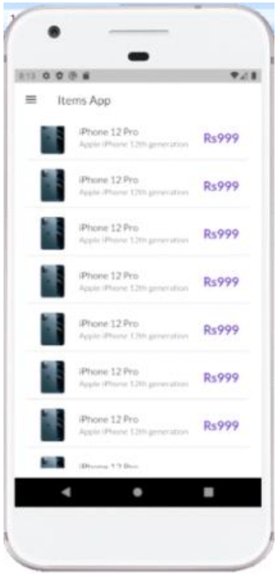
Flutter Part6

Monday, October 11, 2021
1:32 PM

1. How to load Json File with some pre-defined data.
2. We have below data where we are showing one product multiple times. Where we are loading one product from dart file and showing.
3. Now we will use Json file with different items.



Items App

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

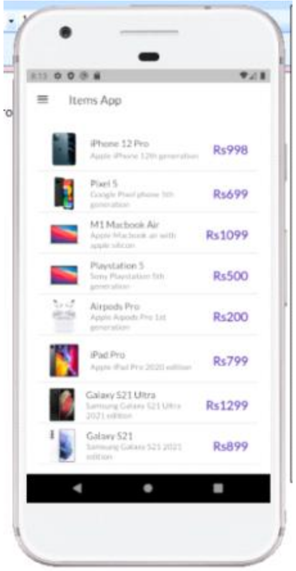
iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro
Apple iPhone 12th generation
Rs999

iPhone 12 Pro

4. Create HomePage into StateFullWidget.

5. Load data from JSON and screen will be like :



Items App

iPhone 12 Pro
Apple iPhone 12th generation
Rs998

Pixel 5
Google Pixel phone 5th generation
Rs699

M1 Macbook Air
Apple Macbook air with apple silicon
Rs1099

Playstation 5
Sony Playstation 5th generation
Rs500

Airpods Pro
Apple Aipods Pro 1st generation
Rs200

iPad Pro
Apple iPad Pro 2020 edition
Rs799

Galaxy S21 Ultra
Samsung Galaxy S21 Ultra 2021 edition
Rs1299

Galaxy S21
Samsung Galaxy S21 2021 edition
Rs899

Code :

Homepage.dart

```
import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:flutterapplication1/models/items.dart';
import 'package:flutterapplication1/widgets/drawer.dart'
;
import 'package:flutterapplication1/widgets/itemwidget.d
art';
class HomePage extends StatefulWidget {
  const HomePage({Key? key}) : super(key: key);
  @override
  State<HomePage> createState() => _HomePageState();
```

Items.dart

```
import 'dart:convert';
class CatalogModel {
  static List<Item> items = <Item>[];
}
class Item {
  final int id;
  final String name;
  final String desc;
  final num price;
  final String color;
  final String image;
  Item(
}
class _HomePageState extends State<HomePage> {
  final int days = 30;
  final String name = "Rajat";
  @override
  void initState() {
    super.initState();
    loadData();
  }
  loadData() async {
    final catalogJson =
        await rootBundle.loadString("assets/files/catalog.json");
    final decodedData = jsonDecode(catalogJson);
    var productsData = decodedData["products"];
    CatalogModel.items = List.from(productsData)
        .map<Item>((item) => Item.fromMap(item))
        .toList();
    setState(() {});
  }
  @override
  Widget build(BuildContext context) {
    // final dummyList = List.generate(20, (index) => CatalogModel.items[0]);
    return Scaffold(
      appBar: AppBar(
        title: Text("Items App"),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: (CatalogModel.items != null && CatalogModel.items.isNotEmpty)
            ? ListView.builder(
                itemCount: CatalogModel.items.length,
                itemBuilder: (context, index) => ItemWidget(
                  item: CatalogModel.items[index],
                ),
              )
            : Center(
                child: CircularProgressIndicator(),
              ),
      ),
      drawer: MyDrawer(),
    );
  }
}
```

```
      {required this.id,
      required this.name,
      required this.desc,
      required this.price,
      required this.color,
      required this.image});
  Item copyWith({
    required int id,
    required String name,
    required String desc,
    required num price,
    required String color,
    required String image,
  }) {
    return Item(
      id: id,
      name: name,
      desc: desc,
      price: price,
      color: color,
      image: image,
    );
  }
  Map<String, dynamic> toMap() {
    return {
      'id': id,
      'name': name,
      'desc': desc,
      'price': price,
      'color': color,
      'image': image,
    };
  }
  factory Item.fromMap(Map<String, dynamic> map) {
    // if (map == null) return null;
    return Item(
      id: map['id'],
      name: map['name'],
      desc: map['desc'],
      price: map['price'],
      color: map['color'],
      image: map['image'],
    );
  }
  String toJson() => json.encode(toMap());
  factory Item.fromJson(String source) => Item.fromMap(json.decode(source));
  @override
  String toString() {
    return 'Item(id: $id, name: $name, desc: $desc, price: $price, color: $color, image: $image)';
  }
  @override
  bool operator ==(Object o) {
    if (identical(this, o)) return true;
    return o is Item &&
        o.id == id &&
        o.name == name &&
        o.desc == desc &&
        o.price == price &&
        o.color == color &&
        o.image == image;
  }
  @override
  int get hashCode {
    return id.hashCode ^
        name.hashCode ^
        desc.hashCode ^
        price.hashCode ^
        color.hashCode ^
        image.hashCode;
  }
}
```