앱개발 2주차

02.

[앱개발 준비]리액트 네이티브&Expo소개

1)리액트 네이티브 = 리액트(React) + 네이티브(Native)

자바스크립트 언어 하나로 안드로이드 앱과 iOS 앱 두가지 모두 만들어 주는 라이브러리

|  |  |
| --- | --- |

Expo란?

(1)안드로이드&iOS코드를 정말 몰라도 개발이 가능(일부 기기제어기능 제외==React Native)

Expo Client 앱 설치

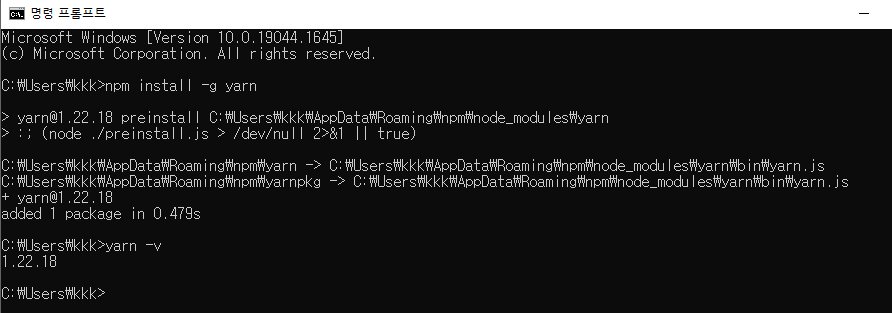
03. Node \* NPM 설치

NPM은 개발자들이 만들어놓은 Library임

Node.js로 Javacript 개발환경을 구축한다…

(2)Yarn 설치

* (NPM설치 후)
* CMD창 Open
* npm install -g yarn (엔터)
* tarn -v (설치 확인)



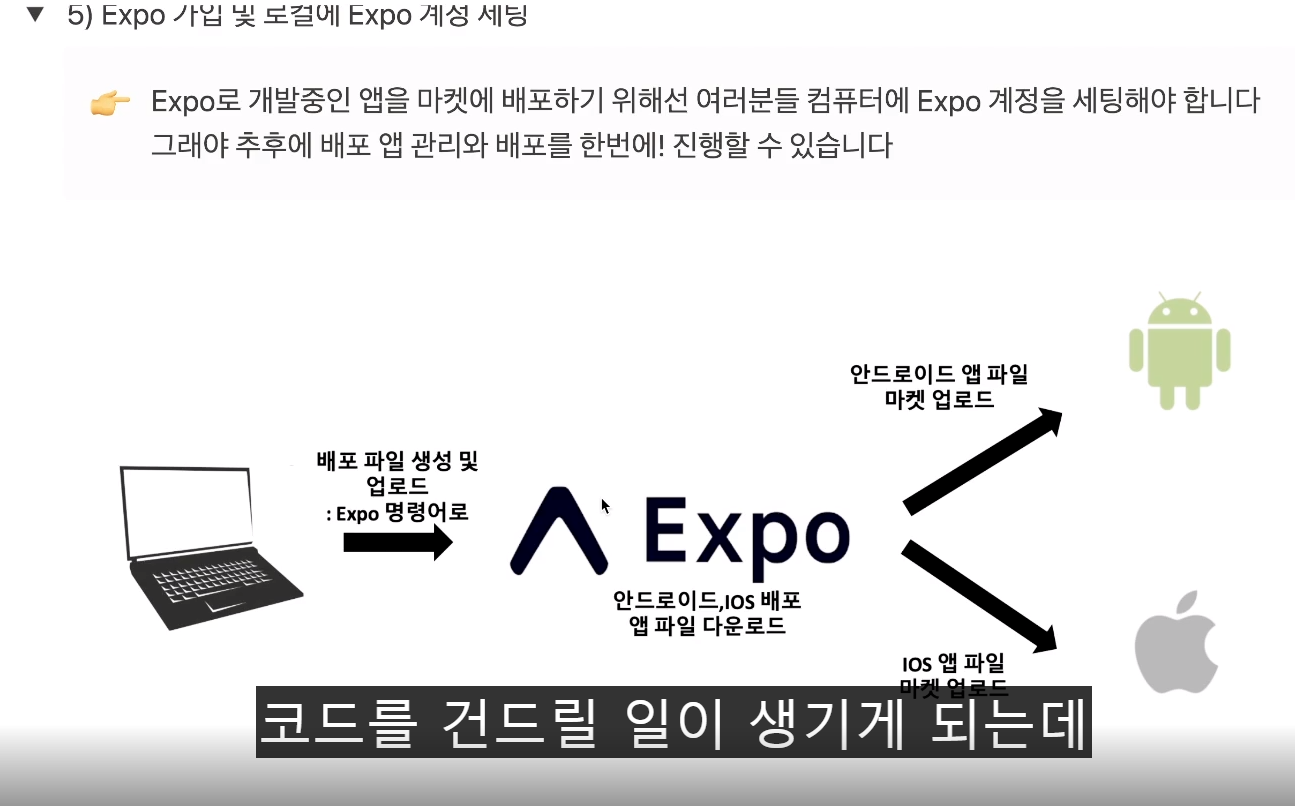
(3)Expo 설치

- npm install -g expo-cli(엔터)



* expo(엔터) (아래와 같은 명령어 도움말이 나타남)

| Usage: expo [command] [options]  Options:  -V, --version output the version number  --non-interactive Fail, if an interactive prompt would be required to continue.  -h, --help output usage information  Commands:  init [name] Create a new Expo project  start [path] Start a local dev server for the app  start:web [path] Start a Webpack dev server for the web app  export [path] Export the static files of the app for hosting it on a web server  install [packages...] Install a module or other package to a project  run:android [path] Run the Android app binary locally  run:ios [path] Run the iOS app binary locally  send [path] Share the project's URL to an email address  login Login to an Expo account  logout Logout of an Expo account  register Sign up for a new Expo account  whoami Return the currently authenticated account  client:install:ios Install Expo Go for iOS on the simulator  client:install:android Install Expo Go for Android on a connected device or emulator  config [path] Show the project config  doctor [path] Diagnose issues with the project  upgrade [sdk-version] Upgrade the project packages and config for the given SDK version  customize:web [path] Eject the default web files for customization  prebuild [path] Create native iOS and Android project files before building natively.  Learn more: https://docs.expo.dev/workflow/customizing/  publish [path] Deploy a project to Expo hosting  publish:set [path] Specify the channel to serve a published release  publish:rollback [path] Undo an update to a channel  publish:history [path] Log the project's releases  publish:details [path] Log details of a published release  build:web [path] Build the web app for production  credentials:manager [path] Manage your credentials  fetch:ios:certs [path] Download the project's iOS standalone app signing credentials  fetch:android:keystore [path] Download the project's Android keystore  fetch:android:hashes [path] Compute and log the project's Android key hashes  fetch:android:upload-cert [path] Download the project's Android keystore  push:android:upload [path] Upload an FCM key for Android push notifications  push:android:show [path] Log the value currently in use for FCM notifications for this project  push:android:clear [path] Delete a previously uploaded FCM credential  url [path] Log a URL for opening the project in Expo Go  url:ipa [path] Log the download URL for the standalone iOS binary  url:apk [path] Log the download URL for the standalone Android binary  webhooks [path] List all webhooks for a project  webhooks:add [path] Add a webhook to a project  webhooks:remove [path] Delete a webhook  webhooks:update [path] Update an existing webhook  build:ios [path] Superseded by eas build in eas-cli  build:android [path] Superseded by eas build in eas-cli  build:status [path] Superseded by eas build:list in eas-cli  eject [path] Superseded by expo prebuild  upload:android [path] Superseded by eas submit in eas-cli  upload:ios [path] Superseded by eas submit in eas-cli  client:ios [path] Superseded by Expo Dev Clients  Run a command with --help for more info �  $ expo start --help |
| --- |



(4)Expo 가입(expo.dev)

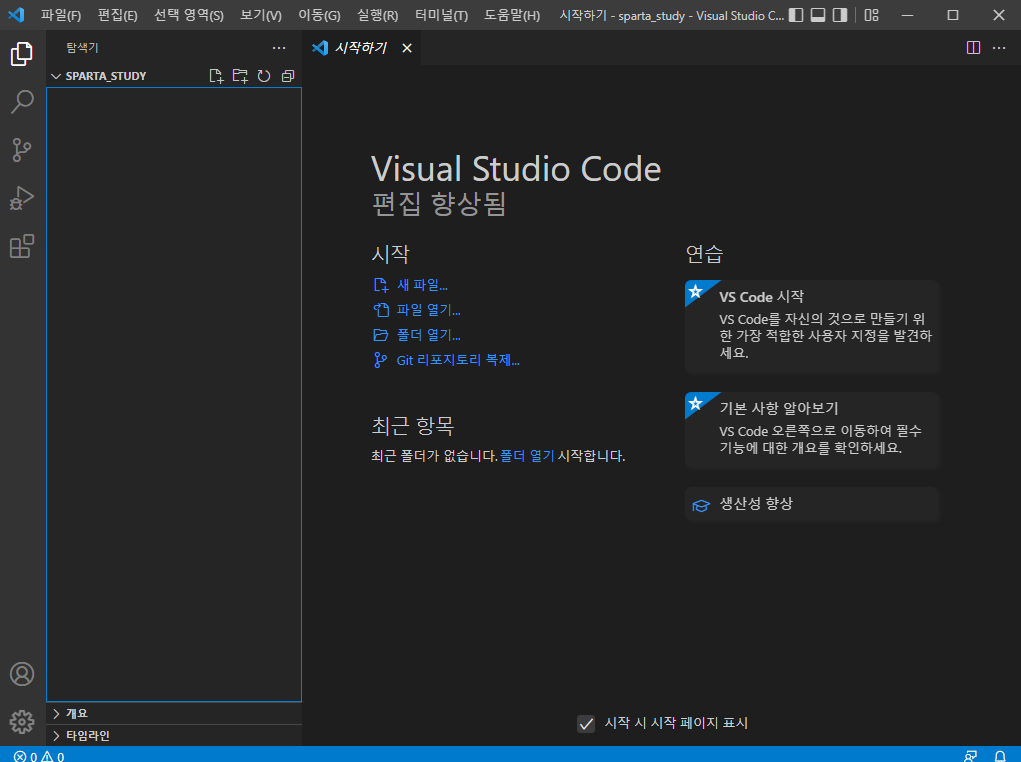
[doltee@naver.com](mailto:doltee@naver.com), Username: jasonkim2022,

(5)expo 계정 연결

* CMD 창 오픈
* expo login(엔터)
* username (jasonkim) / password 가입

(5)개발용 폴더 생성(바탕화면>sparta\_study

(6)VisualStudio Code 실행

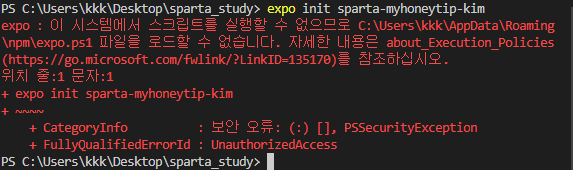


- 폴더 지정 (바탕화면>sparta\_study

- PowerShell 창 띄우기 (단축키: Ctrl+~ 물결)

>(expo 프로젝트 생성)  
> CMD>expo init sparta-myhoneytip-kim (엔터)

(1차 에러발생) ⇒ FAQ 확인, 조치완료



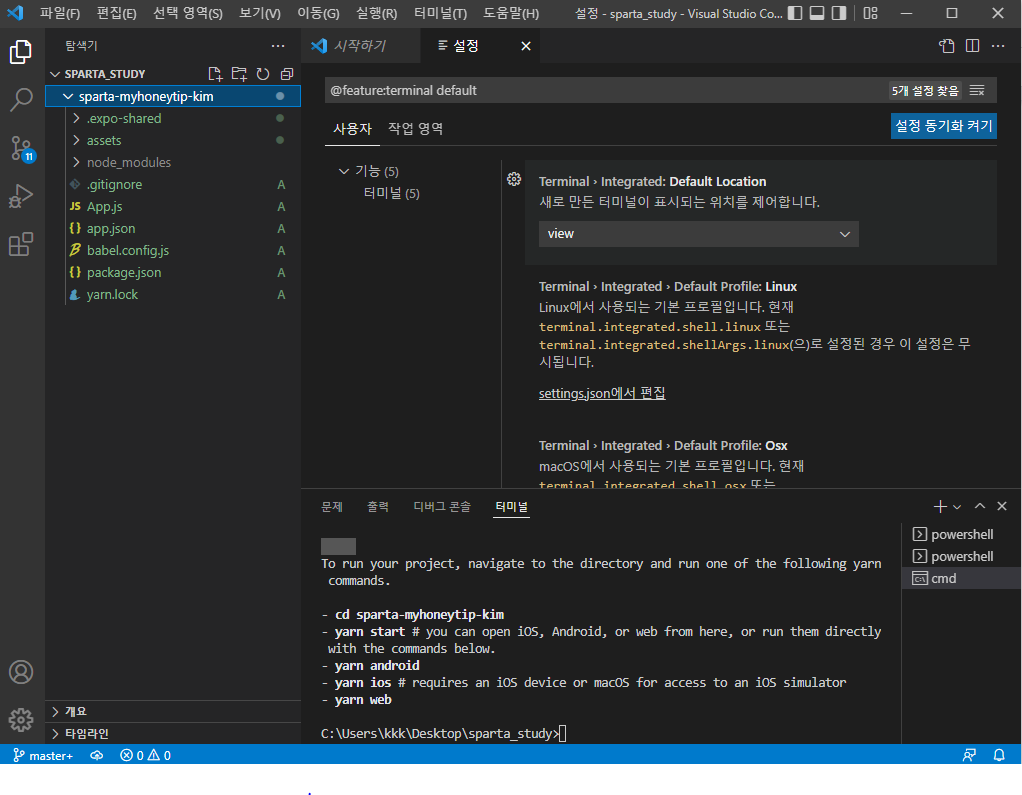
>> 조치방법 <https://s3.us-west-2.amazonaws.com/secure.notion-static.com/7f233dd5-7920-4554-a435-705871419809/change_default_shell.mp4?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Content-Sha256=UNSIGNED-PAYLOAD&X-Amz-Credential=AKIAT73L2G45EIPT3X45%2F20220505%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20220505T064839Z&X-Amz-Expires=86400&X-Amz-Signature=7ec62132a91d32d6fc21a877a446290bd8a642df4836d9c3f09817a928990016&X-Amz-SignedHeaders=host&x-id=GetObject>

- (재실행)

- CMD>expo init sparta-myhoneytip-kim (엔터)

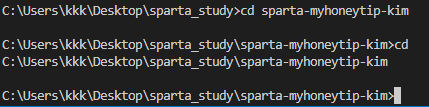
- option은 blank 선택

- (expo 프로젝트 생성작업 진행)



* 좌측에 “sparta-myhoneytip-kim” 폴더가 생긴 것을 확인

\*\* CD sparta-myhoneytip-kim (엔터) ⇒ 해당 폴더로 Change Directory함

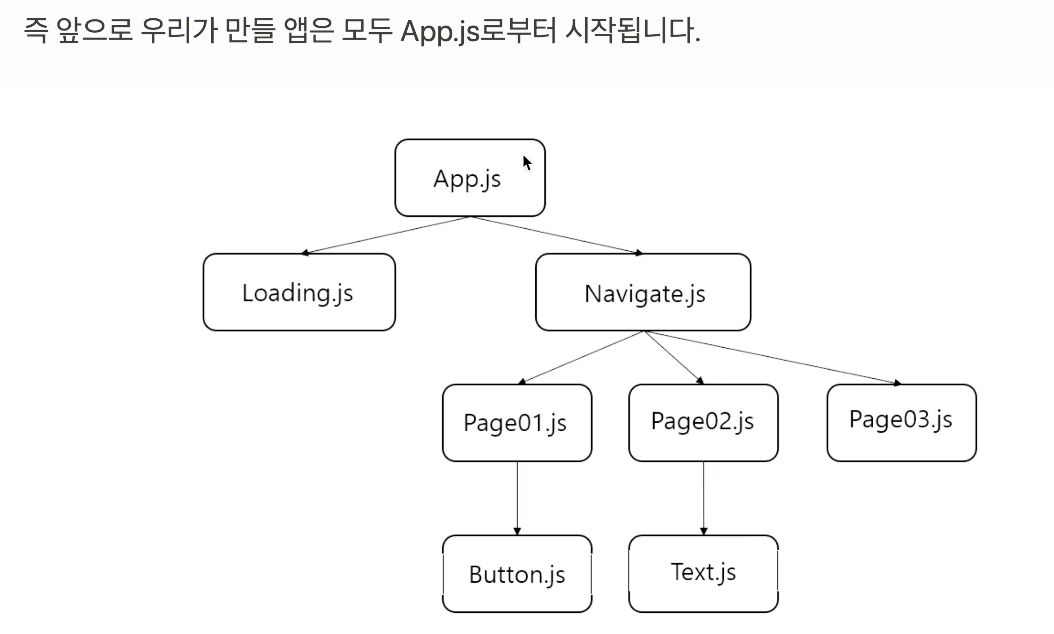


**(Expo 실행)**

- expo start (엔터)

- QR코드 갖다대기(구동)

- 폴더 구조 해설



* assets: 이미지 폴더
* node\_modules: npm 소스프로그램들
* App.js: 시작프로그램(앱)
* App.JSON: 환경설정 파일

**2-5. [앱화면 만들기] JSX문법**

| import { StatusBar } from 'expo-status-bar';  import React from 'react';  import { StyleSheet, Text, View, LogBox } from 'react-native';  export default function App() {  LogBox.ignoreLogs(['Warning: ...']);  return (  <View style={styles.container}>  <Text style={styles.textStyle}>스파르타 진수왔나?</Text>  <StatusBar style="dark" />  </View>  );  }  const styles = StyleSheet.create({  container: {  flex: 1,  backgroundColor: '#fff',  alignItems: 'center',  justifyContent: 'center',  },  textStyle: {  color:'green'  }  }); |
| --- |

**06. [앱화면만들기]View, Text, ScrollView**

11)<View>

12)Text

13) ScrollView

**2-8 [앱 화면만들기]Button, TouchableOpacity, Image**

함수 밖에서 버튼 onPress 함수처리 구현

export default function App() {

const customAlert = () => {

Alert.alert('JSX 밖에서 함수 구현 가능!!')

}

return (

<SafeAreaView style={styles.container}>

<View style={styles.textContainer}>

<Text style={styles.textStyle}>아래 버튼을 눌러주세요</Text>

{/\* 버튼 onPress 속성에 일반 함수를 연결 할 수 있습니다. \*/}

<Button

style={styles.buttonStyle}

title="버튼입니다!! "

color="#f194ff"

onPress={customAlert}

/>

{/\* ES6 문법으로 배웠던 화살표 함수로 연결 할 수도 있습니다. \*/}

<Button

style={styles.buttonStyle}

title="버튼입니다 "

color="#FF0000"

onPress={customAlert}

/>

</View>

</SafeAreaView>

);

}

Button 대신 TouchableOpacity를 쓸 거다(Button보다 이동하기, Style적용이 쉽다)

**2-9.[앱화면 꾸미기]구성한 화면 꾸미기, Styles**

* **자주 사용하는 스타일 속성**

| import React from 'react';  import { StyleSheet, Text, View, Image } from 'react-native';  export default function App() {  return (  <View style={styles.container}>  <View style={styles.textContainer}>  <Text style={styles.textStyle}>스파르타 코딩클럽!!</Text>  </View>  </View>  );  }  const styles = StyleSheet.create({  container: {  //영역을 잡는 속성입니다. 따로 자세히 다룹니다.  //flex: 1은 **전체 화면**을 가져간다는 뜻입니다  **flex**: 1,  //**영역의 배경 색**을 결정합니다  **backgroundColor**: '#fff',  //아래 두 속성은 영역 안의 컨텐츠들의 배치를 결정합니다.  //flex를 자세히 다룰때 같이 자세히 다룹니다  **justifyContent**:"center",  **alignContent**:"center"  },  textContainer: {  //영역의 바깥 공간 이격을 뜻합니다(하단 이미지 참조)  **margin**:10,  //영역 안의 컨텐츠 이격 공간을 뜻합니다(하단 이미지 참조)  padding: 10,  //테두리의 구부러짐을 결정합니다. 지금 보면 조금 둥글죠?  borderRadius:10,  //테두리의 두께를 결정합니다  borderWidth:2,  //테두리 색을 결정합니다  borderColor:"#000",  //테두리 스타일을 결정합니다. 실선은 solid 입니다  borderStyle:"dotted",  },  textStyle: {  //글자 색을 결정합니다. rgb, 값 이름, 색상코드 모두 가능합니다  color:"red",  //글자의 크기를 결정합니다  fontSize:20,  //글자의 두께를 결정합니다  fontWeight:"700",  //가로기준으로 글자의 위치를 결정합니다  textAlign:"center"  }  }); |
| --- |

**2-10. [앱화면 만들기[콘텐츠의 위치-Flex**

**2-11. [앱 화면 만들기]메인화면 꾸미기**

| import React from 'react';  import { StyleSheet, Text, View, Image, TouchableOpacity, ScrollView} from 'react-native';    const main = 'https://firebasestorage.googleapis.com/v0/b/sparta-image.appspot.com/o/lecture%2Fmain.png?alt=media&token=8e5eb78d-19ee-4359-9209-347d125b322c'  export default function App() {  return (  <ScrollView style={styles.container}>  {/\* 페이지 타이틀 처리, 폰트크기:20 굵기:700 상단여백:50 좌측여백:20 \*/}  <Text style={styles.title}>나만의 꿀팁</Text>  {/\* 상단 메인 이미지 처리, 크기:90%, 높이:200, 상단여백: 20, 가운데정렬 \*/}  <Image style={styles.mainImage} source={{uri:main}}/>  {/\* 좌우스크롤버튼 영역 처리, 버튼을 펼쳐 가로 스크롤링:horizontal \*/}  <ScrollView style={styles.middleContainer} horizontal indicatorStyle={"white"}>  {/\* 버튼 추가, 바깥여백:7, 안여백:15, 라운드처리:15, 배경색 각각 다르게 \*/}  <TouchableOpacity style={styles.middleButton01}><Text style={styles.middleButtonText}>생활</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton02}><Text style={styles.middleButtonText}>재테크</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton03}><Text style={styles.middleButtonText}>반려견</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton04}><Text style={styles.middleButtonText}>꿀팁 찜</Text></TouchableOpacity>  </ScrollView>  {/\* 하단 카드 영역 View 처리 좌(카드):우(설명)=1:2 \*/}  <View style={styles.cardContainer}>  <View style={styles.card}>  <Image style={styles.cardImage} source={{uri:"https://firebasestorage.googleapis.com/v0/b/sparta-image.appspot.com/o/lecture%2Fpizza.png?alt=media&token=1a099927-d818-45d4-b48a-7906fd0d2ad3"}}/>  {/\* 카드설명 타이틀 처리, font:20 \*/}  <View style={styles.cardText}>  <Text style={styles.cardTitle}>먹다 남은 피자를 촉촉하게!</Text>  {/\* 3줄 자르기 ... ==> numberOfLines={3}, 폰트:15 \*/}  <Text style={styles.cardDesc} numberOfLines={3}>먹다 남은 피자는 수분이 날라가기 때문에 처음처럼 맛있게 먹을 수 없는데요. 이럴 경우 그릇에 물을 받아 전자레인지 안에서 1분 30초에서 2분 정도 함께 돌려주면 촉촉하게 먹을 수 있습니다. 물이 전자레인지 안에서 수증기를 일으키고, 피자에 촉촉함을 더해줍니다.</Text>  <Text style={styles.cardDate}>2020.09.09</Text>  </View>  </View>  </View>  </ScrollView>  );  }  const styles = StyleSheet.create({  container: {  flex:1  },  //페이지 타이틀 Style  title: {  fontSize: 20,  fontWeight: '700',  marginTop:50,  marginLeft:20  },  //상단이미지 Style  //이미지는 width, height 미지정시 화면에 안나타남. 반드시 지정해야 한다!  mainImage: {  width:'90%',  height:200,  borderRadius:10,  marginTop:20,  //중앙 정렬  alignSelf:"center"  },  //버튼영역 Style  middleContainer:{  marginTop:20,  marginLeft:10,  height:60  },  //버튼 Style  middleButton01: {  width:100,  height:50,  padding:15,  backgroundColor:"#fdc453",  borderColor:"deeppink",  borderRadius:15,  margin:7  },  //버튼 Style  middleButton02: {  width:100,  height:50,  padding:15,  backgroundColor:"#fe8d6f",  borderRadius:15,  margin:7  },  //버튼 Style  middleButton03: {  width:100,  height:50,  padding:15,  backgroundColor:"#9adbc5",  borderRadius:15,  margin:7  },  //버튼 Style  middleButton04: {  width:100,  height:50,  padding:15,  backgroundColor:"#f886a8",  borderRadius:15,  margin:7  },  //버튼 Style  middleButtonText: {  color:"#fff",  fontWeight:"700",  textAlign:"center"  },  //카드영역 Style  cardContainer: {  marginTop:10,  marginLeft:10  },  //카드 Style  //카드 하단 구분선 처리 borderBottom  card:{  flex:1,  flexDirection:"row",  margin:10,  borderBottomWidth:0.5,  borderBottomColor:"#eee",  paddingBottom:10  },  //카드이미지 Style  cardImage: {  flex:1,  width:100,  height:100,  borderRadius:10,  },  //카드타이틀 Style  cardTitle: {  fontSize:20,  fontWeight:"700"  },  //카드텍스트 Style  //텍스트는 위에서 아래로  cardText: {  flex:2,  //텍스트는 위에서 아래로  flexDirection:"column",  marginLeft:10,  },  //카드설명 Style  cardDesc: {  fontSize:15  },  //카드날짜 Style  cardDate: {  fontSize:10,  color:"#A6A6A6",  }  }); |
| --- |

결과 화면

|  |
| --- |

**2-13 모듈과 반복문**

* **앞 화면의 하단 글들을 데이터로 불러와서 처리해 볼까?**
* **데이터 처리를 위해 별도 파일에 저장 후 import처리한다**
* **View 안에서 Javascript를 사용하려면 { 중괄호({)로 묶어 줘야 한다**

data.json에 데이터를 저장하고, app.js에서 호출하여 Display

<data.json>

| {  "tip":[  {  "idx":0,  "category":"생활",  "title":"먹다 남은 피자를 촉촉하게!",  "image":"https://storage.googleapis.com/sparta-image.appspot.com/lecture/pizza.png",  "desc":"먹다 남은 피자는 수분이 날라가기 때문에 처음처럼 맛있게 먹을 수 없는데요. 이럴 경우 그릇에 물을 받아 전자레인지 안에서 1분 30초에서 2분 정도 함께 돌려주면 촉촉하게 먹을 수 있습니다. 물이 전자레인지 안에서 수증기를 일으키고, 피자에 촉촉함을 더해줍니다.",  "date":"2020.09.09"  },  {  "idx":1,  "category":"생활",  "title":"바나나를 싱싱하게 보관하기",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/banana.png",  "desc":"바나나에 날파리가 꼬이거나 금방 익어버리는 것을 예방하기 위한 방법인데요. 바나나 양쪽 끝을 자른 후, 보관용 케이스나 비닐봉지에 묶어 밀봉합니다. 그리고 냉장고에 넣어주면 되는데요. 하루에 1~2개씩 꺼내서 싱싱하게 먹을 수 있습니다.",  "date":"2020.09.09"  },  {  "idx":2,  "category":"생활",  "title":"셔츠에 묻은 볼펜 자국 없애기",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/shirt.png",  "desc":"셔츠를 자주 입는 사람의 경우, 종종 볼펜 자국이 묻기 마련인데요. 이럴 경우에는 집에 있는 물파스로 가볍게 지울 수 있습니다. 옷 뒷부분을 키친타올로 받쳐 번지는 것을 방지한 후, 볼펜 자국 있는 부분을 물파스로 눌러주고, 키친타올로 닦아냅니다.",  "date":"2020.09.09"  },  {  "idx":3,  "category":"재테크",  "title":"잠자는 내 돈을 찾아라",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/money1.png",  "desc":"‘새는 돈’에는 미처 몰랐던 카드 포인트, 휴면예금이나 환급금도 포함됩니다. 확실히 파악하지 못한 잠자는 돈을 찾아보고 자투리 돈들을 모으는 것도 중요합니다. 케이블방송, 위성방송 서비스를 이용하면서 중복 납부한 요금, 셋톱박스 보증금 등 돌려받지 않은 돈이 있는지 확인 해보세요. 또, 카드 포인트 통합 조회 서비스를 이용해 여러 개의 카드 포인트가 모두 얼마인지 체크해두는 것이 좋습니다. 보험해약 환급금, 휴면 보험금이나 휴면 예금을 찾아보고 돌려받는 일도 요즘에는 어렵지 않습니다.",  "date":"2020.09.09"  },  {  "idx":4,  "category":"재테크",  "title":"할인행사, 한정할인판매 문구의 함정 탈출!",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/money2.png",  "desc":"‘안 사면 100% 할인’이라는 말 들어보셨나요? 견물생심, 좋은 물건을 보면 사고 싶기 마련입니다. 특히 대대적인 ‘할인 행사’ 중인 대형 마트에 갔을 때는 말할 것도 없겠죠. 따라서 생필품을 살 때, 한꺼번에 사서 사용하는 것보다 필요할 때 조금씩 구매하는 편이 좋습니다. 장을 보면서 대형마트에 자주 가다 보면 지금 필요한 것뿐 아니라 앞으로 필요할 것까지 사게 되어 지출이 커지기 때문입니다. 특히 할인 품목을 보면 뜻하지 않은 소비를 하는 경우도 많아진다. 홈쇼핑, 대형마트 등의 ‘할인행사’, ‘한정할인판매’ 등의 문구를 조심하세요. ",  "date":"2020.09.09"  },  {  "idx":5,  "category":"생활",  "title":"방전된 건전지 살리기",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/battery.png",  "desc":"건전지를 다 사용하지 않아도 방전되면, 버리는 경우가 종종 있는데요. 건전지의 무게감이 느껴진다면, 드라이기를 활용해 방전된 건전지를 깨울 수 있습니다. 드라이기 열기를 10초~30초 정도 골고루 가해주면 되는데요. 건전지가 불필요하게 낭비되는 것을 막을 수 있습니다.",  "date":"2020.09.09"  },  {  "idx":6,  "category":"반려견",  "title":"반려견에게 배변 교육 시킬 때",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/puppy.png",  "desc":"우선, 배변패드를 순서대로 돌며 간식을 조금씩 떨어뜨려 놓는다. 2단계는 배변패드 앞에서 기다렸다 반려견이 스스로 올라오면 간식을 주어서 보상하고, 3단계는 “화장실 가자”나 “매트” 같은 명령어를 붙여 말한 뒤 배변패드에 올라오면 간식을 주는 것이다. 마지막 단계는 배변패드에 올라간 반려견이 대소변을 본 다음 간식을 줌으로써 이 장소가 즐거운 곳이라는 인식을 심어주는 것이다. 그리고 무엇보다 1, 2회 사용한 배변패드는 바로 갈아줘야 한다.",  "date":"2020.09.09"  },  {  "idx":7,  "category":"반려견",  "title":"반려견이 주인과 떨어지는 것을 무서워 할 때",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/puppy2.png",  "desc":"분리불안교육은 반려견에게 혼자 남는 법을 알려주기 위한 것이 아니라, 보호자가 ‘언제나 너에게 돌아올 거야’라고 알려주는 교육이다. 반려견과 5초 동안 떨어져 있다가 다시 문을 열고 들어가 손 냄새를 맡게 해주는 훈련을 하루 10번씩 7일 동안 반복하는 ‘5,10,7 법칙’을 통해 반려견의 마음을 편안하게 해줄 수 있다.",  "date":"2020.09.09"  },  {  "idx":8,  "category":"반려견",  "title":"반려견을 아이와 함께 키울 때",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/puppy3.png",  "desc":"‘인간의 행복’을 위해 반려동물을 키우는 것에 대해 꾸준히 비판과 우려를 제기해온 그는 특히 ‘아이들의 정서’를 위해 반려견을 키우려 한다는 부모들에게 당부한다. “반려동물을 통해 아이들의 정서가 좋아진다면, 그것은 부모가 나와 생김새와 느낌, 말과 행동이 다른 동물을 아끼는 모습을 보기 때문입니다.” 인간의 뜻에 의해 인간과 함께 살게 된 생명을 좀 더 이해하고 행복하게 살 수 있도록 하는 것은 역시 인간의 노력에 달려 있다.",  "date":"2020.09.09"  },  {  "idx":9,  "category":"재테크",  "title":"렌탈 서비스 금액 비교해보기",  "image": "https://storage.googleapis.com/sparta-image.appspot.com/lecture/money2.png",  "desc":"요즘은 정수기, 공기 청정기, 자동차나 장난감 등 다양한 대여서비스가 활발합니다. 사는 것보다 경제적이라고 생각해 렌탈 서비스를 이용하는 분들이 늘어나고 있는데요. 다만, 이런 렌탈 서비스 이용이 하나둘 늘어나다 보면 그 금액은 겉잡을 수 없이 불어나게 됩니다. 특히, 렌탈 서비스는 빌려주는 물건의 관리비용까지 포함된 것이기에 생각만큼 저렴하지 않습니다. 직접 관리하며 사용할 수 있는 물건이 있는지 살펴보고, 렌탈 서비스 항목에서 제외해보세요. 렌탈 비용과 구매 비용, 관리 비용을 여러모로 비교해보고 고민해보는 것이 좋습니다. ",  "date":"2020.09.09"  }  ]  } |
| --- |

<app.js>

| import React from 'react';  import { StyleSheet, Text, View, Image, TouchableOpacity, ScrollView} from 'react-native';  const main = 'https://firebasestorage.googleapis.com/v0/b/sparta-image.appspot.com/o/lecture%2Fmain.png?alt=media&token=8e5eb78d-19ee-4359-9209-347d125b322c'  // data.json 데이터를 가져옴  import data from './data.json';  export default function App() {  let tip = data.tip;  let todayWeather = 10 + 17;  let todayCondition = "흐림"  //return 구문 밖에서는 슬래시 두개 방식으로 주석  return (  /\*  return 구문 안에서는 {슬래시 + \* 방식으로 주석  \*/  <ScrollView style={styles.container}>  <Text style={styles.title}>나만의 꿀팁</Text>  <Text style={styles.weather}>오늘의 날씨: {todayWeather + '°C ' + todayCondition} </Text>  <Image style={styles.mainImage} source={{uri:main}}/>  <ScrollView style={styles.middleContainer} horizontal indicatorStyle={"white"}>  <TouchableOpacity style={styles.middleButton01}><Text style={styles.middleButtonText}>생활</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton02}><Text style={styles.middleButtonText}>재테크</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton03}><Text style={styles.middleButtonText}>반려견</Text></TouchableOpacity>  <TouchableOpacity style={styles.middleButton04}><Text style={styles.middleButtonText}>꿀팁 찜</Text></TouchableOpacity>  </ScrollView>  <View style={styles.cardContainer}>  {/\* 하나의 카드 영역을 나타내는 View \*/}  { /\* View 안에서 Javasctipt 함수를 쓰기 위해서는 반드시 { 중괄호로 묶어 주어야 한다 \*/  tip.map((content,i)=>{  /\* key={i} 맵 반복 처리시 View에 키값을 부여햐여야 함에 주의! \*/  return (<View style={styles.card} key={i}>  <Image style={styles.cardImage} source={{uri:content.image}}/>  <View style={styles.cardText}>  <Text style={styles.cardTitle} numberOfLines={1}>{content.title}</Text>  <Text style={styles.cardDesc} numberOfLines={3}>{content.desc}</Text>  <Text style={styles.cardDate}>{content.date}</Text>  </View>  </View>)  })  }    </View>    </ScrollView>  );  }  const styles = StyleSheet.create({  container: {  //앱의 배경 색  backgroundColor: '#fff',  },  title: {  //폰트 사이즈  fontSize: 20,  //폰트 두께  fontWeight: '700',  //위 공간으로 부터 이격  marginTop:50,  //왼쪽 공간으로 부터 이격  marginLeft:20  },  weather:{  alignSelf:"flex-end",  paddingRight:20  },  mainImage: {  //컨텐츠의 넓이 값  width:'90%',  //컨텐츠의 높이 값  height:200,  //컨텐츠의 모서리 구부리기  borderRadius:10,  marginTop:20,  //컨텐츠 자체가 앱에서 어떤 곳에 위치시킬지 결정(정렬기능)  //각 속성의 값들은 공식문서에 고대로~ 나와 있음  alignSelf:"center"  },  middleContainer:{  marginTop:20,  marginLeft:10,  height:60  },  middleButton01: {  width:100,  height:50,  padding:15,  backgroundColor:"#fdc453",  borderColor:"deeppink",  borderRadius:15,  margin:7  },  middleButton02: {  width:100,  height:50,  padding:15,  backgroundColor:"#fe8d6f",  borderRadius:15,  margin:7  },  middleButton03: {  width:100,  height:50,  padding:15,  backgroundColor:"#9adbc5",  borderRadius:15,  margin:7  },  middleButtonText: {  color:"#fff",  fontWeight:"700",  //텍스트의 현재 위치에서의 정렬  textAlign:"center"  },  middleButton04: {  width:100,  height:50,  padding:15,  backgroundColor:"#f886a8",  borderRadius:15,  margin:7  },  cardContainer: {  marginTop:10,  marginLeft:10  },  card:{  flex:1,  //컨텐츠들을 가로로 나열  //세로로 나열은 column <- 디폴트 값임  flexDirection:"row",  margin:10,  borderBottomWidth:0.5,  borderBottomColor:"#eee",  paddingBottom:10  },  cardImage: {  flex:1,  width:100,  height:100,  borderRadius:10,  },  cardText: {  flex:2,  flexDirection:"column",  marginLeft:10,  },  cardTitle: {  fontSize:20,  fontWeight:"700"  },  cardDesc: {  fontSize:15  },  cardDate: {  fontSize:10,  color:"#A6A6A6",  },  }); |
| --- |

**2-14 [앱&자바스크립트] {} 표현식과 조건문**

* **상황연산자(ex: let result = 10 !== 2 ? true : false)**
* return i % 2 == 0 ? (<View style={styles.cardEven} key={i}>

| <View style={styles.cardContainer}>  {/\* 하나의 카드 영역을 나타내는 View \*/}  {  tip.map((content,i)=>{  return i % 2 == 0 ? (<View style={styles.cardEven} key={i}>  <Image style={styles.cardImage} source={{uri:content.image}}/>  <View style={styles.cardText}>  <Text style={styles.cardTitle} numberOfLines={1}>{content.title}</Text>  <Text style={styles.cardDesc} numberOfLines={3}>{content.desc}</Text>  <Text style={styles.cardDate}>{content.date}</Text>  </View>  </View>) : (<View style={styles.cardOdd} key={i}>  <Image style={styles.cardImage} source={{uri:content.image}}/>  <View style={styles.cardText}>  <Text style={styles.cardTitle} numberOfLines={1}>{content.title}</Text>  <Text style={styles.cardDesc} numberOfLines={3}>{content.desc}</Text>  <Text style={styles.cardDate}>{content.date}</Text>  </View>  </View>)    })  }    </View> |
| --- |