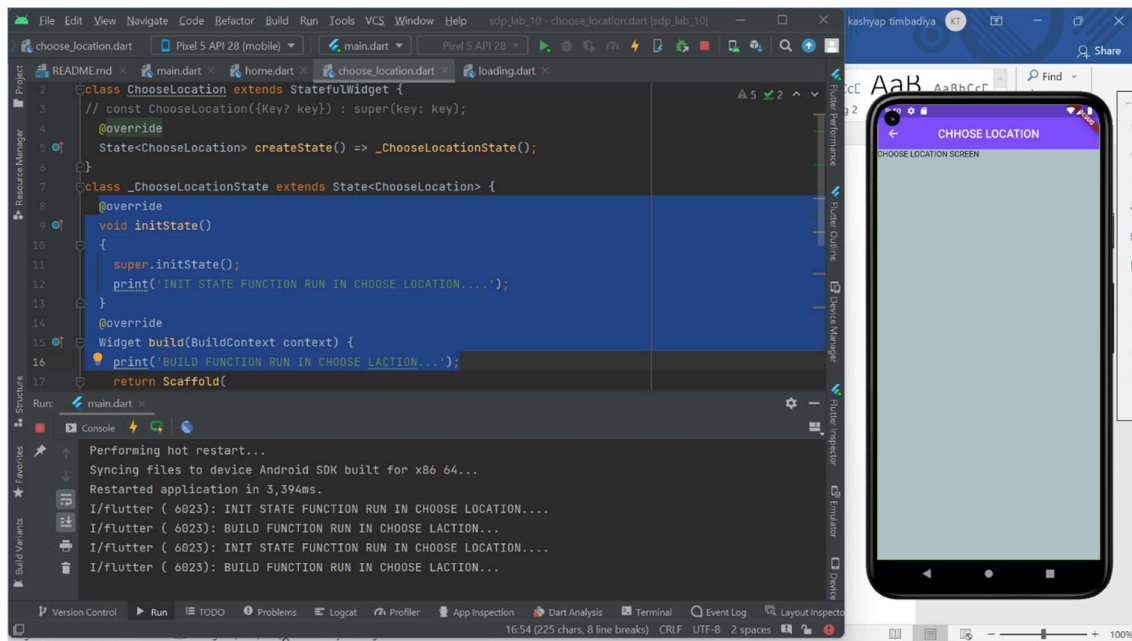
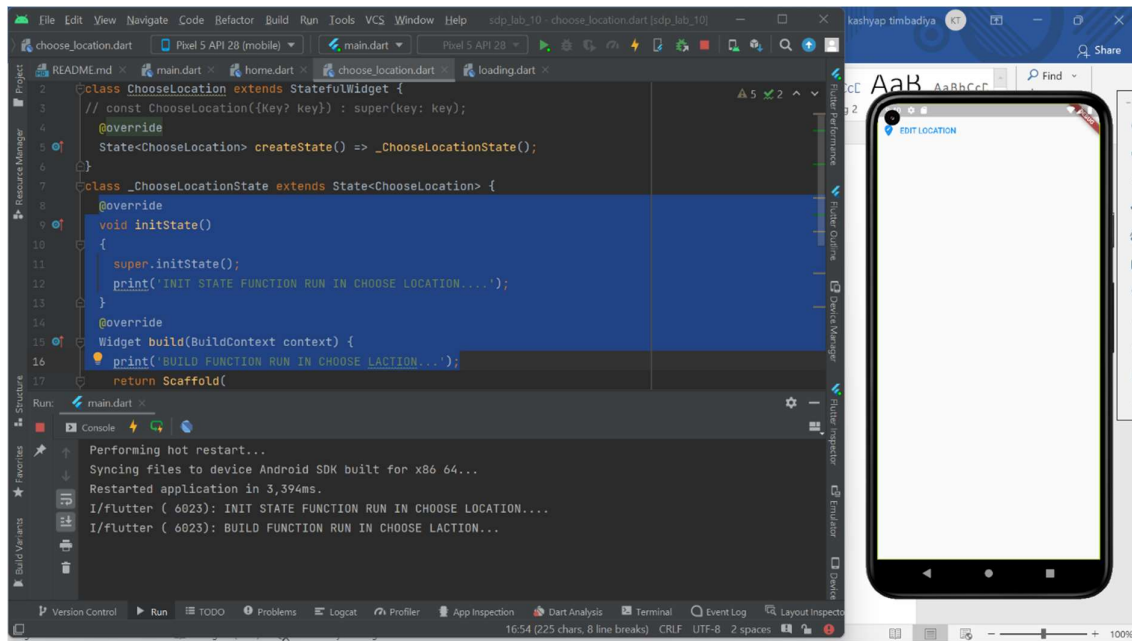


Code Test1: Test for initState() and build()



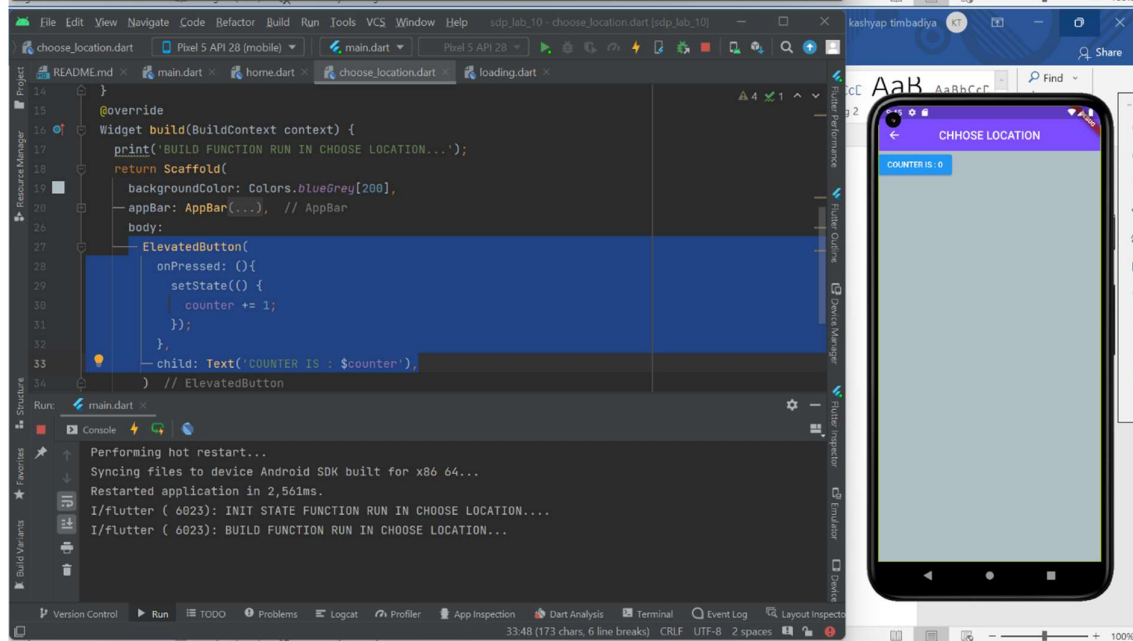
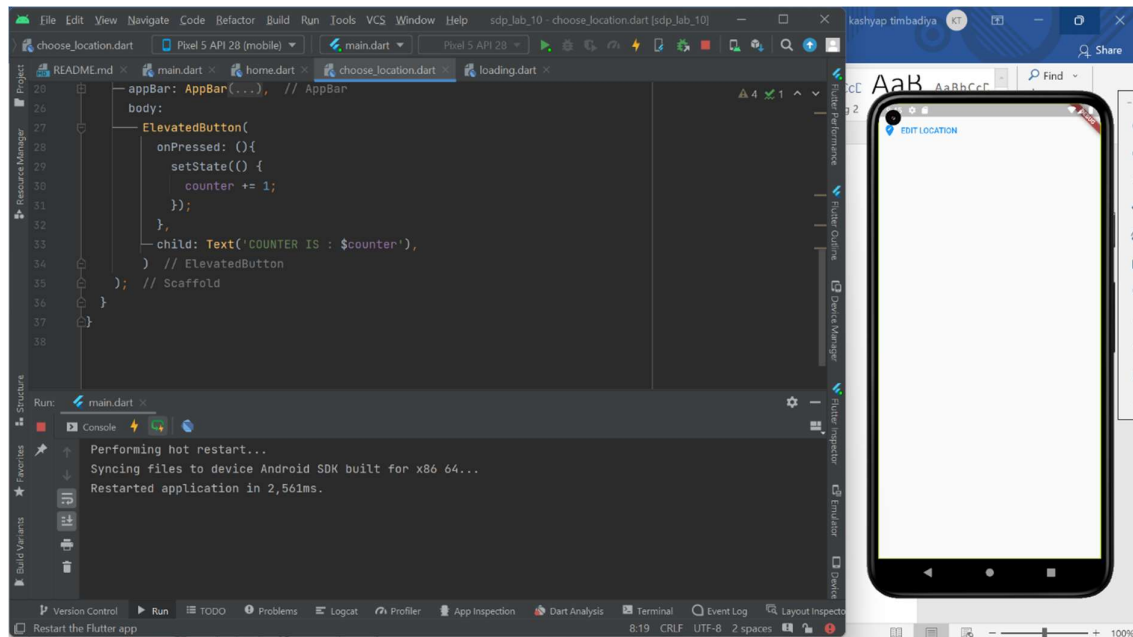
Now test for After setState() check the changes

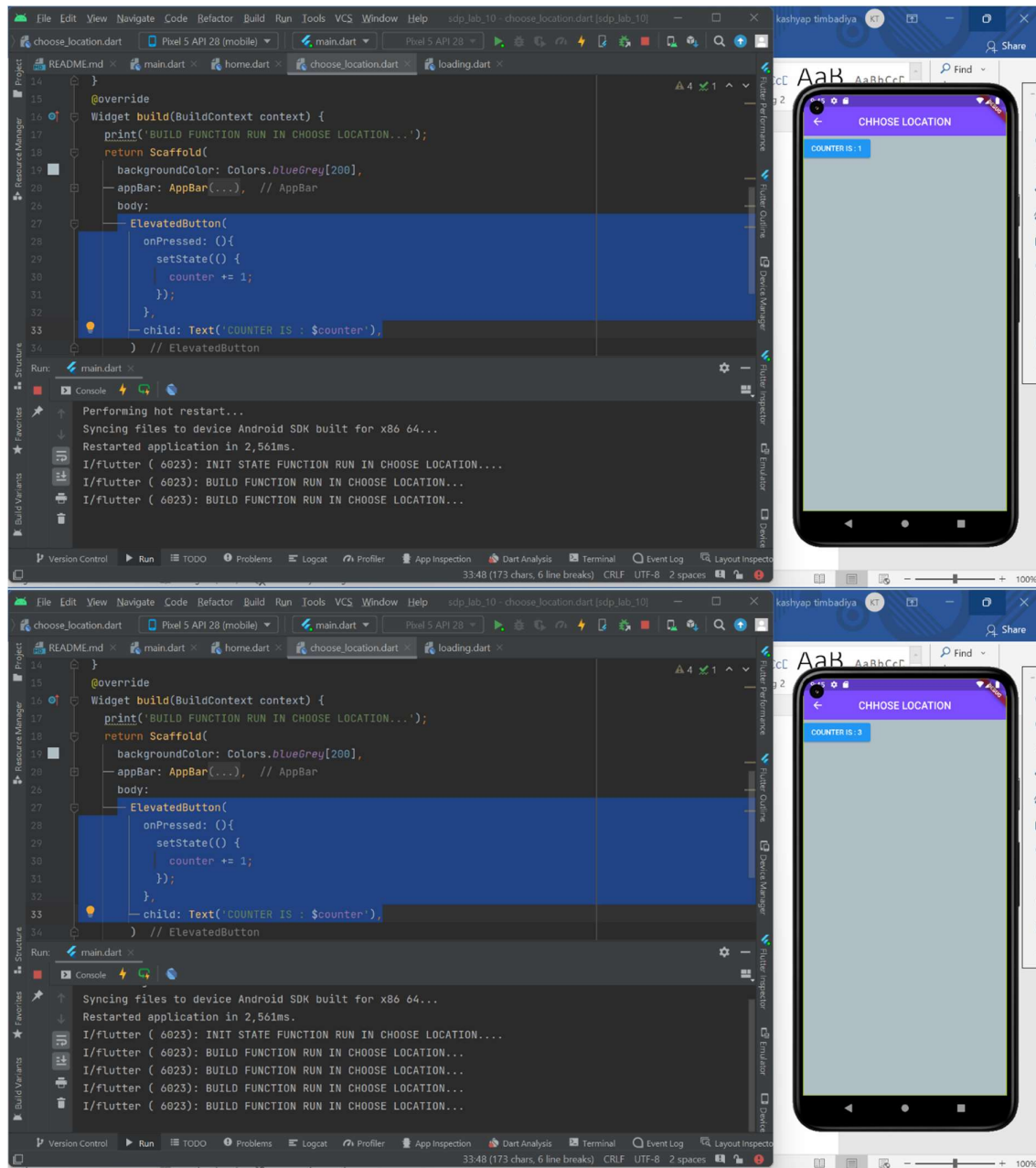
```
import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key: key);
  @override
  State<ChooseLocation> createState() => _ChooseLocationState();
}
class _ChooseLocationState extends State<ChooseLocation> {
```

```

int counter = 0;
@override
void initState()
{
    super.initState();
    print('INIT STATE FUNCTION RUN IN CHOOSE LOCATION....');
}
@override
Widget build(BuildContext context) {
    print('BUILD FUNCTION RUN IN CHOOSE LOCATION...');
    return Scaffold(
        backgroundColor: Colors.blueGrey[200],
        appBar: AppBar(
            backgroundColor: Colors.deepPurpleAccent,
            title: Text('CHHOSE LOCATION'),
            centerTitle: true,
            elevation: 0,
        ),
        body:
            ElevatedButton(
                onPressed: () {
                    setState(() {
                        counter += 1;
                    });
                },
                child: Text('COUNTER IS : $counter'),
            )
    );
}
}

```





Code test 2- Async in flutter

```

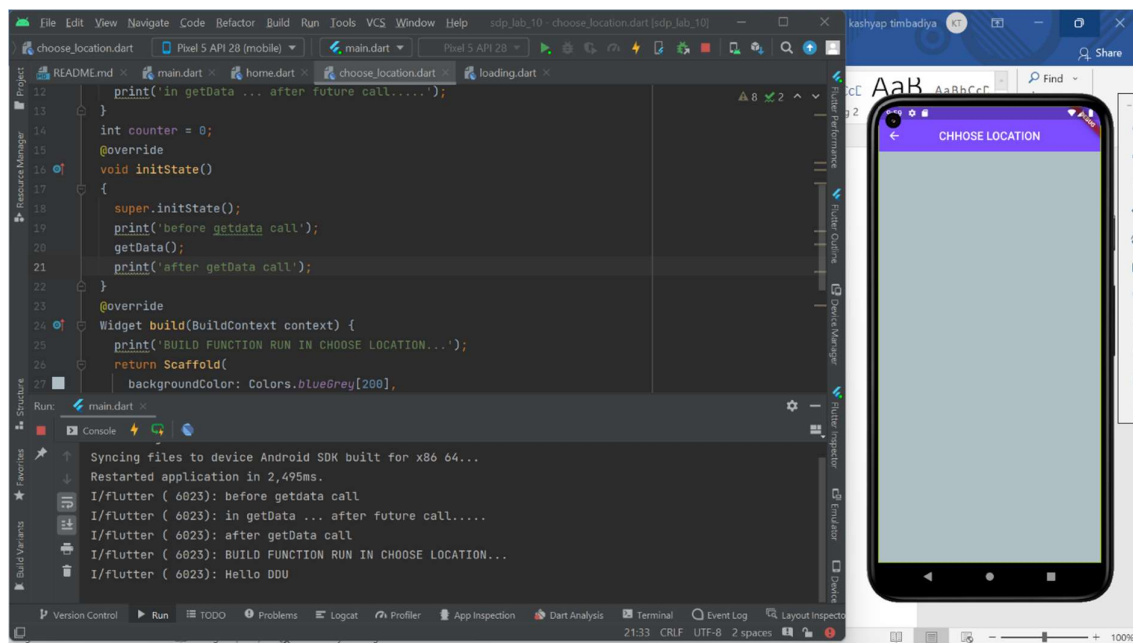
import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key: key);
  @override
  State<ChooseLocation> createState() => _ChooseLocationState();
}
class _ChooseLocationState extends State<ChooseLocation> {
  void getData() {
    Future.delayed(Duration(seconds: 4), () {
      print('Hello DDU');
    });
    print('in getData ... after future call.....');
  }
}

```

```

int counter = 0;
@override
void initState()
{
  super.initState();
  print('before getdata call');
  getData();
  print('after getData call');
}
@override
Widget build(BuildContext context) {
  print('BUILD FUNCTION RUN IN CHOOSE LOCATION...');
  return Scaffold(
    backgroundColor: Colors.blueGrey[200],
    appBar: AppBar(
      backgroundColor: Colors.deepPurpleAccent,
      title: Text('CHHOSE LOCATION'),
      centerTitle: true,
      elevation: 0,
    ),
  );
}

```



Above code teach us that code not blocked due to async call...

But some times we need to wait for response of request...

➤ Some time new request depends on the data of first request... So in such situation second request must have to wait until the first one not completed..

```

import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key: key);
}

```

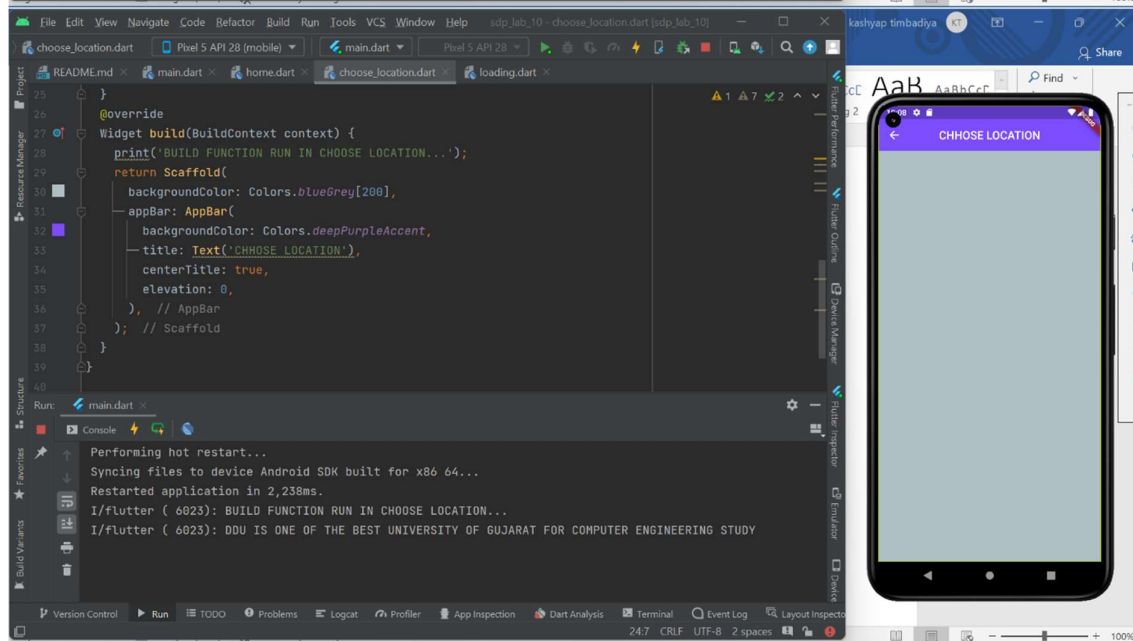
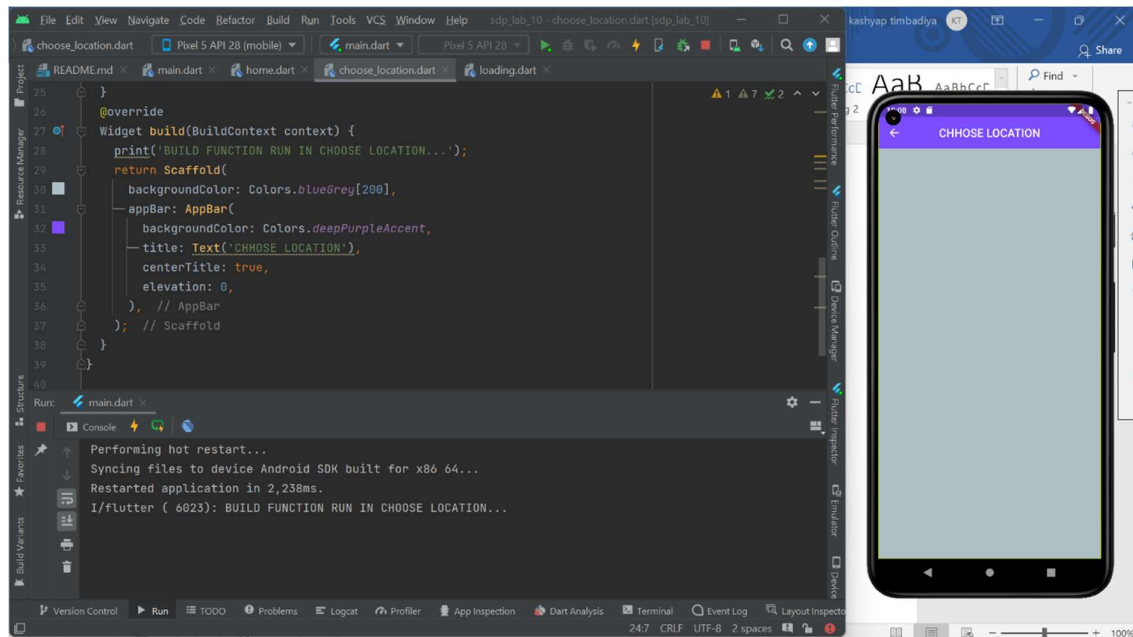
```

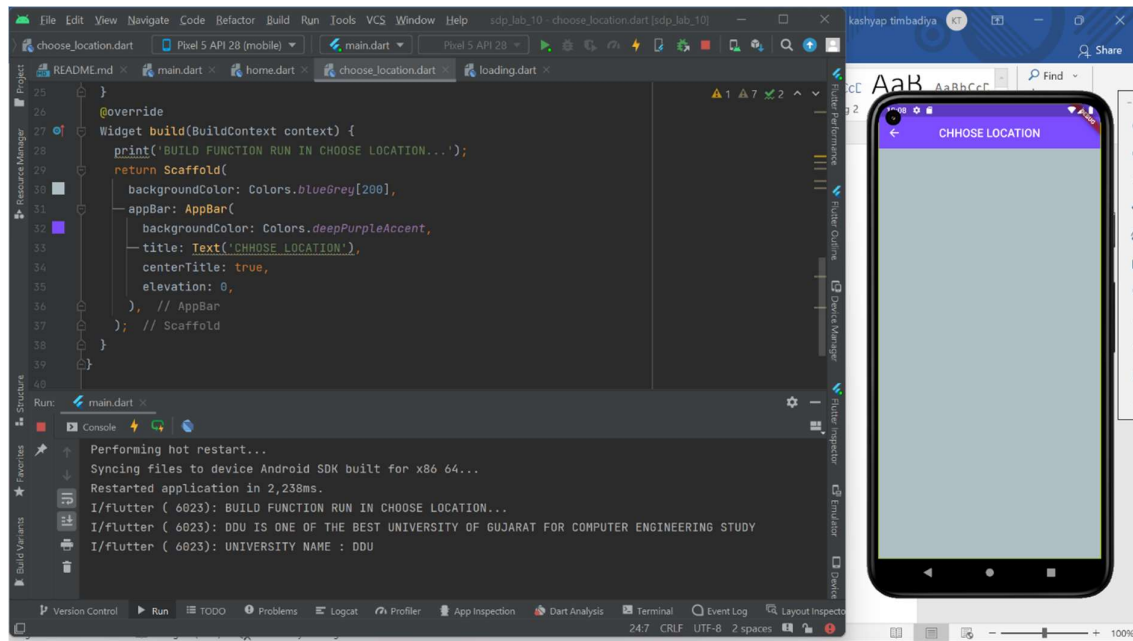
@override
State<ChooseLocation> createState() => _ChooseLocationState();
}
class _ChooseLocationState extends State<ChooseLocation> {
  void getData(){
    int counter = 0;
    Future.delayed(Duration(seconds: 4), () {
      print('UNIVERSITY NAME : DDU');
    });
    Future.delayed(Duration(seconds: 2), () {
      print('DDU IS ONE OF THE BEST UNIVERSITY OF GUJARAT FOR COMPUTER
ENGINEERING STUDY');
    });
  }

  @override
  void initState()
  {
    super.initState();
    //print('before getdata call');
    getData();
    //print('after getdata call');
  }

  @override
  Widget build(BuildContext context) {
    print('BUILD FUNCTION RUN IN CHOOSE LOCATION...');
    return Scaffold(
      backgroundColor: Colors.blueGrey[200],
      appBar: AppBar(
        backgroundColor: Colors.deepPurpleAccent,
        title: Text('CHHOSE LOCATION'),
        centerTitle: true,
        elevation: 0,
      ),
    );
  }
}

```





Solution: use Async, Await, Future, Delayed

```
import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key: key);
  @override
  State<ChooseLocation> createState() => _ChooseLocationState();
}
class _ChooseLocationState extends State<ChooseLocation> {
  int counter = 0;
  void getData() async {
    // below code is similar like to
    // simulate network request for a username
    // its just learning how flutter/dart response to Async
    String username = await Future.delayed(Duration(seconds: 4), () {
      return 'UNIVERSITY NAME : DDU';
    });
    // print('in getData ...after future call....');
    // the following code is depends on 1st one ..but what happened at here?
    // suppose it is network simulation request to fetch out the bio-data of..
    // user name that is received by 1st request.....

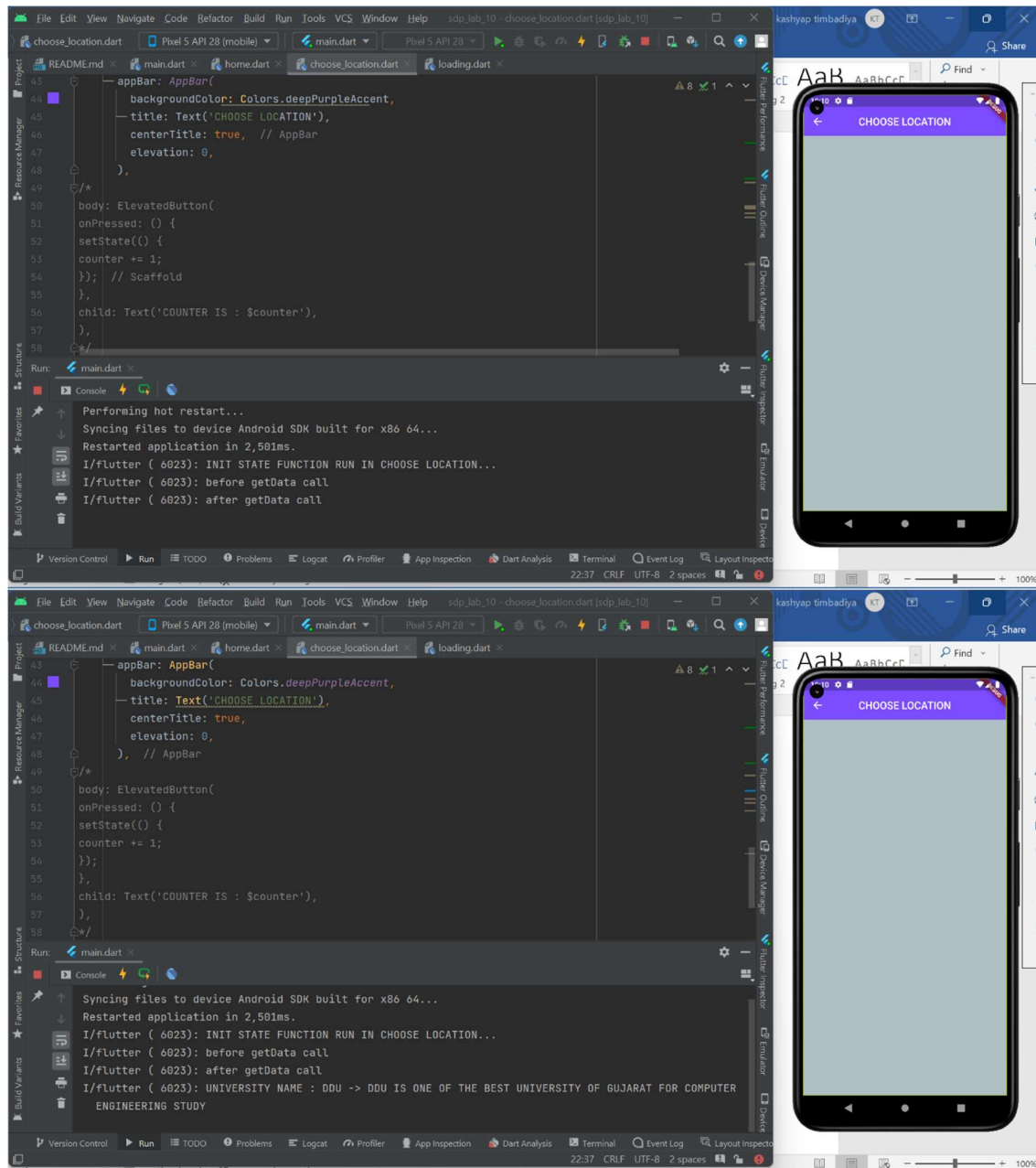
    String bio = await Future.delayed(Duration(seconds: 2), () {
      return 'DDU IS ONE OF THE BEST UNIVERSITY OF GUJARAT FOR COMPUTER
ENGINEERING STUDY';
    });
    print('$username -> $bio');
  }
  @override
  void initState() {
    // TODO: implement initState
    super.initState();
    print('INIT STATE FUNCTION RUN IN CHOOSE LOCATION...');
    print('before getData call');
    getData();
    print('after getData call');
```



```

    }
    @override
    Widget build(BuildContext context) {
// print('BUILD FUNCTION RUN IN CHOOSE LOCATION...');
    return Scaffold(
      backgroundColor: Colors.blueGrey[200],
// when we come/route from different screen IN APPBAR FLUTTER WILL
DEFAULTLY
// PLACED "BACK ARROW <- .....
      appBar: AppBar(
        backgroundColor: Colors.deepPurpleAccent,
        title: Text('CHOOSE LOCATION'),
        centerTitle: true,
        elevation: 0,
      ),
//
body: ElevatedButton(
  onPressed: () {
    setState(() {
      counter += 1;
    });
  },
  child: Text('COUNTER IS : $counter'),
),
*/
);
}
}

```



Code Test 3: packages and API calling basic

Classwork for SDP x http | Dart Package x Lab7 - Google Drive x +

pub.dev/packages/http/install

Classroom

Run this command:

With Dart:

```
$ dart pub add http
```

With Flutter:

```
$ flutter pub add http
```

This will add a line like this to your package's pubspec.yaml (and run an implicit `dart pub get`):

```
dependencies:
  http: ^0.13.5
```

Alternatively, your editor might support `dart pub get` or `flutter pub get`. Check the docs for your editor to learn more.

Import it

Now in your Dart code, you can use:

Metadata

A composable, multi-platform, Future-based API for HTTP requests.

[Repository \(GitHub\)](#)

[View/report issues](#)

Documentation

[API reference](#)

License

[BSD-3-Clause \(LICENSE\)](#)

Dependencies

[async](#), [http_parser](#), [meta](#), [path](#)

More

[Packages that depend on http](#)

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help sdp_lab_10 - pubspec.yaml [sdp_lab_10]

sdp_lab_10 pubspec.yaml

Flutter commands

Pub get Pub upgrade Pub outdated Flutter doctor

```
25 # consider running 'flutter pub upgrade --major-versions'. Alternatively,
26 # dependencies can be manually updated by changing the version numbers below to
27 # the latest version available on pub.dev. To see which dependencies have newer
28 # versions available, run 'flutter pub outdated'.
29 dependencies:
30   flutter:
31     sdk: flutter
32   http: ^0.13.5
33
34 # The following adds the Cupertino Icons font to your application.
35 # Use with the CupertinoIcons class for iOS style icons.
36 cupertino_icons: ^1.0.2
37
38 Document 1/1 dependencies: http: ^0.13.5
```

Project

android [sdp_lab_10_android]

build

ios

lib

pages

choose_location.dart

home.dart

loading.dart

main.dart

test

.gitignore

.metadata

.packages

analysis_options.yaml

pubspec.lock

pubspec.yaml

README.md

Run

main.dart

Console

Performing hot restart...

Syncing files to device Android SDK built for x86 64...

Restarted application in 2,959ms.

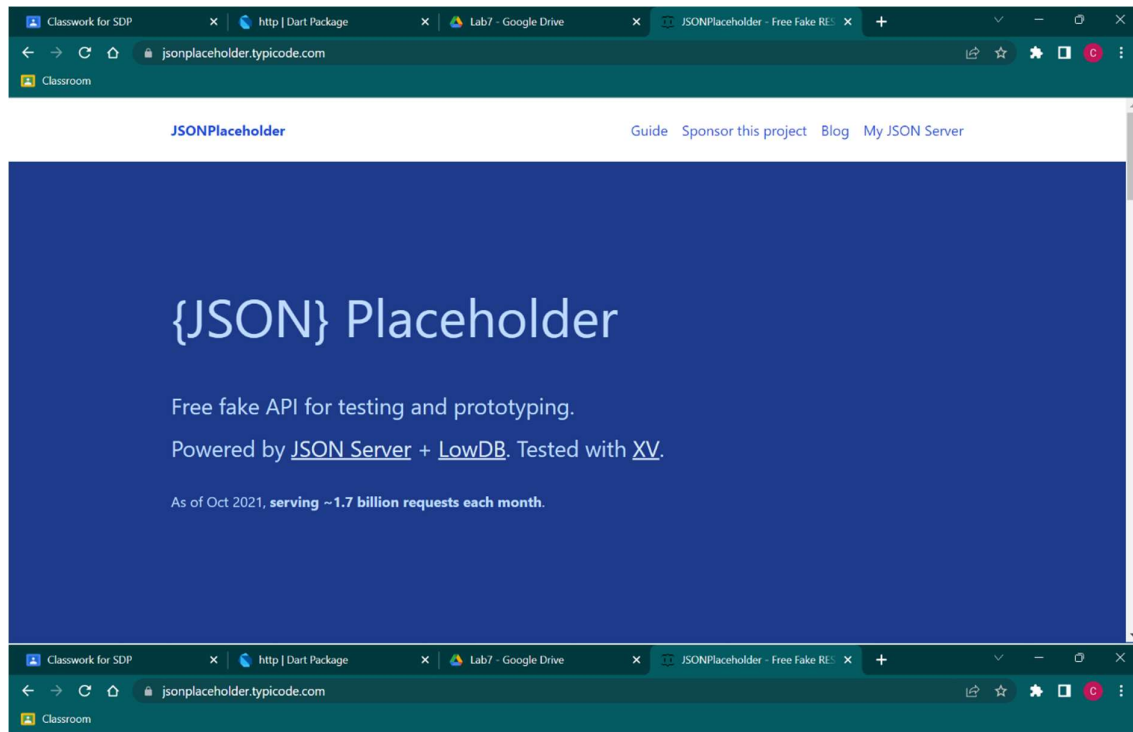
I/flutter (6023): BUILD FUNCTION RUN IN CHOOSE LOCATION...

I/flutter (6023): DDU IS ONE OF THE BEST UNIVERSITY OF GUJARAT FOR COMPUTER ENGINEERING STUDY

I/flutter (6023): UNIVERSITY NAME : DDU

Version Control Run TODO Problems Logcat Profiler App Inspection Dart Analysis Terminal

6:41 CRLF UTF-8 2 spaces Schema: pubspec.json



Try it

Run this code here, in a console or from any site:

```
fetch('https://jsonplaceholder.typicode.com/todos/1')  
  .then(response => response.json())  
  .then(json => console.log(json))
```

Run script

```
{}
```

- Copy Ctrl+C
- Copy link to highlight
- Go to <https://jsonplaceholder.typicode.com/todos/1>
- Print... Ctrl+P
- Inspect

Main.dart

```
import 'package:flutter/material.dart';  
import 'package:sdp_lab_10/pages/choose_location.dart';
```

```

import 'package:sdp_lab_10/pages/home.dart';
import 'package:sdp_lab_10/pages/loading.dart';
/*
10

void main() => runApp(MaterialApp(
  // home: Home(),
  routes: { // routes is Map variable with 'key:value' pairs
    // '/about' ....eg.....routes the screen through 'about' widgets
    // '/contact'....
    '/': (context) => Loading(), // base routes....base widget file...main
    file of project
    // above statement will creates error...because it conflict with 'home:
    Home(),
    // because both statements tell flutter to initialize the app from their
    given
    location
    '/home': (context) => Home(),
    '/location': (context) => ChooseLocation(),
  }
));
*/
void main() => runApp(MaterialApp(
  // home: Home(),
  // instead of making home: property to make any page to initialize at
  beginning...
  // we can use following code ....
  // initialRoute: '/home',
  initialRoute: '/',
  routes: {
    '/': (context) => Loading(),
    '/home': (context) => Home(),
    '/location': (context) => ChooseLocation(),
  }
));

```

Home.dart

```

import 'package:flutter/material.dart';
class Home extends StatefulWidget {
  @override
  State<Home> createState() => _HomeState();
}
class _HomeState extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Column(
          children: [
            TextButton.icon(
              onPressed: () {
                Navigator.pushNamed(context, '/location');
              },
              icon: Icon(Icons.edit_location),
              label: Text('EDIT LOCATION'),
            )
          ]
        )
      )
    );
  }
}

```

```

        ],
      ),
    ),
  );
}
}

```

Choose_location.dart

```

import 'package:flutter/material.dart';
class ChooseLocation extends StatefulWidget {
  // const ChooseLocation({Key? key}) : super(key: key);
  @override
  State<ChooseLocation> createState() => _ChooseLocationState();
}
class _ChooseLocationState extends State<ChooseLocation> {
  int counter = 0;

  @override
  Widget build(BuildContext context) {
    // print('BUILD FUNCTION RUN IN CHOOSE LOCATION...');
    return Scaffold(
      backgroundColor: Colors.blueGrey[200],
      appBar: AppBar(
        backgroundColor: Colors.deepPurpleAccent,
        title: Text('CHHOSE LOCATION'),
        centerTitle: true,
        elevation: 0,
      ),
    );
  }
}

```

Loading.dart

```

import 'package:flutter/material.dart';
import 'package:http/http.dart';
import 'dart:convert';

class Loading extends StatefulWidget {
  // const Loading({Key? key}) : super(key: key);
  @override
  State<Loading> createState() => _LoadingState();
}
class _LoadingState extends State<Loading> {
  // void getData() async {
  //   String username = await Future.delayed(Duration(seconds: 4), () {
  //     return 'UNIVERSITY NAME : DDU';
  //   });
  //   //
  //   String bio = await Future.delayed(Duration(seconds: 2), () {

```

```

//      return 'DDU IS ONE OF THE BEST UNIVERSITY OF GUJARAT FOR '
//      'COMPUTER ENGINEERING STUDY';
//    });
//
//    print('$username -> $bio');
//
//  }
//
//  @override
//  void initState() {
//    super.initState();
//    print('INIT STATE FUNCTION RUN IN CHOOSE LOCATION...');
//    print('before getdata call');
//    getData();
//    print('after getdata call');
//  }

void getData() async {
  final response = await
    get(Uri.parse('https://jsonplaceholder.typicode.com/albums/1'));
  print(response.body); // it response in JSON form out put ...we need
MAPformat..
// print(response.body.userId); // this will not work. because its not
MAPformat..
// TO CONVERT JSON TO MAP..WE NEED TO IMPORT convert package....
  Map data = jsonDecode(response.body);
  print(data);
  print(data['title']);
}

@override
void initState() {
  super.initState();
  getData();
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Text('LOADING SCREEN'),
  );
}
}

```