**How To Read Data From Local JSON Files In Flutter**

JSON is an open standard file format and data interchange format that uses human-readable text to store and transmit data objects consisting of attribute-value pairs and arrays.

Because of its human readability and with it being lightweight, it is commonly used as an easy data storage format. That means it can be used also in ways not involving data exchange but also in a form of a database.

In this article, we want to see how to load data from a local JSON file. For example, you may download a list of products in the form of JSON and may want to render them in your Flutter app. Well, then this tutorial is helpful in showing you how to do that.

Reading and parsing local JSON data is very simple in Flutter. In fact, it takes just two lines of code to set up access local JSON data as Future async function and make a List for Json Data. In this article, we’re going to see how we can read and parse JSON files locally and show the data in a listview.

**Add local Json in Flutter Assets:**

{

"items": [

{

"id": "1",

"name": "Apple",

"description": "An apple is an edible fruit produced by an apple tree (Malus domestica). Apple trees are cultivated worldwide and are the most widely grown species in the"

},

{

"id": "2",

"name": "Banana",

"description": "banana is an elongated, edible fruit - botanically a berry - produced by several kinds of large herbaceous flowering plants in the genus Musa."

},

{

"id": "3",

"name": "Grape",

"description": "The grapefruit (Citrus × paradisi) is a subtropical citrus tree known for its relatively large, sour to semi-sweet, somewhat bitter fruit."

},

{

"id": "4",

"name": "Mango",

"description": "A mango is an edible stone fruit produced by the tropical tree Mangifera indica which is believed to have originated from the region between northwestern "

},

{

"id": "5",

"name": "Orange",

"description": "An orange is a fruit of various citrus species in the family Rutaceae ); it primarily refers to Citrus × sinensis, which is also called sweet orange"

}

]

}

The code which is used to fetch data from the JSON file is given below

// Fetch content from the json file

Future<void> readJson() async {

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

final data = await json.decode(response);

setState(() {

jsonData = data["items"];

});

}

Add “**assets**” folder to our project and move **data.json** into it. Now, declare the JSON file in the assets section in your **pubspec.yaml** file.

# To add assets to your application, add an assets section, like this:

assets:

- assets/data.json

Let’s take an example of how to read data from local JSON files in Flutter in our flutter application.

import 'dart:convert';

import 'package:flutter/material.dart';

import 'package:flutter/services.dart';

class LocalJsonScreen extends StatefulWidget {

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

@override

State<LocalJsonScreen> createState() => \_LocalJsonScreenState();

}

class \_LocalJsonScreenState extends State<LocalJsonScreen> {

List jsonData = [];

// Fetch content from the json file

Future<void> readJson() async {

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

final data = await json.decode(response);

setState(() {

jsonData = data["items"];

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

centerTitle: true,

title: const Text(

'Read Data From Local Json',

),

),

body: Padding(

padding: const EdgeInsets.all(25),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Center(

child: ElevatedButton(

child: const Text('Load Fruit Data'),

onPressed: readJson,

),

),

// Display the data loaded from data.json

jsonData.isNotEmpty

? Expanded(

child: ListView.builder(

itemCount: jsonData.length,

itemBuilder: (context, index) {

return Card(

margin: const EdgeInsets.all(10),

child: ListTile(

leading: CircleAvatar(

child: Text(jsonData[index]["id"])),

title: Text(jsonData[index]["name"]),

subtitle: Text(jsonData[index]["description"]),

),

);

},

),

)

: const SizedBox()

],

),

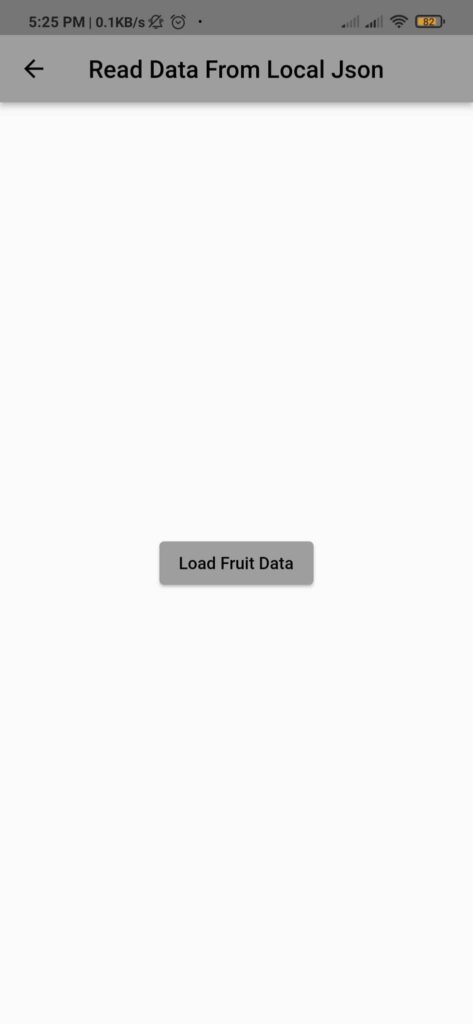
),

);

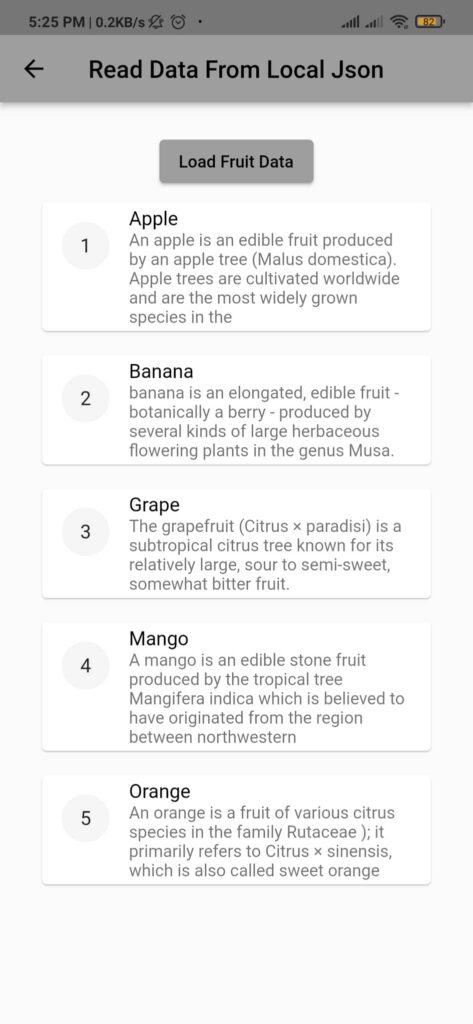
}

}

Meanwhile, when we run this application in the emulator or device, we should get the UI similar to the screenshot below:



When we click on Load Fruit Data the Following JSON data is Load from the local device which is shown below



**Get the full snippet from** [here](https://github.com/sagarkoju/ArticleBlog/tree/master)