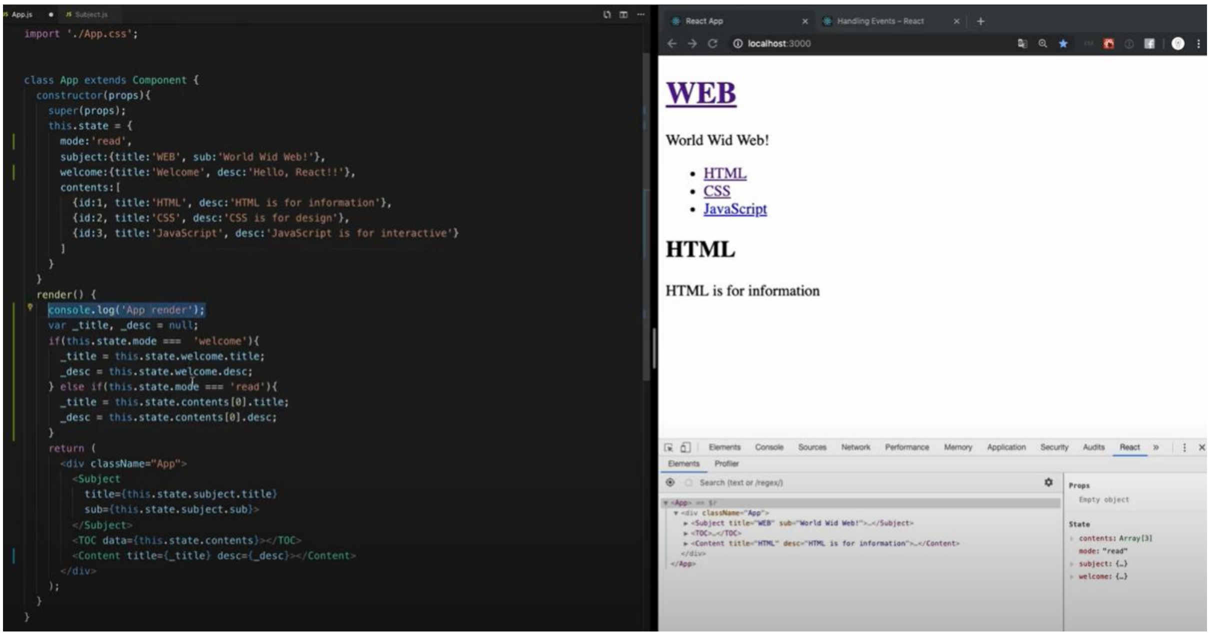
- HTML
- CSS
- JavaScript
- HTML1
- CSS1
- JavaScript1

welcome

hello react

5-6. 이벤트 설치

Render 호출 확인

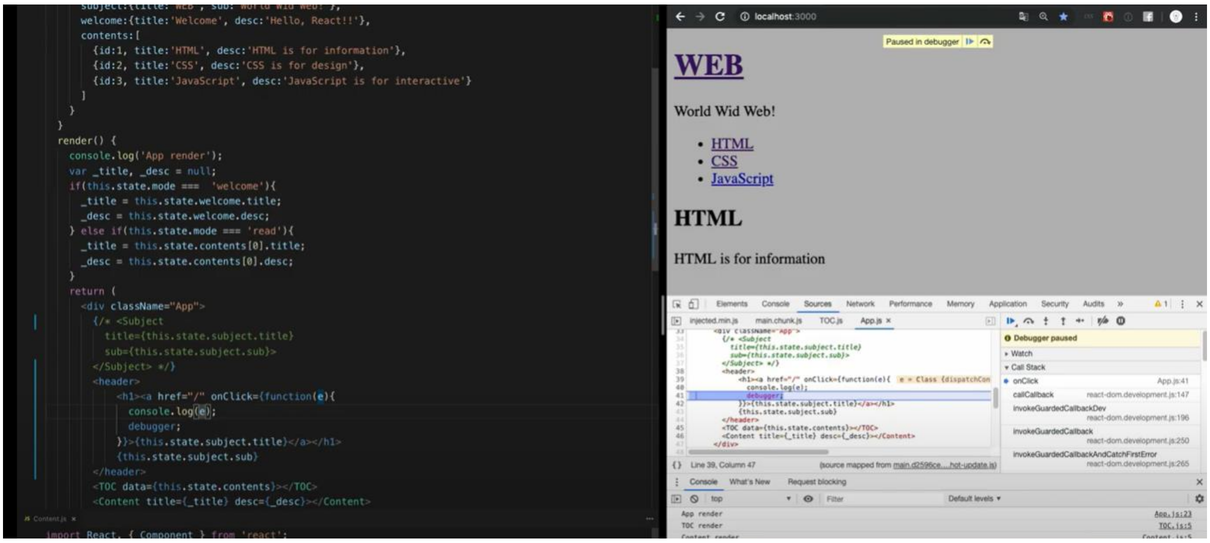


```
import './App.css';


class App extends Component {
  constructor(props){
    super(props);
    this.state = {
      mode:'read',
      subject:{title:'WEB', sub:'World Wid Web!'},
      welcome:{title:'Welcome', desc:'Hello, React!!'},
      contents:[
        {id:1, title:'HTML', desc:'HTML is for information'},
        {id:2, title:'CSS', desc:'CSS is for design'},
        {id:3, title:'JavaScript', desc:'JavaScript is for interactive'}
      ]
    }
  }
  render() {
    console.log('App render');
    var _title, _desc = null;
    if(this.state.mode ===  'welcome'){
      _title = this.state.welcome.title;
      _desc = this.state.welcome.desc;
    } else if(this.state.mode === 'read'){
      _title = this.state.contents[0].title;
      _desc = this.state.contents[0].desc;
    }
    return (
      <div className="App">
        <Subject
          title={this.state.subject.title}
          sub={this.state.subject.sub}>
        </Subject>
        <TOC data={this.state.contents}></TOC>
        <Content title={_title} desc={_desc}></Content>
      </div>
    );
  }
}
```

```
<App> == $r
  <div className="App">
    <Subject title="WEB" sub="World Wid Web!">…</Subject>
    <TOC>…</TOC>
    <Content title="HTML" desc="HTML is for information">…</Content>
  </div>
</App>
```

```
Props
  Empty object

State
  contents: Array(3)
  mode: "read"
  subject: {…}
  welcome: {…}
```

onClick

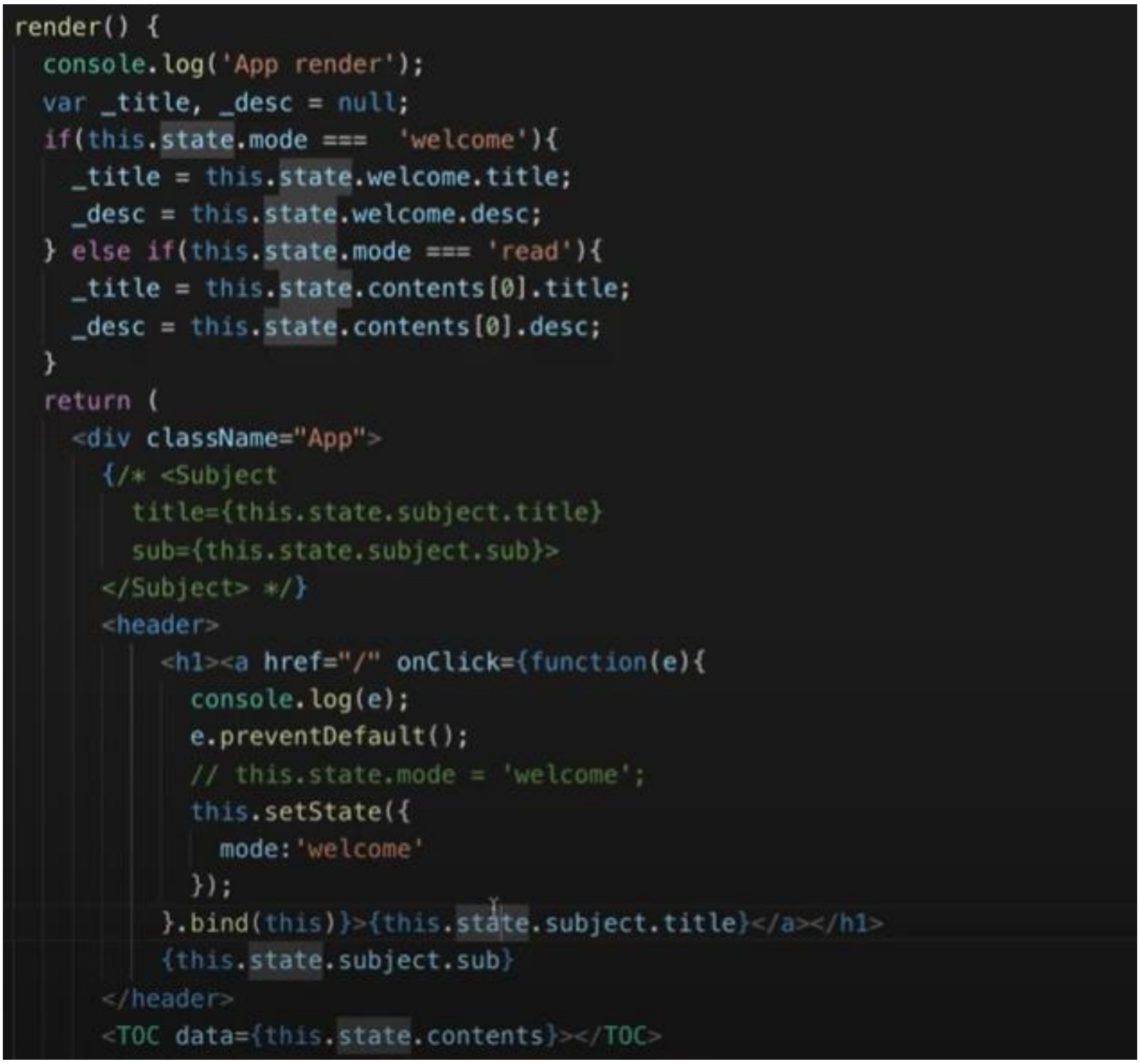


```
      welcome:{title:'Welcome', desc:'Hello, React!!'},
      contents:[
        {id:1, title:'HTML', desc:'HTML is for information'},
        {id:2, title:'CSS', desc:'CSS is for design'},
        {id:3, title:'JavaScript', desc:'JavaScript is for interactive'}
      ]
    }
  }
  render() {
    console.log('App render');
    var _title, _desc = null;
    if(this.state.mode ===  'welcome'){
      _title = this.state.welcome.title;
      _desc = this.state.welcome.desc;
    } else if(this.state.mode === 'read'){
      _title = this.state.contents[0].title;
      _desc = this.state.contents[0].desc;
    }
    return (
      <div className="App">
        {/* <Subject
          title={this.state.subject.title}
          sub={this.state.subject.sub}>
        </Subject> */}
        <header>
            <h1><a href="/" onClick={function(e){
              console.log(e);
              debugger;
            }}>{this.state.subject.title}</a></h1>
            {this.state.subject.sub}
        </header>
        <TOC data={this.state.contents}></TOC>
        <Content title={_title} desc={_desc}></Content>
```

```
Debugger paused
Call Stack
  onClick                                   App.js:41
  callCallback          react-dom.development.js:147
  invokeGuardedCallbackDev
                        react-dom.development.js:196
  invokeGuardedCallback
                        react-dom.development.js:250
  invokeGuardedCallbackAndCatchFirstError
                        react-dom.development.js:265
```

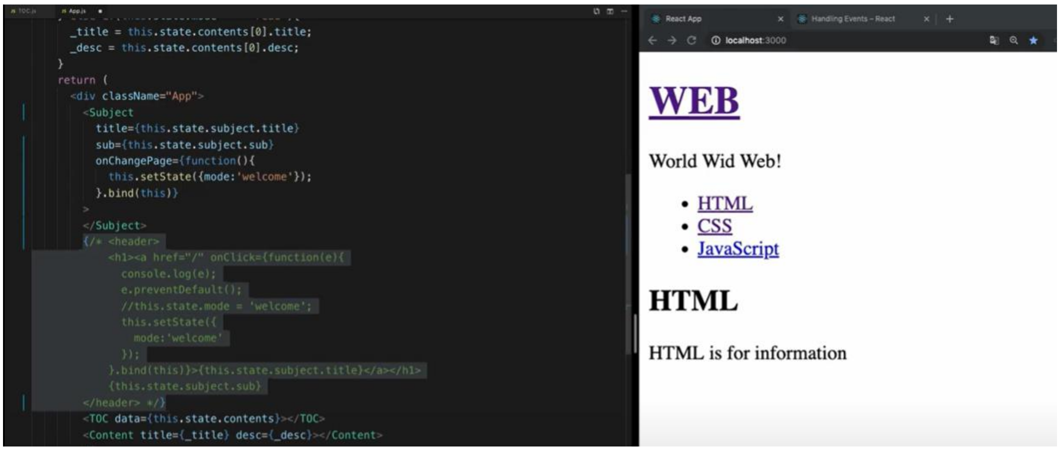
```
App render                                  App.js:23
TOC render                                  TOC.js:5
Content render                              Content.js:5
```

setState



```
render() {
  console.log('App render');
  var _title, _desc = null;
  if(this.state.mode ===  'welcome'){
    _title = this.state.welcome.title;
    _desc = this.state.welcome.desc;
  } else if(this.state.mode === 'read'){
    _title = this.state.contents[0].title;
    _desc = this.state.contents[0].desc;
  }
  return (
    <div className="App">
      {/* <Subject
        title={this.state.subject.title}
        sub={this.state.subject.sub}>
      </Subject> */}
      <header>
          <h1><a href="/" onClick={function(e){
            console.log(e);
            e.preventDefault();
            // this.state.mode = 'welcome';
            this.setState({
              mode:'welcome'
            });
          }.bind(this)}>{this.state.subject.title}</a></h1>
          {this.state.subject.sub}
      </header>
      <TOC data={this.state.contents}></TOC>
```

5-7. 컴퍼넌트 만들기



```
    _title = this.state.contents[0].title;
    _desc = this.state.contents[0].desc;
  }
  return (
    <div className="App">
      <Subject
        title={this.state.subject.title}
        sub={this.state.subject.sub}
        onChangePage={function(){
          this.setState({mode:'welcome'});
        }.bind(this)}
      >
      </Subject>
      {/* <header>
          <h1><a href="/" onClick={function(e){
            console.log(e);
            e.preventDefault();
            //this.state.mode = 'welcome';
            this.setState({
              mode:'welcome'
            });
          }.bind(this)}>{this.state.subject.title}</a></h1>
          {this.state.subject.sub}
      </header> */}
      <TOC data={this.state.contents}></TOC>
      <Content title={_title} desc={_desc}></Content>
```

WEB

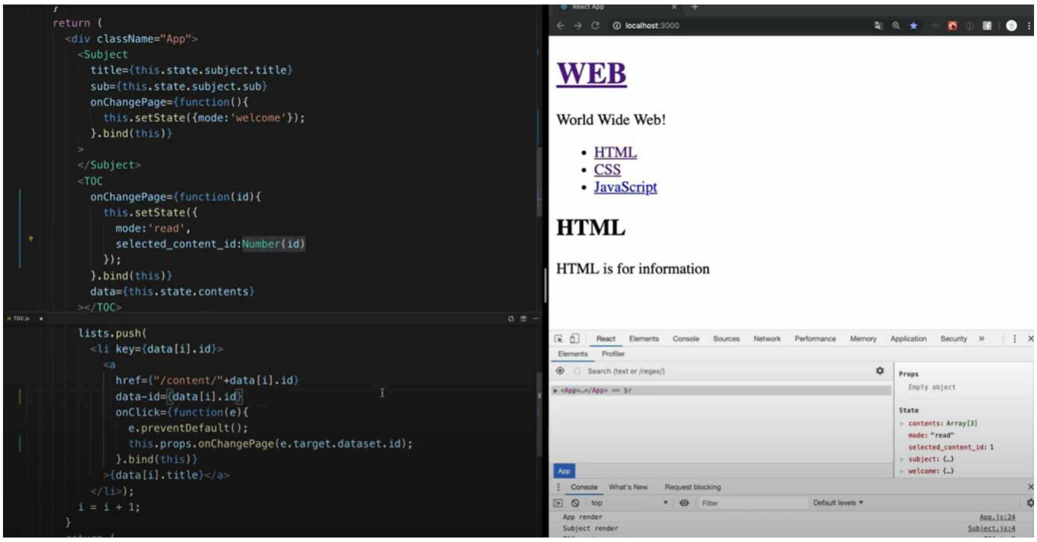
World Wid Web!

- HTML
- CSS
- JavaScript

HTML

HTML is for information

Toc



```
  return (
    <div className="App">
      <Subject
        title={this.state.subject.title}
        sub={this.state.subject.sub}
        onChangePage={function(){
          this.setState({mode:'welcome'});
        }.bind(this)}
      >
      </Subject>
      <TOC
        onChangePage={function(id){
          this.setState({
            mode:'read',
            selected_content_id:Number(id)
          });
        }.bind(this)}
        data={this.state.contents}
      ></TOC>
```

```
      lists.push(
        <li key={data[i].id}>
          <a
            href={"/content/"+data[i].id}
            data-id={data[i].id}
            onClick={function(e){
              e.preventDefault();
              this.props.onChangePage(e.target.dataset.id);
            }.bind(this)}
          >{data[i].title}</a>
        </li>);
      i = i + 1;
    }
```

WEB

World Wide Web!

- HTML
- CSS
- JavaScript

HTML

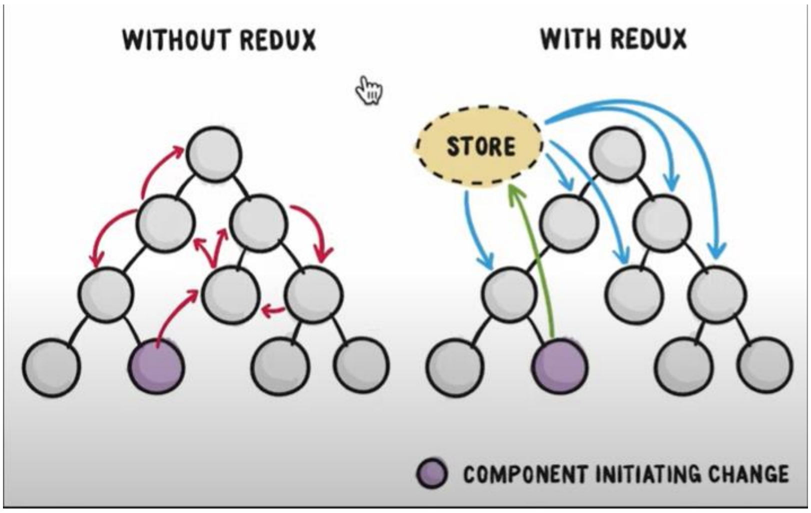
HTML is for information

```
▶ <App>…</App> == $r
```

```
Props
  Empty object

State
▶ contents: Array[3]
  mode: "read"
  selected_content_id: 1
▶ subject: {…}
▶ welcome: {…}
```

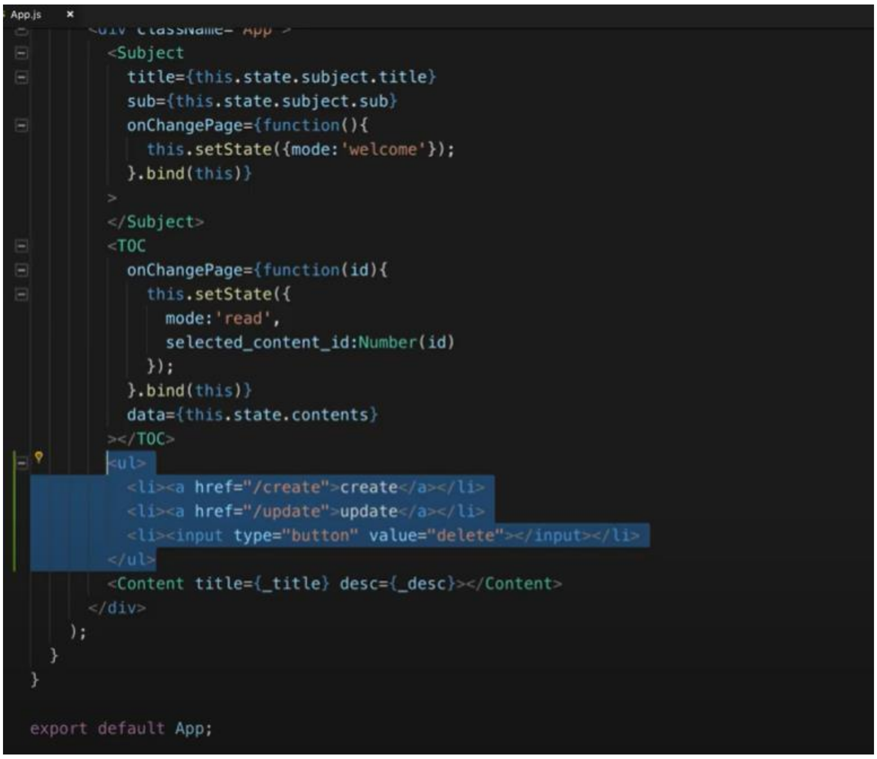
```
App render                                   App.js:24
Subject render                           Subject.js:4
```

5-7-1. Redux



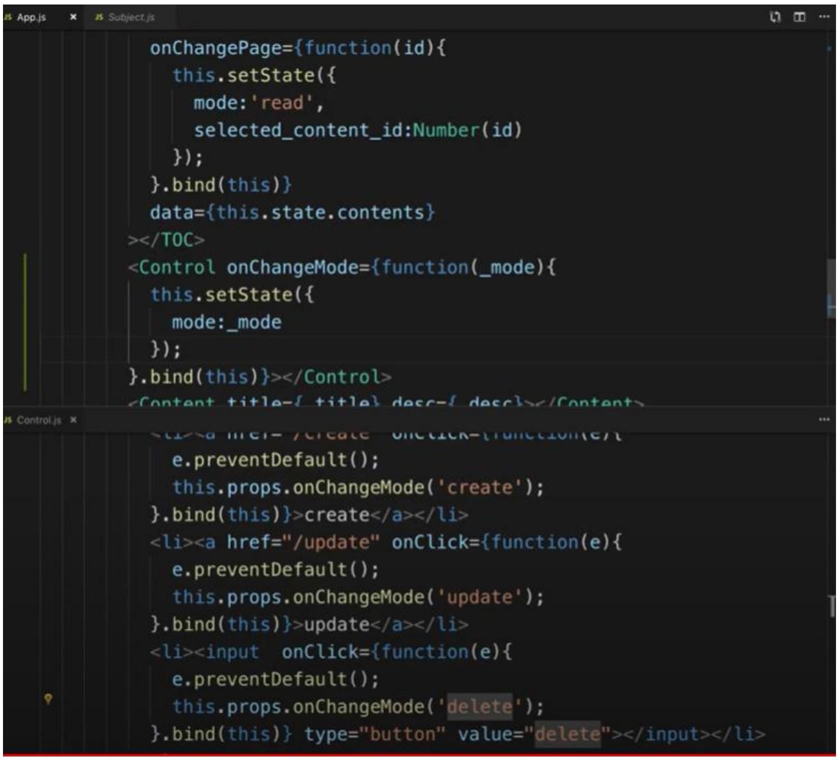
5-8. crud – create

App.js



```
        <div className="App">
          <Subject
            title={this.state.subject.title}
            sub={this.state.subject.sub}
            onChangePage={function(){
              this.setState({mode:'welcome'});
            }.bind(this)}
          >
          </Subject>
          <TOC
            onChangePage={function(id){
              this.setState({
                mode:'read',
                selected_content_id:Number(id)
              });
            }.bind(this)}
            data={this.state.contents}
          ></TOC>
          <ul>
            <li><a href="/create">create</a></li>
            <li><a href="/update">update</a></li>
            <li><input type="button" value="delete"></input></li>
          </ul>
          <Content title={_title} desc={_desc}></Content>
        </div>
      );
    }
  }

export default App;
```

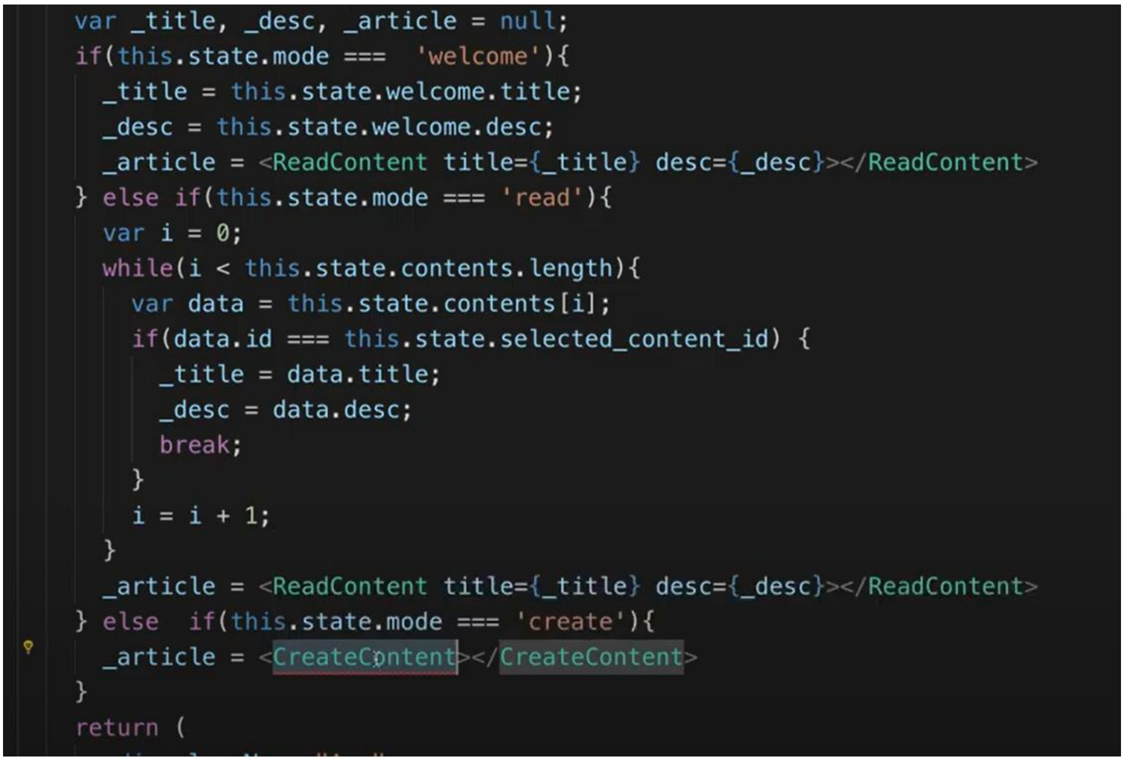
Subject.js



```
            onChangePage={function(id){
              this.setState({
                mode:'read',
                selected_content_id:Number(id)
              });
            }.bind(this)}
            data={this.state.contents}
          ></TOC>
          <Control onChangeMode={function(_mode){
            this.setState({
              mode:_mode
            });
          }.bind(this)}></Control>
          <Content title={_title} desc={_desc}></Content>
```

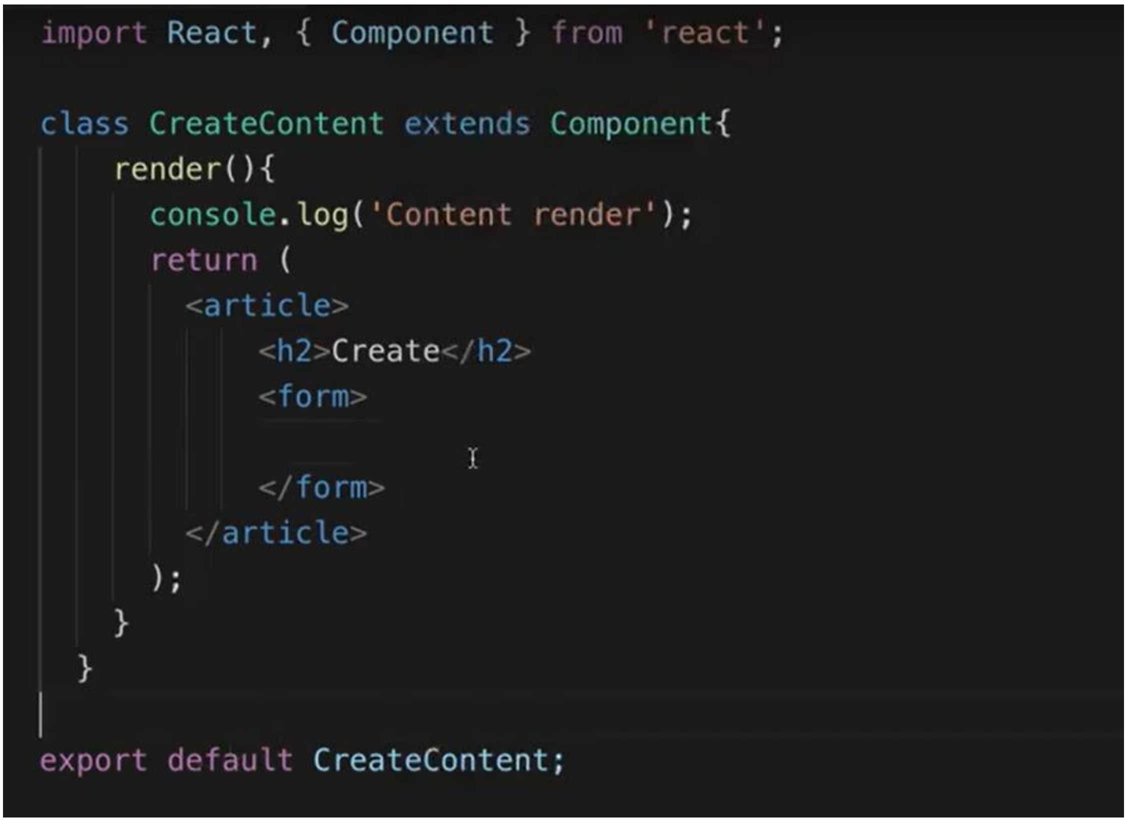
```
            <li><a href="/create" onClick={function(e){
              e.preventDefault();
              this.props.onChangeMode('create');
            }.bind(this)}>create</a></li>
            <li><a href="/update" onClick={function(e){
              e.preventDefault();
              this.props.onChangeMode('update');
            }.bind(this)}>update</a></li>
            <li><input  onClick={function(e){
              e.preventDefault();
              this.props.onChangeMode('delete');
            }.bind(this)} type="button" value="delete"></input></li>
```

App.js

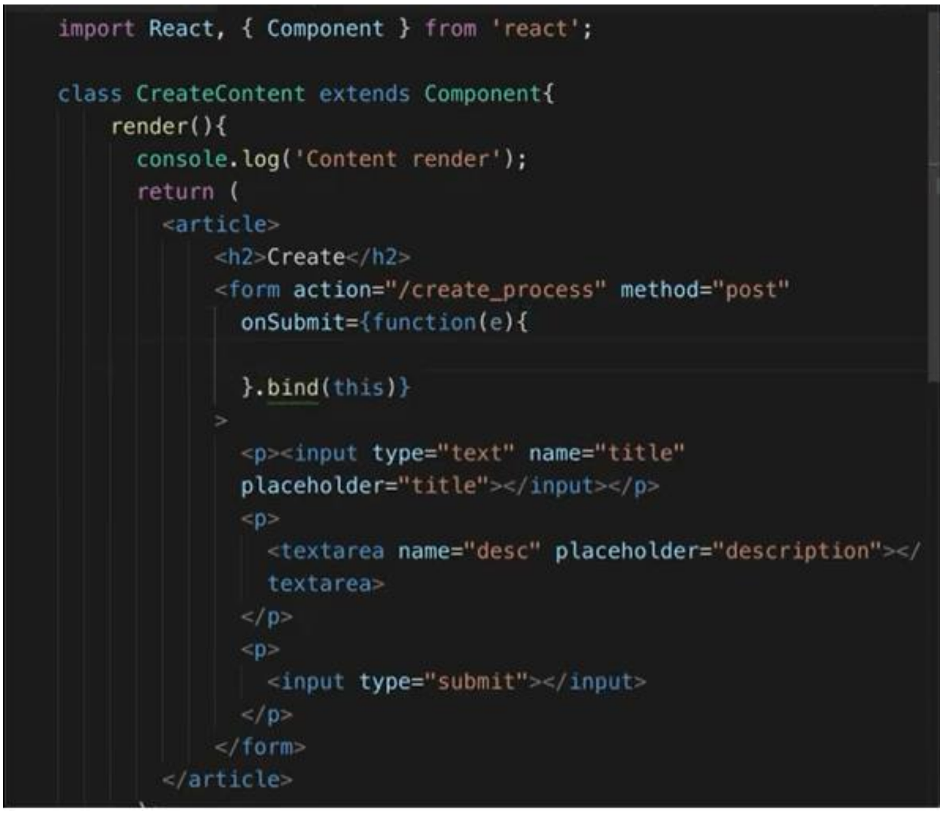


```
var _title, _desc, _article = null;
if(this.state.mode ===  'welcome'){
  _title = this.state.welcome.title;
  _desc = this.state.welcome.desc;
  _article = <ReadContent title={_title} desc={_desc}></ReadContent>
} else if(this.state.mode === 'read'){
  var i = 0;
  while(i < this.state.contents.length){
    var data = this.state.contents[i];
    if(data.id === this.state.selected_content_id) {
      _title = data.title;
      _desc = data.desc;
      break;
    }
    i = i + 1;
  }
  _article = <ReadContent title={_title} desc={_desc}></ReadContent>
} else  if(this.state.mode === 'create'){
  _article = <CreateContent></CreateContent>
}
return (
```

CreateContent.js

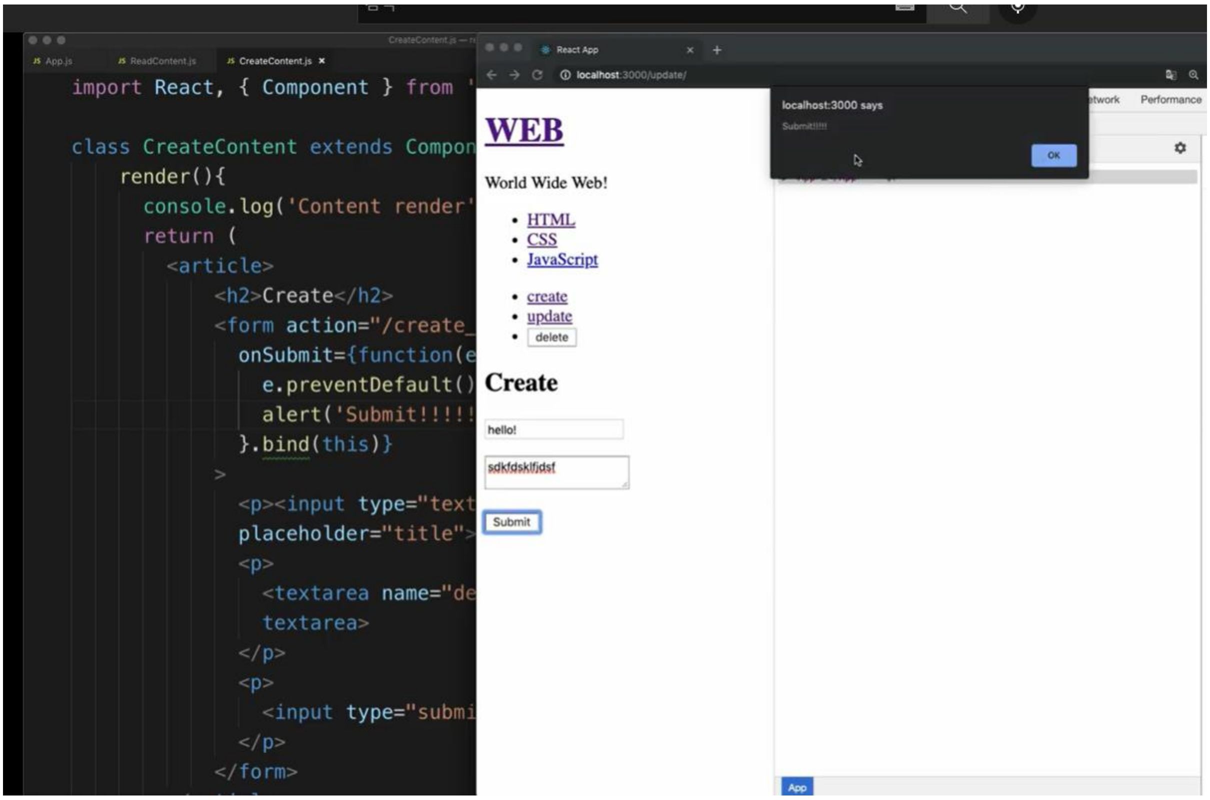


```
import React, { Component } from 'react';

class CreateContent extends Component{
    render(){
      console.log('Content render');
      return (
        <article>
            <h2>Create</h2>
            <form>

            </form>
        </article>
      );
    }
  }

export default CreateContent;
```



```
import React, { Component } from 'react';

class CreateContent extends Component{
    render(){
      console.log('Content render');
      return (
        <article>
            <h2>Create</h2>
            <form action="/create_process" method="post"
              onSubmit={function(e){

              }.bind(this)}
            >
              <p><input type="text" name="title"
              placeholder="title"></input></p>
              <p>
                <textarea name="desc" placeholder="description"></
                textarea>
              </p>
              <p>
                <input type="submit"></input>
              </p>
            </form>
        </article>
```

실행



```
import React, { Component } from '
  
class CreateContent extends Compon
    render(){
      console.log('Content render'
      return (
        <article>
            <h2>Create</h2>
            <form action="/create_
              onSubmit={function(e
                e.preventDefault()
                alert('Submit!!!!!
              }.bind(this)}
            >
              <p><input type="text
              placeholder="title">
              <p>
                <textarea name="de
                textarea>
              </p>
              <p>
                <input type="submi
              </p>
            </form>
```

WEB

World Wide Web!

- HTML
- CSS
- JavaScript

- create
- update
- delete

Create

hello!

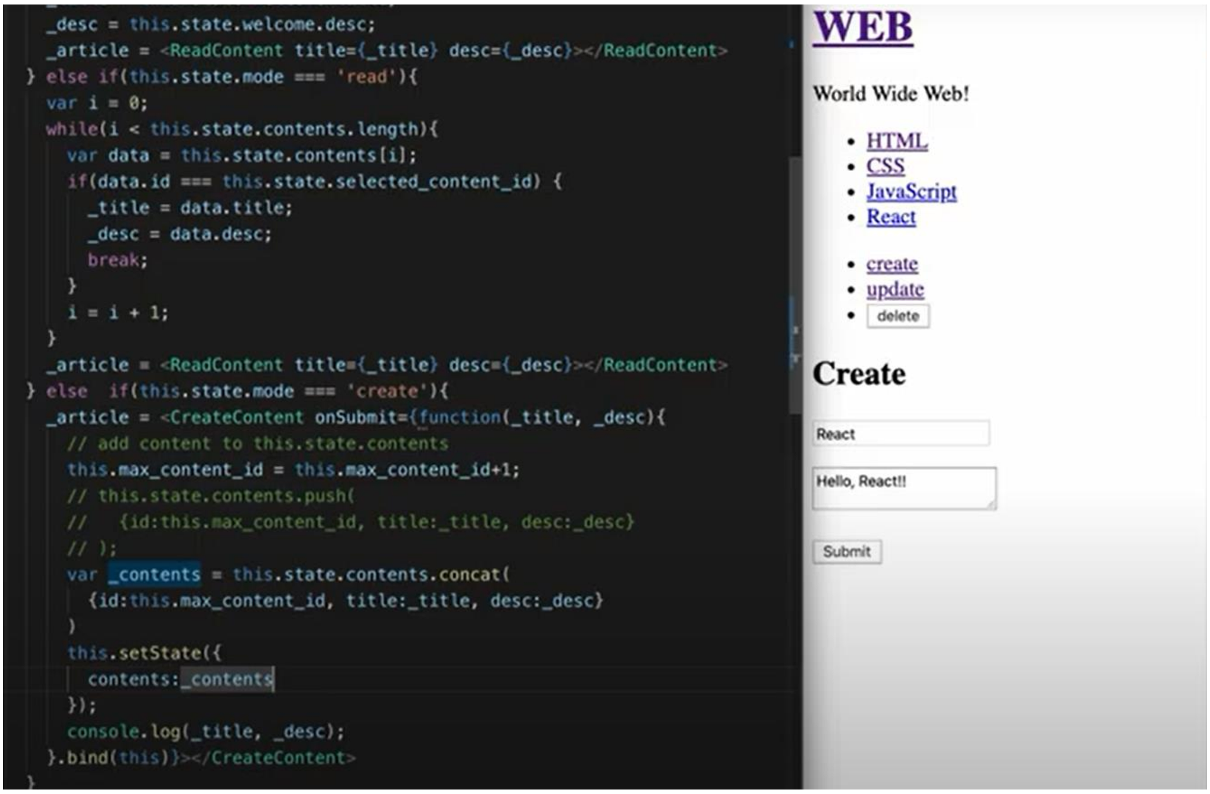
sdkfdsklfjdsf

Submit

localhost:3000 says

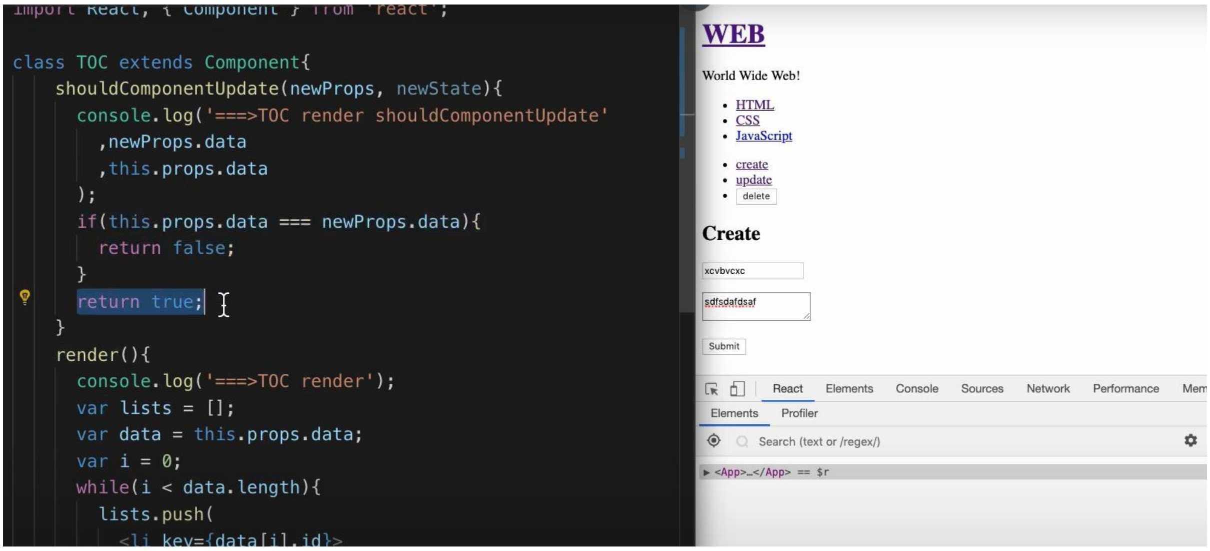
Submit!!!!!

OK



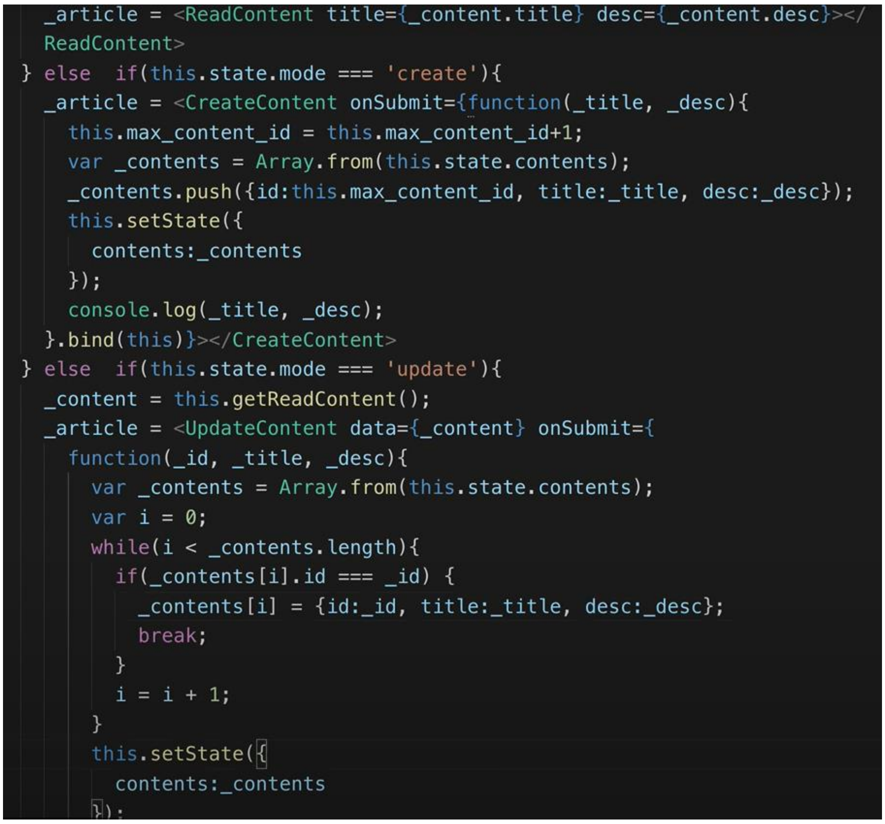
```
  _desc = this.state.welcome.desc;
  _article = <ReadContent title={_title} desc={_desc}></ReadContent>
} else if(this.state.mode === 'read'){
  var i = 0;
  while(i < this.state.contents.length){
    var data = this.state.contents[i];
    if(data.id === this.state.selected_content_id) {
      _title = data.title;
      _desc = data.desc;
      break;
    }
    i = i + 1;
  }
  _article = <ReadContent title={_title} desc={_desc}></ReadContent>
} else  if(this.state.mode === 'create'){
  _article = <CreateContent onSubmit={function(_title, _desc){
    // add content to this.state.contents
    this.max_content_id = this.max_content_id+1;
    // this.state.contents.push(
    //   {id:this.max_content_id, title:_title, desc:_desc}
    // );
    var _contents = this.state.contents.concat(
      {id:this.max_content_id, title:_title, desc:_desc}
    )
    this.setState({
      contents:_contents
    });
    console.log(_title, _desc);
  }.bind(this)}></CreateContent>
}
```

shouldComponetUpdate



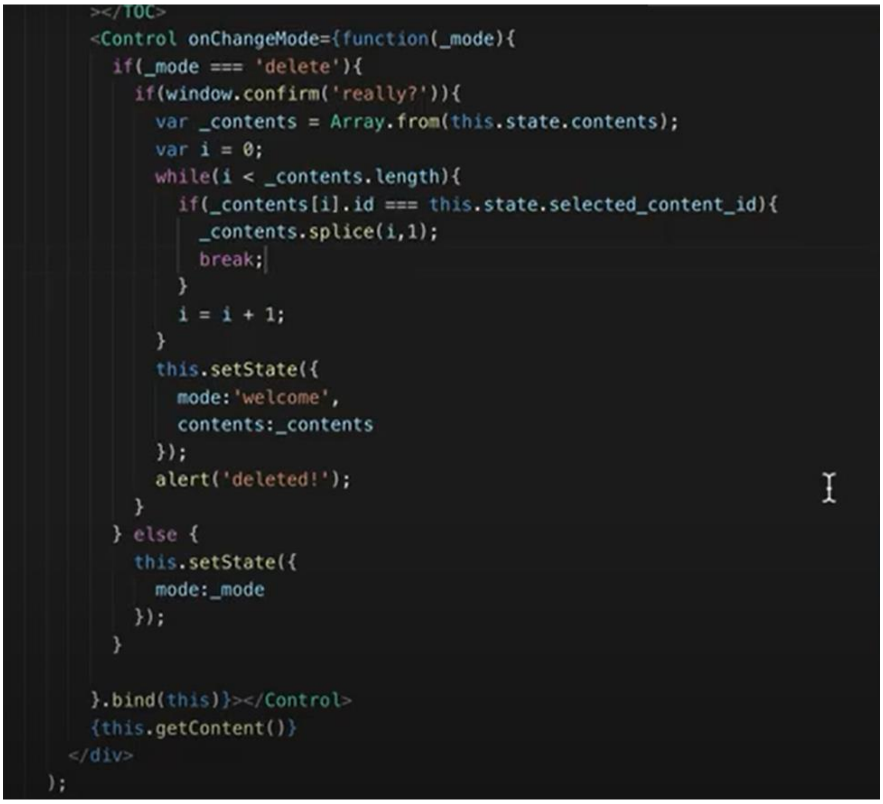
```
import React, { Component } from 'react';

class TOC extends Component{
    shouldComponentUpdate(newProps, newState){
      console.log('===>TOC render shouldComponentUpdate'
        ,newProps.data
        ,this.props.data
      );
      if(this.props.data === newProps.data){
        return false;
      }
      return true;
    }
    render(){
      console.log('===>TOC render');
      var lists = [];
      var data = this.props.data;
      var i = 0;
      while(i < data.length){
        lists.push(
          <li key={data[i].id}>
```

5-9. crud – update



```
  _article = <ReadContent title={_content.title} desc={_content.desc}></
  ReadContent>
} else  if(this.state.mode === 'create'){
  _article = <CreateContent onSubmit={function(_title, _desc){
    this.max_content_id = this.max_content_id+1;
    var _contents = Array.from(this.state.contents);
    _contents.push({id:this.max_content_id, title:_title, desc:_desc});
    this.setState({
      contents:_contents
    });
    console.log(_title, _desc);
  }.bind(this)}></CreateContent>
} else  if(this.state.mode === 'update'){
  _content = this.getReadContent();
  _article = <UpdateContent data={_content} onSubmit={
    function(_id, _title, _desc){
      var _contents = Array.from(this.state.contents);
      var i = 0;
      while(i < _contents.length){
        if(_contents[i].id === _id) {
          _contents[i] = {id:_id, title:_title, desc:_desc};
          break;
        }
        i = i + 1;
      }
      this.setState({
        contents:_contents
      });
```

5-10. crud – delete



```
        ></TOC>
        <Control onChangeMode={function(_mode){
          if(_mode === 'delete'){
            if(window.confirm('really?')){
              var _contents = Array.from(this.state.contents);
              var i = 0;
              while(i < _contents.length){
                if(_contents[i].id === this.state.selected_content_id){
                  _contents.splice(i,1);
                  break;
                }
                i = i + 1;
              }
              this.setState({
                mode:'welcome',
                contents:_contents
              });
              alert('deleted!');
            }
          } else {
            this.setState({
              mode:_mode
            });
          }

        }.bind(this)}></Control>
        {this.getContent()}
      </div>
  );
```