

**Lab # 9**

[Document subtitle]



**Noman Ali Shah  
19-NTU-CS-118**

**Mobile App Development**

**CSE-4078**

**Lab # 8**

**Q1:**

**QUESTION # 1:**

**Main.dart**

import 'package:flutter/material.dart';

import 'loadingscreen.dart';

void main() {

  runApp(const MyApp());

}

class MyApp extends StatefulWidget {

  const MyApp({Key? key}) : super(key: key);

  @override

  \_MyAppState createState() => \_MyAppState();

}

class \_MyAppState extends State<MyApp> {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(home: LoadingScreen());

  }

}

**Loading Screen.dart**

import 'package:flutter/material.dart';

import 'package:geolocator/geolocator.dart';

import 'locationscreen.dart';

import 'network.dart';

import 'package:flutter\_spinkit/flutter\_spinkit.dart';

var weatherdata;

class LoadingScreen extends StatefulWidget {

const LoadingScreen({Key? key}) : super(key: key);

@override

\_LoadingScreenState createState() => \_LoadingScreenState();

}

class \_LoadingScreenState extends State<LoadingScreen> {

var permissions;

var longitude;

var lattitude;

void getCurrentPosition() async {

permissions = await Geolocator.checkPermission();

if (permissions == LocationPermission.denied) {

permissions = Geolocator.requestPermission();

if (permissions == LocationPermission.denied) {

print('Permission denied');

}

}

Position position = await Geolocator.getCurrentPosition(

desiredAccuracy: LocationAccuracy.best);

lattitude = position.latitude;

longitude = position.longitude;

networkhelper helper = networkhelper(Uri.parse(

'http://api.openweathermap.org/data/2.5/weather?units=metric&lat=$lattitude&lon=$longitude&appid=a67cbb0bb50f199ee71118ac40c1e561'));

weatherdata = await helper.getdata();

print(weatherdata);

Navigator.push(context, MaterialPageRoute(builder: (context) {

return LocationScreen(locationWeather: weatherdata);

}));

print(position.latitude);

print(position.longitude);

}

@override

void initState() {

super.initState();

getCurrentPosition();

}

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(title: Center(child: Text('Abrera Lab 9'))),

body: Column(children: [

Center(

child: SpinKitChasingDots(

color: Colors.grey,

size: 40,

),

),

TextButton(

onPressed: () {

setState(() {

getCurrentPosition();

});

},

child: Text("Get Location"),

),

Text('$longitude'),

Text('$lattitude'),

ElevatedButton(

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(

builder: (context) => LocationScreen(

locationWeather: weatherdata,

)));

},

child: Text("Go to Screen 2")),

]),

);

}

}

**LocationScreen.dart**

import 'package:flutter/material.dart';

import 'weatherIcons.dart';

import 'dart:convert';

class LocationScreen extends StatefulWidget {

const LocationScreen({this.locationWeather});

final dynamic locationWeather;

@override

\_LocationScreenState createState() => \_LocationScreenState();

}

class \_LocationScreenState extends State<LocationScreen> {

WeatherIcon icon = WeatherIcon();

var temp;

var cityname;

var weathericon;

@override

void updateUI(dynamic locationweather) {

temp = locationweather['main']['temp'];

var weathercode = locationweather['weather'][0]['id'];

weathericon = icon.getIcon(weathercode);

cityname = locationweather['name'].toString();

}

void initState() {

super.initState();

print(widget.locationWeather);

updateUI(widget.locationWeather);

}

Widget build(BuildContext context) {

return Column(

children: [

Text(temp.toString()),

Text(cityname.toString()),

weathericon,

ElevatedButton(

onPressed: () {

Navigator.pop(context);

},

child: Text('Back'))

],

);

}

}

**Network.dart**

import 'package:http/http.dart' as http;

import 'dart:convert';

class networkhelper {

final Uri uri;

networkhelper(this.uri);

Future getdata() async {

http.Response response = await http.get(uri);

if (response.statusCode == 200) {

String data = response.body;

return jsonDecode(data);

} else {

print(response.statusCode);

}

}

}

**WeatherIcon.dart**

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

class WeatherIcon {

Widget getIcon(int condition) {

if (condition < 300) {

return Image.network('http://openweathermap.org/img/wn/02d@2x.png');

} else if (condition < 400) {

return Image.network('http://openweathermap.org/img/wn/09d@2x.png');

} else if (condition < 600) {

return Image.network('http://openweathermap.org/img/wn/10d@2x.png');

} else if (condition < 700) {

return Image.network('http://openweathermap.org/img/wn/13d@2x.png');

} else if (condition < 800) {

return Image.network('http://openweathermap.org/img/wn/04d@2x.png');

} else if (condition == 800) {

return Image.network('http://openweathermap.org/img/wn/01d@2x.png');

} else if (condition <= 804) {

return Image.network('http://openweathermap.org/img/wn/03d@2x.png');

} else {

return Image.network(

'https://www.pngfind.com/mpng/iJhbi\_question-mark-png-image-transparent-white-question-mark/');

}

}

}