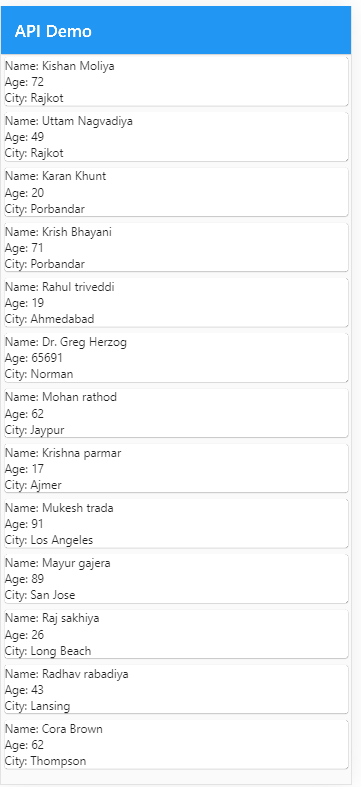
***Unit-5, Lab-2, Programs-3, 4***

***Program-2.3* WAP to call rest api and display data in listview.builder with pagination (Lazzy Loading).**

import 'dart:convert';  
import 'package:diploma\_projects/Lab-7/services/MyAPI.dart';  
import 'package:flutter/material.dart';  
import 'package:http/http.dart' as http;  
  
class StudentListPage extends StatefulWidget {  
 @override  
 State<StudentListPage> createState() => \_StudentListPageState();  
}  
  
class \_StudentListPageState extends State<StudentListPage> {  
 Future<List> getStudentData() async {  
 http.Response student = await http.get(Uri.*parse*(MyAPI().api));  
 return jsonDecode(student.body.toString());  
 }  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('API Demo'),  
 ),  
 body: FutureBuilder(  
 future: getStudentData(),  
 builder: (context, snapshot) {  
 if (snapshot.hasData &&  
 snapshot.connectionState != ConnectionState.waiting) {  
 return ListView.builder(  
 itemCount: snapshot.data!.length,  
 itemBuilder: (context, index) {  
 return Card(  
 elevation: 3,  
 child: Column(  
 crossAxisAlignment: CrossAxisAlignment.start,  
 children: [  
 Text('Name: ${snapshot.data![index]['Name']}'),  
 Text('Age: ${snapshot.data![index]['Age']}'),  
 Text('City: ${snapshot.data![index]['City']}'),  
 ],  
 ),  
 );  
 },  
 );  
 } else {  
 return Center(  
 child: CircularProgressIndicator(),  
 );  
 }  
 },  
 ),  
 );  
 }  
}

***OutPut:***



**Program-2.4 WAP to display country list in dropdown on selection of country call state api and fill data in state list after selection state call city api and fill city list in driodown.**

import 'dart:convert';  
import 'package:diploma\_projects/Lab-7/services/MyAPI.dart';  
import 'package:flutter/material.dart';  
import 'package:http/http.dart' as http;  
  
class DropDownData extends StatefulWidget {  
 @override  
 State<DropDownData> createState() => \_DropDownDataState();  
}  
class \_DropDownDataState extends State<DropDownData> {  
 Future<List> getStudentData() async {  
 http.Response student = await http.get(Uri.*parse*(MyAPI().api));  
 return jsonDecode(student.body.toString());  
 }  
  
 List<dynamic> apiData = [];  
 List<String> Countrys = [];  
 List<String> Citys = [];  
 List<String> States = [];  
 String? CountryName;  
 String? StateName;  
 String? CityName;  
 @override  
 void initState(){  
 super.initState();  
 getData();  
 }  
  
 void getData() async {  
 await getStudentData().then((value) {  
 setState(() {  
 apiData = value;  
 });  
 });  
 getCountry();  
 }  
  
 void getCountry(){  
 for(int i=0; i<apiData.length ; i++){  
 if (!Countrys.contains(apiData[i]['Country'])) {  
 Countrys.add(apiData[i]['Country']);  
 }  
 }  
 }  
  
 void getState(){  
 for(int i=0; i<apiData.length ; i++){  
 if (!States.contains(apiData[i]['State']) && apiData[i]['Country'] == CountryName) {  
 States.add(apiData[i]['State']);  
 }  
 }  
 }  
  
 void getCity(){  
 for(int i=0; i<apiData.length ; i++){  
 if (!Citys.contains(apiData[i]['City']) && apiData[i]['Country'] == CountryName && apiData[i]['State'] == StateName) {  
 Citys.add(apiData[i]['City']);  
 }  
 }  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('API Demo'),  
 ),  
 body: FutureBuilder(  
 future: getStudentData(),  
 builder: (context, snapshot) {  
 if (snapshot.hasData &&  
 snapshot.connectionState != ConnectionState.waiting) {  
 return Card(  
 elevation: 3,  
 child: Column(  
 crossAxisAlignment: CrossAxisAlignment.start,  
 children: [  
 DropdownButton(  
 isExpanded: true,  
 value: CountryName,  
 hint: Text('Select Country'),  
 items: Countrys.map((item) {  
 return DropdownMenuItem(  
 value: item,  
 child: Text(item),  
 );  
 }).toList(),  
 onChanged: (value) {  
 setState(() {  
 CountryName = value;  
 getState();  
 });  
 },  
 ),  
 DropdownButton(  
 isExpanded: true,  
 value: StateName,  
 hint: Text('Select State'),  
 items: States.map((item) {  
 return DropdownMenuItem(  
 value: item,  
 child: Text(item),  
 );  
 }).toList(),  
 onChanged: (value) {  
 setState(() {  
 StateName = value;  
 getCity();  
 });  
 },  
 ),  
 DropdownButton(  
 isExpanded: true,  
 value: CityName,  
 hint: Text('Select City'),  
 items: Citys.map((item) {  
 return DropdownMenuItem(  
 value: item,  
 child: Text(item),  
 );  
 }).toList(),  
 onChanged: (value) {  
 setState(() {  
 CityName = value;  
 });  
 },  
 ),  
 ],  
 ),  
 );  
 } else {  
 return const Center(child: CircularProgressIndicator());  
 }  
 },  
 ),  
 );  
 }  
}

***OutPut:***

