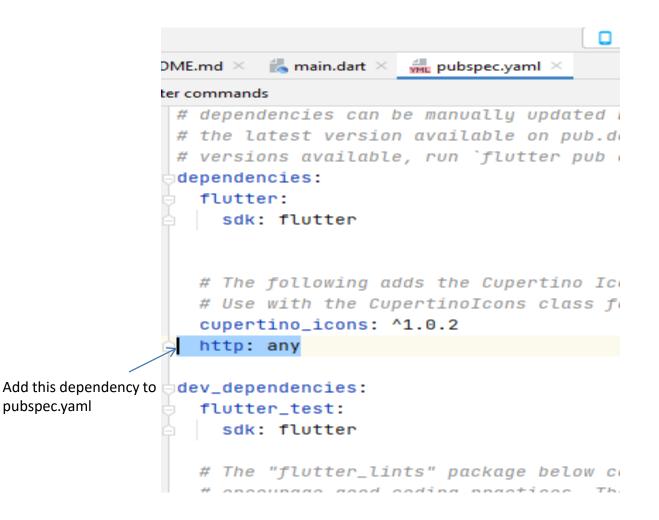
Flutter and Rest api

Access the REST API in the Flutter app

- Flutter provides http package to use http resources.
- The http package uses await and async features and provides many high-level methods such as read, get, post, put, head, and delete methods for sending and receiving data from remote locations.
- These methods simplify the development of RESTbased mobile applications.
- Note: To install the http package, open the pubspec.yaml file in your project folder and add http package in the dependency section.



pubspec.yaml

- Async means that this function is
 asynchronous and you might need to wait a
 bit to get its result.
- Await literally means wait here until this function is finished and you will get its return value.

Future<T> class

- The result of an asynchronous computation.
- An asynchronous computation cannot provide a result immediately when it is started, unlike a synchronous computation which does compute a result immediately by either returning a value or by throwing.
- An asynchronous computation may need to wait for something external to the program (reading a file, querying a database, fetching a web page) which takes time.
- Instead of blocking all computation until the result is available, the asynchronous computation immediately returns a Future which will *eventually* "complete" with the result.

Asynchronous programming

- To perform an asynchronous computation, you use an async function which always produces a future.
- Inside such an asynchronous function, you can use the await operation to delay execution until another asynchronous computation has a result.
- While execution of the awaiting function is delayed, the program is not blocked, and can continue doing other things.

Listview.builder

- **ListView.builder** creates a scrollable, linear array of widgets.
- ListView.builder by default does not support child reordering.

- initState()
 - The framework will call this method exactly once for each State object it creates.

Flutter Card

- A card is a sheet used to represent the information related to each other, such as an album, a geographical location, contact details, etc.
- A card in Flutter is in rounded corner shape and has a shadow.
- We mainly use it to store the content and action of a single object.

Write a Flutter program based on RestAPI.

```
main.dart
import 'dart:async';
import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
void main() {
 runApp(const MaterialApp(
   home: HomePage()
 ));
class HomePage extends StatefulWidget {
 const HomePage({Key? key}) : super(key: key);
 @override
 HomePageState createState() => HomePageState();
```

```
class HomePageState extends State<HomePage> {
 late final List data;
 Future<String> getData() async {
  var response = await http.get(
     Uri.parse("https://jsonplaceholder.typicode.com/posts"),
     headers: {
      "Accept": "application/json"
  setState(() {
   data = json.decode(response.body);
  });
  // ignore: avoid_print
  // print(data[1]["title"]);
  return "Success!";
```

```
@override
// ignore: must_call_super
void initState(){
  getData();
 @override
Widget build(BuildContext context){
  return Scaffold(
   appBar: AppBar(title: const Text("Listview"), backgroundColor: Colors.blue),
   body: ListView.builder(
    // ignore: unnecessary_null_comparison
    itemCount: data.length,
    itemBuilder: (BuildContext context, int index){
      return Card(
       child: Text(data[index]["title"]),
```