



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

Mobile Computing

3rd Lecture

프로젝트 구성, 변수
JSX, Component, Props, State



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Learning objective

React Native 프로젝트 구성에 대해 살펴보고, 변수 선언 방법을 익힌다. 아울러 React Native는 React와 비슷하지만 빌딩 블록으로 웹 구성 요소 대신 기본 구성 요소를 사용한다. 따라서 본 강의에서는 React Native 앱의 기본 구조를 이해하기 위해 JSX, Component, State 및 props와 같은 몇 가지 기본 React 개념을 이해하고 익힐 수 있도록 한다.

Ref : <https://reactnative.dev/docs/tutorial#components>



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



React Native 프로젝트 구성

```
> _tests_
> .bundle
> .idea
> android
> ios
> node_modules
≡ .buckconfig
⊗ .eslintrc.js
⬢ .gitignore
JS .prettierrc.js
≡ .ruby-version
{} .watchmanconfig
{} app.json
TS App.tsx
Babel.config.js
Gemfile
≡ Gemfile.lock
JS index.js
JS metro.config.js
{} package-lock.json
{} package.json
TS tsconfig.json
```

좌측은 React Native 프로젝트 생성시 볼수 있는 프로젝트 구조이다.

1. index.js

리액트 네이티브 프로젝트의 시작 파일 (앱을 `npx react-native run-android` (Mac의 경우 `npx react-native run-ios`)로 수행할 경우 최초로 실행)

이 파일을 시작으로 리액트 네이티브의 자바스크립트 코드가 번들링(Bundling) 된다. 번들링(Bundling)이란 "어떤 것들을 묶는다"라는 의미로 기능별로 모듈화 했던 자바스크립트 파일들을 묶어 준다는 뜻이다.

```
/**
 * @format
 */

import {AppRegistry} from 'react-native';
import App from './App';
import {name as appName} from './app.json';

AppRegistry.registerComponent(appName, () => App);
```

- 리액트 네이티브는 `AppRegistry.registerComponent`를 사용하여 네이티브 브릿지에서 사용할 모듈을 지정 한다.
- `registerComponent` 함수의 첫 번째 매개변수(`appName`)는 모듈 이름을 지정하며
- 두 번째 매개변수에는 처음으로 렌더링(화면에 표시) 될 컴포넌트를 지정한다



Department of
Media Software

2023 1st semester
Mobile Computing

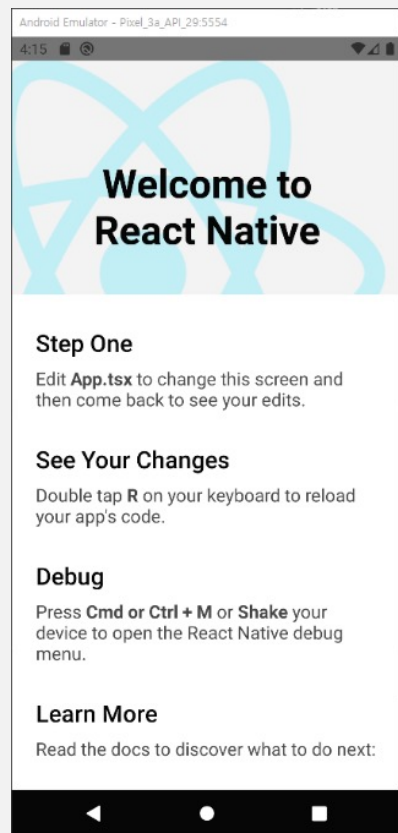
Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



React Native 프로젝트 구성



2. App.tsx (또는 App.js)

아래 화면에 내용이 들어 있는 파일로 react native 에서는 View, Button, Text 등 여러가지 기본 컴포넌트를 제공하며, 실제 디바이스에 표시되는 코드는 App.tsx (또는 App.js)에 작성되어 있고 App.tsx의 소스를 살펴보면 프로그램 코드와 컴포넌트가 XML 문서처럼 결합되어 있는 형태로 이를 JSX(Javascript XML)라 부른다.

```
60
61 const backgroundStyle = {
62   backgroundColor: isDarkMode ? Colors.darker : Colors.lighter,
63 };
64
65
66 return (
67   <SafeAreaView style={backgroundStyle}>
68     <StatusBar barStyle={isDarkMode ? 'light-content' : 'dark-content'} />
69     <ScrollView
70       contentInsetAdjustmentBehavior="automatic"
71       style={backgroundStyle}>
72       <Header />
73       <View
74         style={{
75           backgroundColor: isDarkMode ? Colors.black : Colors.white,
76         }}>
77         <Section title="Step One">
78           Edit <Text style={styles.highlight}>App.tsx</Text> to change this
79           screen and then come back to see your edits.
80         </Section>
81         <Section title="See Your Changes">
82           <ReloadInstructions />
83         </Section>
84         <Section title="Debug">
85           <DebugInstructions />
86         </Section>
87         <Section title="Learn More">
88           Read the docs to discover what to do next:
89           <LearnMoreLinks />
90         </Section>
91       </ScrollView>
92     </SafeAreaView>
93   );
94 }
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



React Native 프로젝트 구성

```
1 {
2   "name": "typescriptstapp",
3   "version": "0.0.1",
4   "private": true,
5   "scripts": {
6     "android": "react-native run-android",
7     "ios": "react-native run-ios",
8     "start": "react-native start",
9     "test": "jest",
10    "lint": "eslint . --ext .js,.jsx,.ts,.tsx"
11  },
12  "dependencies": {
13    "react": "17.0.2",
14    "react-native": "0.67.3"
15  },
16  "devDependencies": {
17    "@babel/core": "^7.12.9",
18    "@babel/runtime": "^7.12.5",
19    "@react-native-community/eslint-config": "^2.0.0",
20    "@types/jest": "^26.0.23",
21    "@types/react-native": "^0.66.15",
22    "@types/react-test-renderer": "^17.0.1",
23    "@typescript-eslint/eslint-plugin": "^5.7.0",
24    "@typescript-eslint/parser": "^5.7.0",
25    "babel-jest": "^26.6.3",
26    "eslint": "^7.14.0",
27    "jest": "^26.6.3",
28    "metro-react-native-babel-preset": "^0.66.2",
29    "react-test-renderer": "17.0.2",
30    "typescript": "^4.4.4"
31  },
32  "resolutions": {
33    "@types/react": "^17"
34  },
35 }
```

3. package.json

해당 파일은 프로젝트와 관련된 모듈들이 기록된 파일로 이를 통해 **모듈들의 버전 관리**를 하게되며, 파일 내용의 중간에 **dependencies** 오브젝트를 보면 내가 설치한 추가 라이브러리 목록을 볼 수 있습니다

프로그램을 진행 하다 보면 설치한 npm 라이브러리들이 서로 충돌하고 앱 실행이 안되기도 하는 경우가 있는데 이럴 때 이 폴더에 들어가 그동안 설치했던 라이브러리들을 확인하고 새로 설치할 수 있습니다



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

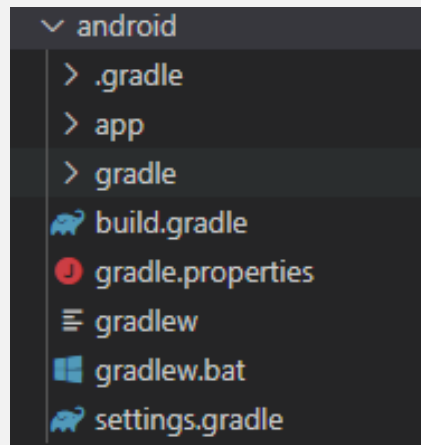
3rd Lecture

Mobile Computing



React Native 프로젝트 구성

4. android 폴더



안드로이드 프로젝트가 담겨 있는 폴더로 안드로이드 네이티브 모듈을 담당하고 있는 부분입니다. react-native에서 지원하지 않는 기능들을 사용하기 위해서는 네이티브 모듈을 직접 구현해야 합니다. 혹은 npm 에 공개 모듈로 올려놓은 것을 다운받아 사용할 수 있는데, 이를 사용하기 위해서는 dependency 등을 설정해야 할 수 있습니다. 주로 이러한 경우에 이 폴더 안에 있는 소스를 수정하게 되겠습니다.

1) build.gradle

안드로이드 앱을 빌드 배포할때 사용하는 파일

2) app/src/main/res

안드로이드 앱의 아이콘 및 시작화면 등의 리소스들을 관리하는 폴더

3) app/src/java/com/앱이름/MainActivity.java, MainApplication.java

안드로이드 앱의 메인 파일

위 네가지 파일은 실제로 각종 안드로이드라이브러리를 설치 및 안드로이드 실행을 위한 중요한 파일들입니다. android 폴더안의 파일들은 안드로이드 앱을 구성하기 위한 설정파일들, 앱아이콘, SPLASH 그림파일 등이 들어 있으며, SPLASH 파일은 앱이 처음실행될때 나오는그림입니다. 또 앱을 apk나 aab파일로 저장하여 배포/복사시에도 이 폴더에저장됩니다.



Department of
Media Software

2023 1st semester
Mobile Computing

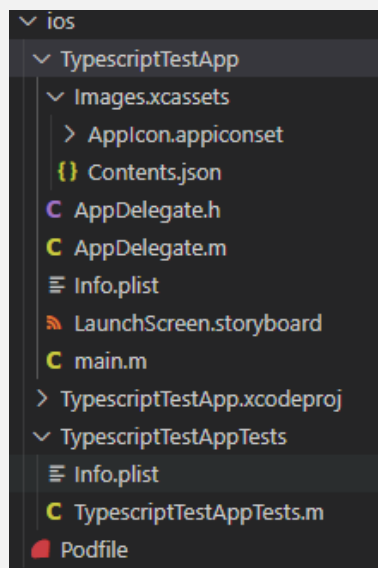
Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



React Native 프로젝트 구성



5. ios 폴더

ios 프로젝트가 담겨있는 폴더로, ios의 네이티브 모듈을 담당하고 있는 부분입니다.
ios 폴더는 android 폴더와 마찬가지로 ios app을 build 할때 사용 됩니다.

AppDelegate.h, AppDelegate.m

iOS 앱의 메인파일입니다.

Info.plist

iOS 프로젝트설정파일입니다.

Profile

ios에서 코코아포드라는 라이브러리에 대한 의존성 관리를 하는 파일입니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



React Native 프로젝트 구성

6. node_modules

JavaScript로 만들어진 React이기 때문에 node_modules 폴더에서 패키지를 관리하게 됩니다. 프로젝트와 관련된 모듈들이며, npm을 통해 설치한 모듈도 이곳에 위치하게 됩니다.

```
node_modules
├── .bin
├── @ampproject
├── @babel
├── @bcoe
├── @cnakazawa
├── @eslint
├── @hapi
├── @humanwhocodes
├── @istanbuljs
├── @jest
├── @jridgewell
├── @nodelib
└── @react-native
```




Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



변수선언

```
const name = 'Choi Gabkeun';  
const element = <h1>Hello, {name}</h1>;
```

- ✓ 위 예시처럼 name이라는 변수를 선언한 후 중괄호로 감싸 JSX 안에 사용하였다.
- ✓ 속성에 따옴표를 이용해 문자열 리터럴을 정의할 수 있다.

```
const element = <div tabIndex="0"></div>;
```

- ✓ 중괄호를 사용하여 어트리뷰트에 Javascript 표현식을 삽입할 수도 있다.

```
const element = <img src={user.avatarUrl}></img>;
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



변수선언

- ✓ 여러 자식요소를 사용할 경우 반드시 하나의 부모로 감싸야 한다.

```
export default function App() {  
  return (  
    <View style={styles.container}>  
      <Text>Open up App.js to start working on your app!</Text>  
      <StatusBar style="auto" />  
    </View>  
  );  
}
```

- ✓ 만약 <View> 부모 요소가 없다면 컴파일시 에러가 발생한다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

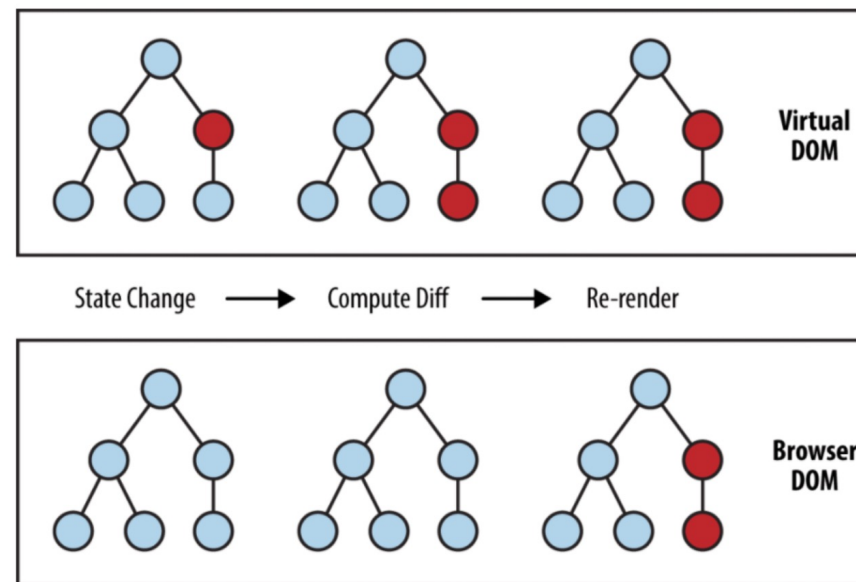
3rd Lecture

Mobile Computing



React Native DOM

- ✓ 리액트에서 가상 DOM은 “화면이 어떤 모습이어야 하는지 개발자가 작성한 내용과 실제 화면에 렌더링되는 것 사이에 존재하는 레이어”에 해당된다.
- ✓ 브라우저에서 상호작용하는 사용자 인터페이스를 렌더링하기 위해서 개발자는 반드시 브라우저의 DOM을 수정해야 한다. 하지만 과도한 DOM 수정은 심각한 성능 저하를 유발한다. 리액트는 페이지의 변화를 바로 렌더링하지 않고 먼저 메모리에 존재하는 가상 DOM에서 변화가 필요한 곳을 계산하고 필요한 최소한의 변경사항만 렌더링 한다.



브라우저 DOM의 렌더링을 최소화하는 가상 DOM의 계산 방법

state 변경 -> 차이점 계산 -> 다시 렌더링



Department of
Media Software

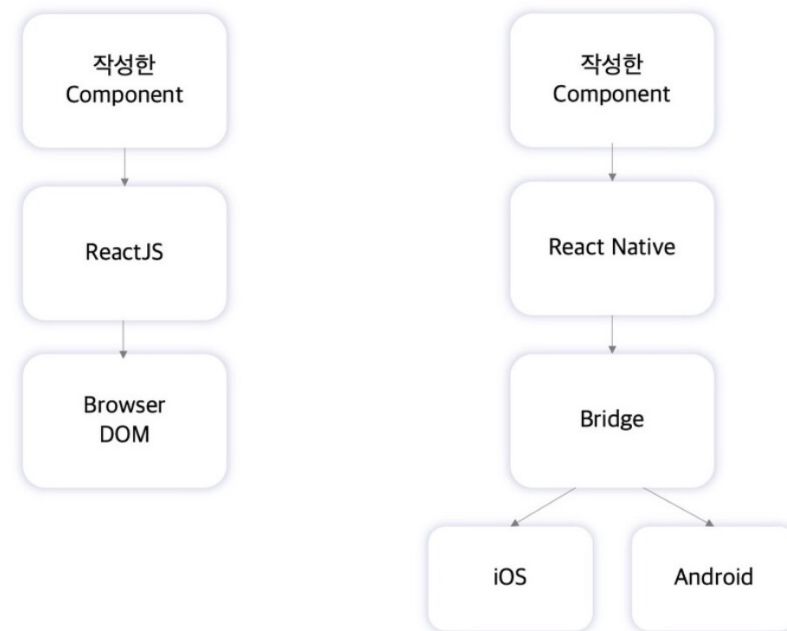
2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

React Native Bridge

- ✓ 리액트 네이티브는 브라우저 DOM으로 렌더링하는 대신 swift API를 호출하여 iOS컴포넌트로 렌더링하고, Java API를 호출하여 안드로이드 컴포넌트로 렌더링한다. 웹기반의 화면으로 최종 렌더링되는 대부분의 크로스 플랫폼 앱 개발 방법들과 구별되는 리액트 네이티브의 특징이다.





Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



JSX

- ✓ JSX(JavaScript XML)는 Javascript에 XML을 추가한 확장 문법이다.
- ✓ JSX는 리액트로 프로젝트를 개발할 때 사용되므로 공식적인 자바스크립트 문법은 아니다.
- ✓ 브라우저에서 실행하기 전에 바벨을 사용하여 일반 자바스크립트 형태의 코드로 변환된다.

```
import React from 'react';
import {Text, View} from 'react-native';

const HelloWorldApp = () => {
  return (
    <View
      style={{
        flex: 1,
        justifyContent: 'center',
        alignItems: 'center',
      }}>
      <Text>Hello, world!</Text>
    </View>
  );
};

export default HelloWorldApp;
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



JSX

- ✓ JSX는 하나의 파일에 자바스크립트와 HTML을 동시에 작성하여 편리하다.
- ✓ 자바스크립트에서 HTML을 작성하듯이 하기 때문에 가독성이 높고 작성하기 쉽다.
- ✓ **JSX(JavaScript Syntax Extension)**는 객체 생성과 함수 호출을 위한 문법적 편의를 제공하기 위해 만들어진 확장기능으로 리액트 프로젝트에서 사용된다.
가독성이 높고 작성하기 쉽다는 장점이 있으며 XML과 유사하다는 점에서 중첩된 nested 구조를 잘 나타낼 수 있다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component

- ✓ 리액트로 만들어진 앱을 이루는 최소한의 단위
- ✓ 기존의 웹 프레임워크는 MVC방식으로 분리하여 관리하여 각 요소의 의존성이 높아 재활용이 어렵다는 단점이 있었다. 반면 컴포넌트는 MVC의 뷰를 독립적으로 구성하여 재사용을 할 수 있고 이를 통해 새로운 컴포넌트를 쉽게 만들 수 있다.
- ✓ **컴포넌트는** 데이터(props)를 입력 받아 View(state) 상태에 따라 DOM Node를 출력하는 **함수**.
- ✓ 컴포넌트 이름은 항상 대문자로 시작하도록 한다.
(리액트는 소문자로 시작하는 컴포넌트를 DOM 태그로 취급하기 때문이다.)
- ✓ UI를 재사용 가능한 개별적인 여러 조각으로 나누고, 각 조각을 개별적으로 나누어 코딩한다.
- ✓ “props”라고 하는 임의의 입력을 받은 후, 화면에 어떻게 표시되는지를 기술하는 React 엘리먼트를 반환한다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component

- ✓ 리액트에서 정의하는 컴포넌트 종류는 크게 함수형 컴포넌트, 클래스형 컴포넌트 2가지가 있다.

1) 함수형 컴포넌트 (Stateless Functional Component)

가장 간단하게 컴포넌트를 정의하는 방법은 자바스크립트 함수를 작성하는 것이다. 여기서 import시 js, jsx 등 파일 확장자를 생략해도 자동으로 찾을 수 있다. 이는 "웹팩 코드 검색 확장자(webpack module resolution)" 기능 때문이다. 확장자가 파일 이름에 없는 경우 웹팩의 확장자 옵션(extentions)에 정의된 확장자 목록을 통해 확장자 이름을 포함한 파일이 있는지 확인하여 자동으로 импорт 한다.

```
import React from 'react';

function MyComponent(props) {
  return <div>Hello, {props.name}</div>;
}

export default MyComponent; //다른 JS파일에서 불러올 수 있도록 내보내 주기
```




Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component

2) 클래스형 컴포넌트 (Class Component)

- 컴포넌트 구성 요소, 리액트 생명주기를 모두 포함하고 있다.
- 프로퍼티, state, 생명주기 함수가 필요한 구조의 컴포넌트를 만들 때 사용한다.

```
import React from 'react';

class MyComponent extends React.Component {
  constructor(props) { // 생성함수
    super(props);
  }

  componentDidMount() { // 상속받은 생명주기 함수
  }

  render() { // 상속받은 화면 출력 함수, 클래스형 컴포넌트는 render() 필수
    return <div>Hello, {this.props.name}</div>;
  }
}

export default MyComponent; //다른 JS파일에서 불러올 수 있도록 내보내주기
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component

- ✓ MyButton.js 를 만들어 익스포트 하고 App.js 에서 불러 사용한다.

```
import React from "react";
import { TouchableOpacity, Text } from "react-native";

const MyButton = () => {
  return (
    <TouchableOpacity
      style={{
        backgroundColor: "#4B778D",
        padding: 16,
        margin: 10,
        borderRadius: 8,
      }}
      onPress={() => alert("Click!")}
    >
      <Text style={{ fontSize: 24, color: "white" }}>My Button</Text>
    </TouchableOpacity>
  );
};

export default MyButton;
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component

✓ MyButton.js 를 import 하여 동작시킨다.

```
import React from "react";
import { Text, View } from "react-native";
import MyButton from "./MyButton";

const App = () => {
  return (
    <View
      style={{
        flex: 1,
        backgroundColor: "#fff",
        alignItems: "center",
        justifyContent: "center",
      }}
    >
      <Text style={{ fontSize: 30, marginBottom: 10, textAlign: "center" }}>
        My Button Component
      </Text>
      <MyButton />
    </View>
  );
};

export default App;
```



Department of
Media Software

2023 1st semester
Mobile Computing

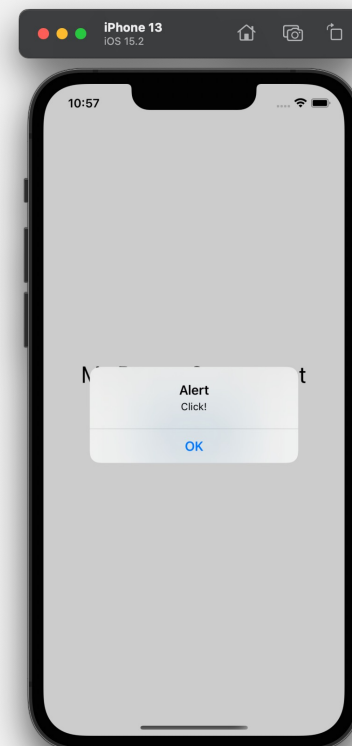
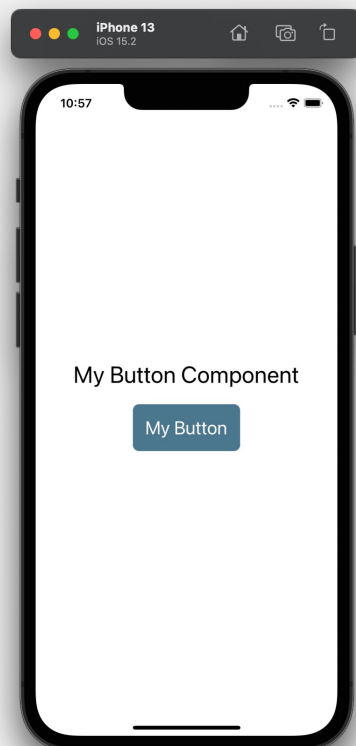
Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Component





Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Props(Properties)

- ✓ 상위 컴포넌트가 하위 컴포넌트에 값을 전달할 때 사용한다.(단방향 데이터 흐름 갖는다.)
- ✓ 프로퍼티는 수정할 수 없다는 특징이 있다.(자식입장에선 읽기 전용인 데이터이다.)
- ✓ 프로퍼티에 문자열을 전달할 때는 큰따옴표(" ")를, 문자열 외의 값을 전달할 때는 중괄호({ })를 사용 한다.
- ✓ 대부분의 구성요소는 생성 시 다른 매개변수를 사용하여 사용자 정의할 수 있다. 이러한 생성 매개변수를 props라고 한다.
- ✓ 자신의 구성 요소도 props를 사용할 수 있다. 이렇게 하면 각 위치에서 약간 다른 속성을 사용하여 앱의 여러 위치에서 사용되는 단일 구성 요소를 만들 수 있다. 기능 구성요소의 `props.YOUR_PROP_NAME` 또는 클래스 구성요소의 `this.props.YOUR_PROP_NAME`을 참조한다. 예를 들면 다음과 같다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Prop

```
import React from 'react';
import {Text, View, StyleSheet} from 'react-native';

const styles = StyleSheet.create({
  center: {
    alignItems: 'center',
  },
});

type GreetingProps = {
  name: string;
};

const Greeting = (props: GreetingProps) => {
  return (
    <View style={styles.center}>
      <Text>Hello {props.name}!</Text>
    </View>
  );
};

const LotsOfGreetings = () => {
  return (
    <View style={[styles.center, {top: 50}]}>
      <Greeting name="Rexxar" />
      <Greeting name="Jaina" />
      <Greeting name="Valeera" />
    </View>
  );
};

export default LotsOfGreetings;
```



Department of
Media Software

2023 1st semester
Mobile Computing

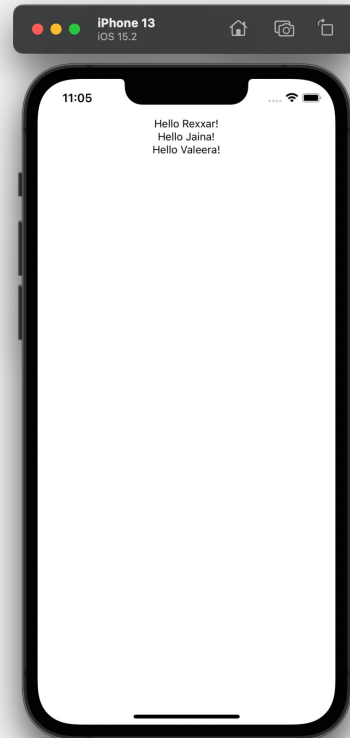
Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Props(Properties)





Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



Props(Properties)

- ✓ 이름을 Props로 사용하면 인사말 구성 요소를 사용자 정의할 수 있으므로 각 인사말에 해당 구성 요소를 재사용할 수 있습니다. 이 예제에서는 JSX의 인사말 구성 요소도 사용합니다. 이를 수행할 수 있는 힘이 React를 정말 멋지게 만드는 것입니다.
- ✓ 여기서 진행되는 또 다른 새로운 기능은 View 구성 요소입니다. 보기는 스타일과 레이아웃을 제어하는 데 도움이 되는 다른 구성 요소의 컨테이너로 유용합니다.
- ✓ Props와 기본 텍스트, 이미지 및 보기 구성 요소를 사용하여 다양한 정적 화면을 구축할 수 있습니다. 시간 경과에 따라 앱을 변경하는 방법을 배우려면 state에 대해 알아야 합니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



State

- ✓ 읽기 전용이고 수정해서는 안 되는 props와 달리 상태는 React 구성 요소가 사용자 작업, 네트워크 응답 및 기타 모든 것에 대한 응답으로 시간이 지남에 따라 출력을 변경할 수 있도록 합니다.
- ✓ React에서 state와 props의 차이점은 무엇인가요?
- ✓ React 구성 요소에서 props는 부모 구성 요소에서 자식 구성 요소로 전달하는 변수입니다. 마찬가지로 상태도 변수이지만 매개 변수로 전달되지 않고 구성 요소가 내부적으로 초기화하고 관리한다는 차이점이 있습니다.



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



State

```
import React, {Component} from 'react';
import {StyleSheet, TouchableOpacity, Text, View} from 'react-native';

class App extends Component {
  state = {
    count: 0,
  };

  onPress = () => {
    this.setState({
      count: this.state.count + 1,
    });
  };

  render() {
    return (
      <View style={styles.container}>
        <TouchableOpacity style={styles.button} onPress={this.onPress}>
          <Text>Click me</Text>
        </TouchableOpacity>
        <View>
          <Text>You clicked {this.state.count} times</Text>
        </View>
      </View>
    );
  }
}
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  button: {
    alignItems: 'center',
    backgroundColor: '#DDDDDD',
    padding: 10,
    marginBottom: 10,
  },
});

export default App;
```



Department of
Media Software

2023 1st semester
Mobile Computing

Prof.
Gabkeun Choi Ph.D.

3rd Lecture

Mobile Computing



State

