**프로젝트 진행 보고서(1주차)**

**\* 프로젝트를 위해서 진행한 내용을 간단하게 정리**

1. 기능 구현 목록

|  |
| --- |
| - 메인 페이지 구현  - 현재 위치의 경도, 위도 값 찾기  - OpenWeather API 연동  - API에서 호출한 JSON 값 화면 출력  - 예외 처리 페이지 구현  - 로딩 화면 구현 |

⦁main.dart

|  |
| --- |
| import 'package:flutter/material.dart';  import 'package:meonji/screens/loading.dart';  void main() {    runApp(MyApp());  }  class MyApp extends StatelessWidget {    const MyApp({ Key? key }) : super(key: key);    @override    Widget build(BuildContext context) {      return MaterialApp(        title: '먼지어때?',        theme: ThemeData(          primaryColor: Colors.black,        ),        home: Loading(),      );    }  } |

⦁data/my\_location.dart

|  |
| --- |
| import 'package:geolocator/geolocator.dart';  class Mylocation {    late double latitude2;    late double longitude2;    Future<void> getMyCurrentLocation() async {      try {        Position position = await Geolocator.getCurrentPosition(            // 위치 정보를 position에 삽입            desiredAccuracy: LocationAccuracy.high); // 위치 정확도        latitude2 = position.latitude;        longitude2 = position.longitude;        print(latitude2);        print(longitude2);      } catch (e) {        print('인터넷 연결에 문제가 생겼습니다!');      }    }  } |

⦁data/network.dart

|  |
| --- |
| import 'package:http/http.dart' as http;  import 'dart:convert';  class Network {    final String url;    Network(this.url);    Future<dynamic> getJsonData() async {      // uri 파싱      http.Response response = await http.get(Uri.parse(url));      // api 호출 성공 시      if (response.statusCode == 200) {        String jsonData = response.body; // 바디부분 추출        var parsingData = jsonDecode(jsonData);        return parsingData;      }    }  } |

⦁screens/weather\_screen.dart

|  |
| --- |
| import 'package:flutter/material.dart';  // 빌더를 전달받음  class WeatherScreen extends StatefulWidget {    WeatherScreen({this.parseWeatherData}); // 생성자 선언    final dynamic parseWeatherData;    @override    \_WeatherScreenState createState() => \_WeatherScreenState();  }  // 날씨를 텍스트로 표시하는 부분  class \_WeatherScreenState extends State<WeatherScreen> {    late String cityName; // 도시명    late int temp; // 온도    @override    void initState() {      // 부모 StatefulWidget(WeatherScreen)에서 parseWeatherData를 받아서 쓸 수 있음!      super.initState();      updateData(widget.parseWeatherData);    }    // 날씨 업데이트    void updateData(dynamic weatherData) {      double temp2 = weatherData['main']['temp'];      temp = temp2.round(); // 반올림      cityName = weatherData['name'];      print(temp);      print(cityName);    }    @override    Widget build(BuildContext context) {      return Scaffold(        body: SafeArea(          child: Center(            child: Column(              mainAxisAlignment: MainAxisAlignment.center,              children: [                Text(                  '$cityName',                  style: TextStyle(fontSize: 30.0),                ),                SizedBox(                  height: 20.0,                ),                Text(                  '$temp',                  style: TextStyle(fontSize: 30.0),                ),              ],            ),          ),        ),      );    }  } |

⦁Screens/loading.dart

|  |
| --- |
| import 'package:flutter/material.dart';  import 'package:meonji/data/my\_location.dart'; // 위치정보관련  import 'package:meonji/data/network.dart'; // 날씨데이터 관련  import 'package:meonji/api/key.dart';  import 'package:meonji/screens/weather\_screen.dart';  class Loading extends StatefulWidget {    const Loading({Key? key}) : super(key: key);    @override    State<Loading> createState() => \_LoadingState();  }  class \_LoadingState extends State<Loading> {    late double latitude3; // 위도 변수 선언    late double longitude3; // 경도 변수 선언    @override    void initState() {      super.initState(); // 상태초기화 1회만 실행      getLocation(); // 위치 정보 가져오기    }    void getLocation() async {      Mylocation myLocation = Mylocation();      await myLocation.getMyCurrentLocation();      latitude3 = myLocation.latitude2; // 위도 변수 삽입      longitude3 = myLocation.longitude2; // 경도 변수 삽입      print(latitude3);      print(longitude3);      Network network = Network(          'https://api.openweathermap.org/data/2.5/weather?lat=$latitude3&lon=$longitude3&appid=$apiKey&units=metric');      var weatherData = await network.getJsonData(); // api값을 그릇에 담기      print(weatherData);      // weather\_screen으로 빌더를 넘기고 weatherData값 전달해서 라우트      Navigator.push(context, MaterialPageRoute(builder: (context) {        return WeatherScreen(          parseWeatherData: weatherData,        );      }));    }    @override    Widget build(BuildContext context) {      return Scaffold(        body: Center(            child: ElevatedButton(          style: ElevatedButton.styleFrom(            primary: Colors.black, // background            onPrimary: Colors.white, // foreground          ),          onPressed: () {},          child: Text('Get my location'),        )),      );    }  } |

위에 있는 코드는

<https://github.com/SoonMinKwun/history_of_studying/tree/main/flutter/meonji> 에서도 볼 수 있습니다.

- 관련 기술

geolocator : <https://pub.dev/packages/geolocator>

http : <https://pub.dev/packages/http>

convert : <https://pub.dev/packages/convert>

material : <https://api.flutter.dev/flutter/material/material-library.html>

OpenWeather API : <https://openweathermap.org/api>

Loading : <https://pub.dev/packages/loading>