Angella Benjamin

5023104

IT B1

**MAD Laboratory**

**PRACTICAL NO. 08**

AIM: To include libraries and work with files in the Flutter app.

CO. No. 3. Integrate Firebase services and manage data persistence using file handling and JSON, demonstrating the ability to work with backend technologies. (LO 3.2, LO 4.1, LO 4.2, LO 5.1)

4. Evaluate and execute testing and deployment processes for Flutter applications, ensuring functionality and performance on Android and iOS platforms. (LO5.1, LO 5.2, LO6.1, LO 6.2)

**1. Including Libraries in Flutter**

Pubspec.yaml

flutter:

assets:

- assets/data.json

**2. Including a File with Your App**

assets/data.json:

{

"name": "Angella",

"age": 20,

"skills": ["Flutter", "Dart"]

}

**3. Reading from a File (JSON Asset)**

Main.dart:

import 'dart:convert';

import 'package:flutter/material.dart';

import 'package:flutter/services.dart' show rootBundle;

void main() {

runApp(const MyApp());}

class MyApp extends StatefulWidget {

const MyApp({super.key});

@override

State<MyApp> createState() => \_MyAppState();}

class \_MyAppState extends State<MyApp> {

String output = "Loading...";

@override

void initState() {

super.initState();

loadJson(); }

Future<void> loadJson() async {

final String response = await rootBundle.loadString('assets/data.json');

final data = json.decode(response);

setState(() {

output = "Name: ${data['name']}, Age: ${data['age']}, Skills: ${data['skills']}"; } }

@override

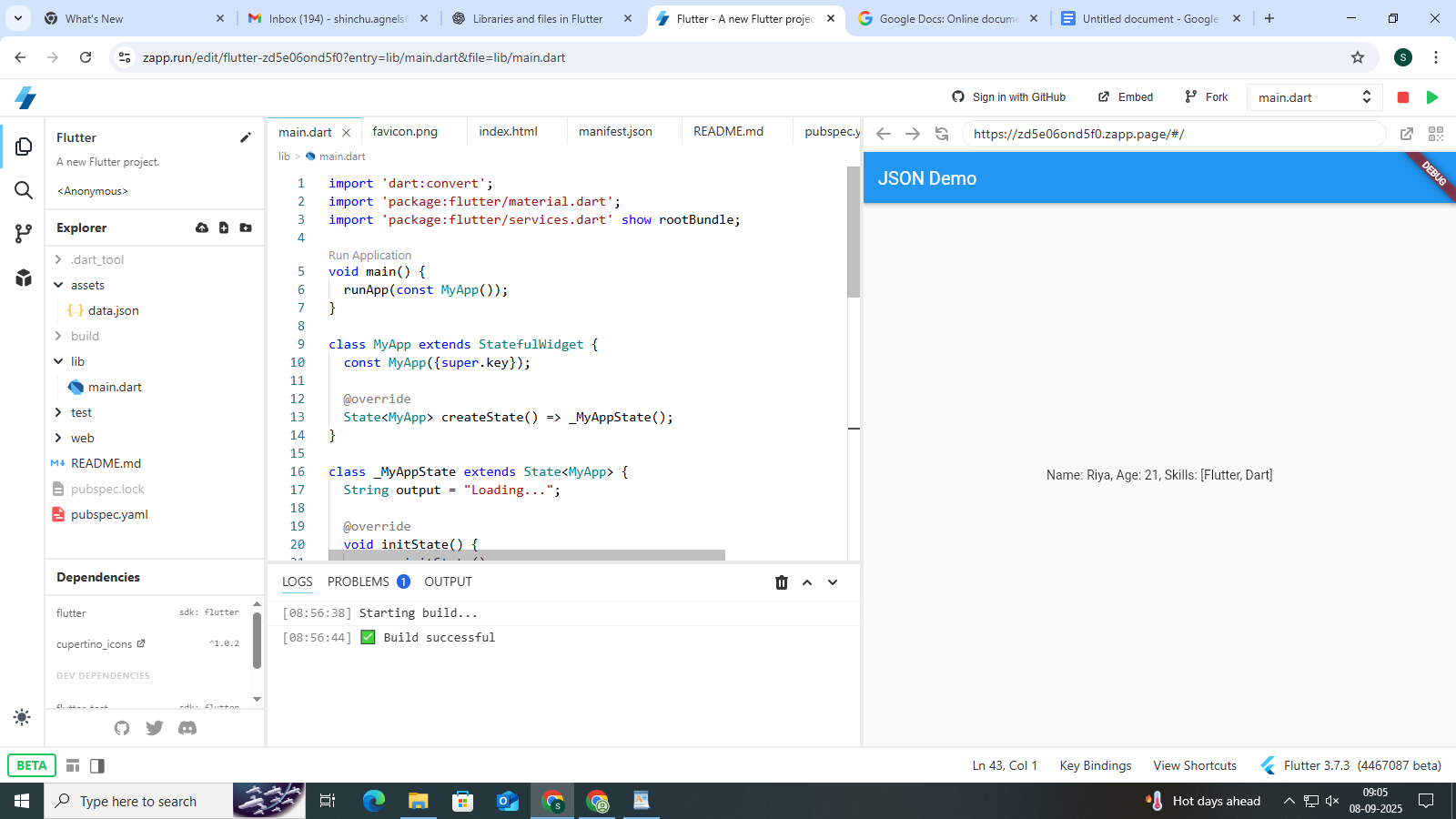
Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(title: const Text("JSON Demo")),

body: Center(child: Text(output)), ), ); }}



**Conclusion**: In Flutter, external libraries are included via the pubspec.yaml file using dependencies. Files such as JSON can be bundled with the app by declaring them in the assets section of pubspec.yaml. JSON is read using rootBundle and dart:convert, while local file read/write operations are handled with dart:io and path\_provider. This enables Flutter apps to efficiently manage data storage, retrieval, and structured exchange through JSON.