Lab 10

Link for assignment **:** https://github.com/Diya-Lad/SDP

Tutorial-1

Stateful Widget:

When a Flutter builds a Stateful Widget, it creates a State object. This object is where all the mutable state for that widget is held.

The concept of state is defined by two things:

1. The data used by the widget might change.
2. The data can't be read synchronously when the widget is built. (All state must be established by the time the build method is called).

The lifecycle of stateful widget has the following simplified steps:

- createState()
- mounted == true
- initState()
- didChangeDependencies()
- build()
- didUpdateWidget()
- setState()
- deactivate()
- dispose()
- mounted == false

In main.dart:

```
void main() => runApp(MaterialApp(
// home: Home(),
// instead of making home: property to make any page
to initialize a beginning...
// we can use following code ....
    initialRoute: '/home',
    routes: {
      '/': (context) => Loading(),
      '/home': (context) => Home(),
      '/location': (context) => ChooseLocation(),
    }
));
```

In loading.dart:

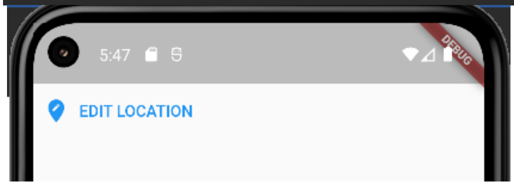
```
import 'package:flutter/material.dart';
class Loading extends StatefulWidget {
// const Loading({Key? key}) : super(key: key);
  @override
  State<Loading> createState() => _LoadingState();
}
class _LoadingState extends State<Loading> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Text('LOADING SCREEN'),
    );
  }
}
```

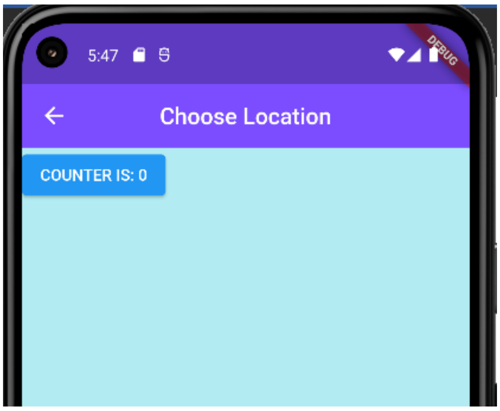
In home.dart:

```
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'),
              )
            ],
          )
      ),
    );
  }
}
```

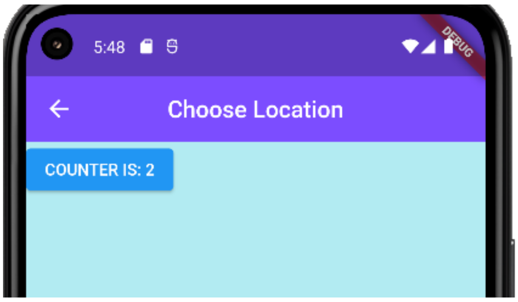
In choose_location.dart:

```
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> {
  late int counter;
  @override
  void initState(){
    super.initState();
     counter=0;

    print("INIT STATE FUNCTION RUN IN CHOOSE
LOCATION");
  }
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.cyan[100],
      appBar: AppBar(
        backgroundColor: Colors.deepPurpleAccent,
        title: Text('Choose Location'),
        centerTitle: true,
        elevation: 0,
      ),

      body: ElevatedButton(
        onPressed: (){
          setState((){
            counter +=1;
          });
        },
        child: Text('COUNTER IS: $counter'),
      )
    );
  }
}
```



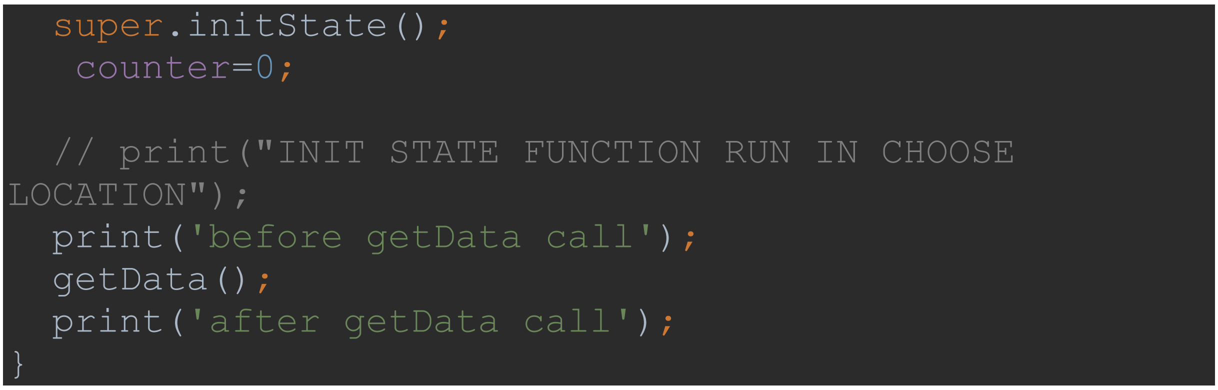


After pressing button twice:

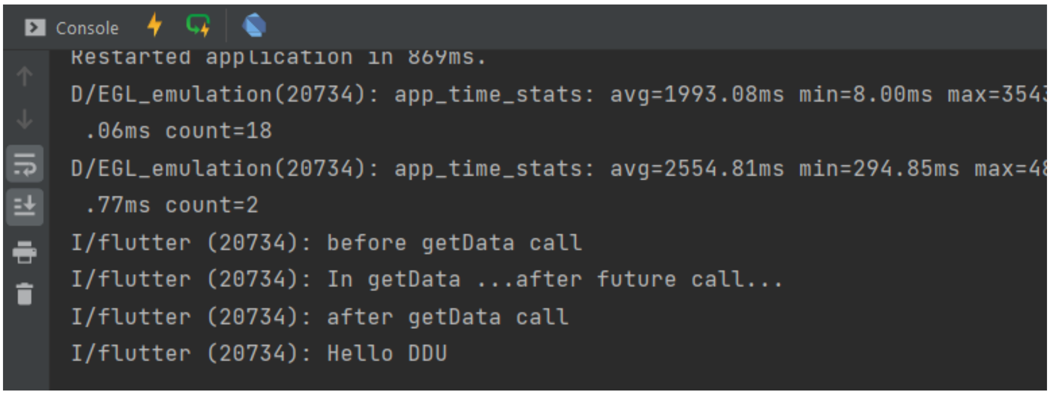


In choose_location.dart:

```
void getData(){
  Future.delayed(Duration(seconds:4),(){
    print("Hello DDU");
  });
  print('In getData ...after future call...');
}

@override
void initState(){
```

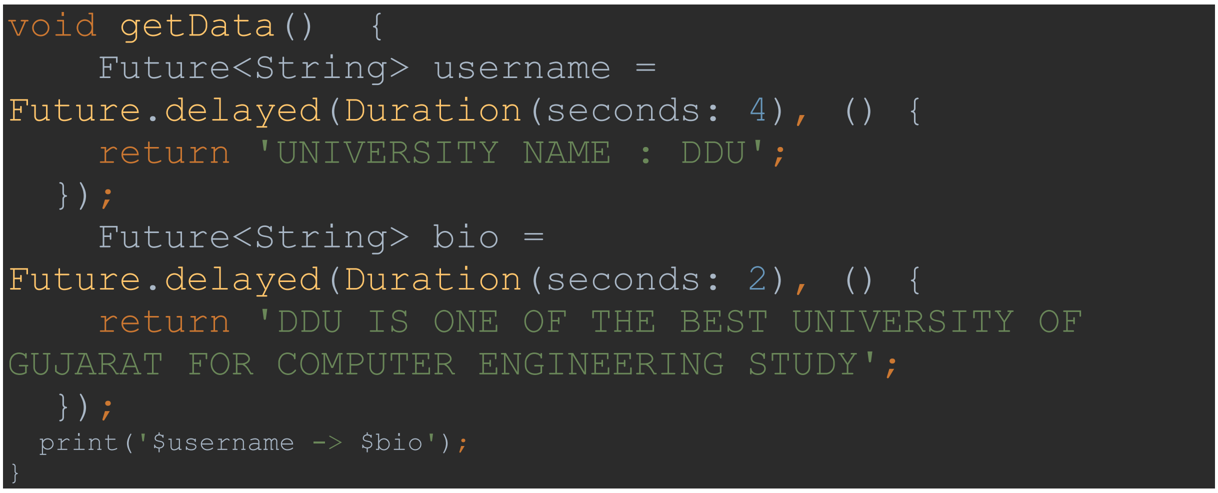


```
  super.initState();
   counter=0;

  // print("INIT STATE FUNCTION RUN IN CHOOSE
LOCATION");
  print('before getData call');
  getData();
  print('after getData call');
}
```

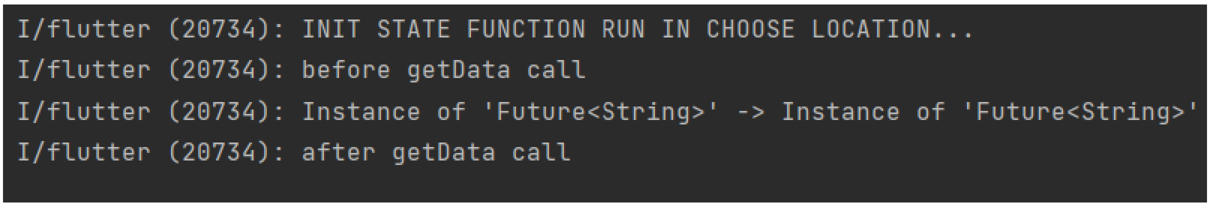


```
Restarted application in 869ms.
D/EGL_emulation(20734): app_time_stats: avg=1993.08ms min=8.00ms max=354
 .06ms count=18
D/EGL_emulation(20734): app_time_stats: avg=2554.81ms min=294.85ms max=4
 .77ms count=2
I/flutter (20734): before getData call
I/flutter (20734): In getData ...after future call...
I/flutter (20734): after getData call
I/flutter (20734): Hello DDU
```

If we don’t use async , await then…

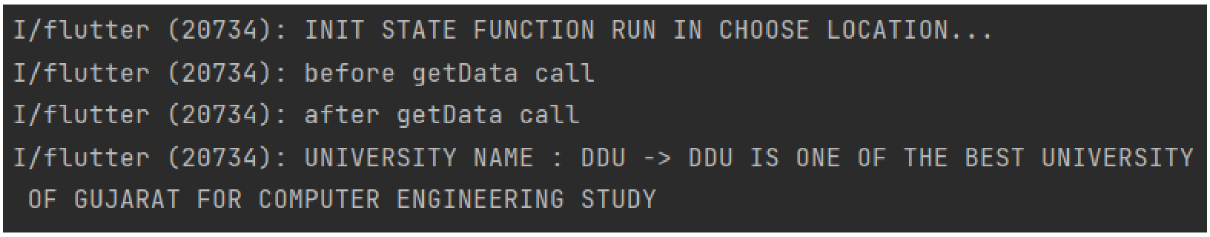


```
void getData()  {
    Future<String> username =
Future.delayed(Duration(seconds: 4), () {
    return 'UNIVERSITY NAME : DDU';
  });
    Future<String> bio =
Future.delayed(Duration(seconds: 2), () {
    return 'DDU IS ONE OF THE BEST UNIVERSITY OF
GUJARAT FOR COMPUTER ENGINEERING STUDY';
  });
  print('$username -> $bio');
}
```

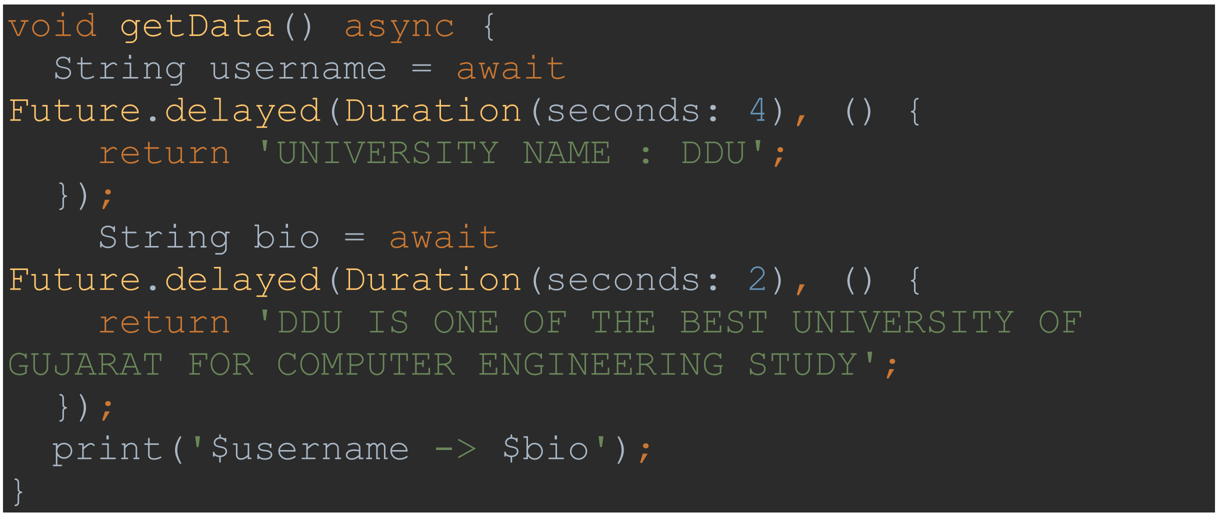


```
I/flutter (20734): INIT STATE FUNCTION RUN IN CHOOSE LOCATION...
I/flutter (20734): before getData call
I/flutter (20734): Instance of 'Future<String>' -> Instance of 'Future<String>'
I/flutter (20734): after getData call
```

After adding async and await print can’t execute so we get desired output:



```
I/flutter (20734): INIT STATE FUNCTION RUN IN CHOOSE LOCATION...
I/flutter (20734): before getData call
I/flutter (20734): after getData call
I/flutter (20734): UNIVERSITY NAME : DDU -> DDU IS ONE OF THE BEST UNIVERSITY
 OF GUJARAT FOR COMPUTER ENGINEERING STUDY
```



```
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');
}
```

async, await, Future, delayed, Duration,

async (Asynchronous function):

- When an async function is called, a Future is immediately returned and the body of the function is executed later.
- As the body of the async function is executed, the Future returned by the function call will be completed along with its result.

Await:

- In async function we can use await keyword which will wait for the result.

future:

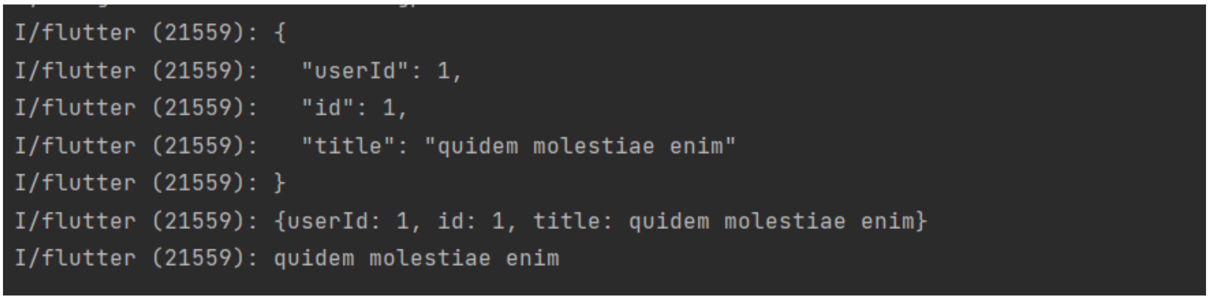
- Dart is a single-threaded programming language.
- Future<T> object represents the result of an asynchronous operation which produces a result of type T.
- If the result is not usable value, then the future’s type is Future<void>.
- A Future represents a single value either a data or an error asynchronously

There are 2 ways to handle Futures:

- Using the Future API
- Using the async and await operation.

In loading.dart:

```
import 'package:flutter/material.dart';
import 'package:http/http.dart';
import 'dart:convert';
class Loading extends StatefulWidget {
  @override
  State<Loading> createState() => _LoadingState();
}
class _LoadingState extends State<Loading> {
  void getData() async {
    final response = await

get(Uri.parse('https://jsonplaceholder.typicode.com/a
lbums/1'));
    print(response.body); // it response in JSON form
out put ...we need MAP format..
    // print(response.body.userId); // this will not
work. because its not MAP format..
    // 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'),
    );
  }
}
```



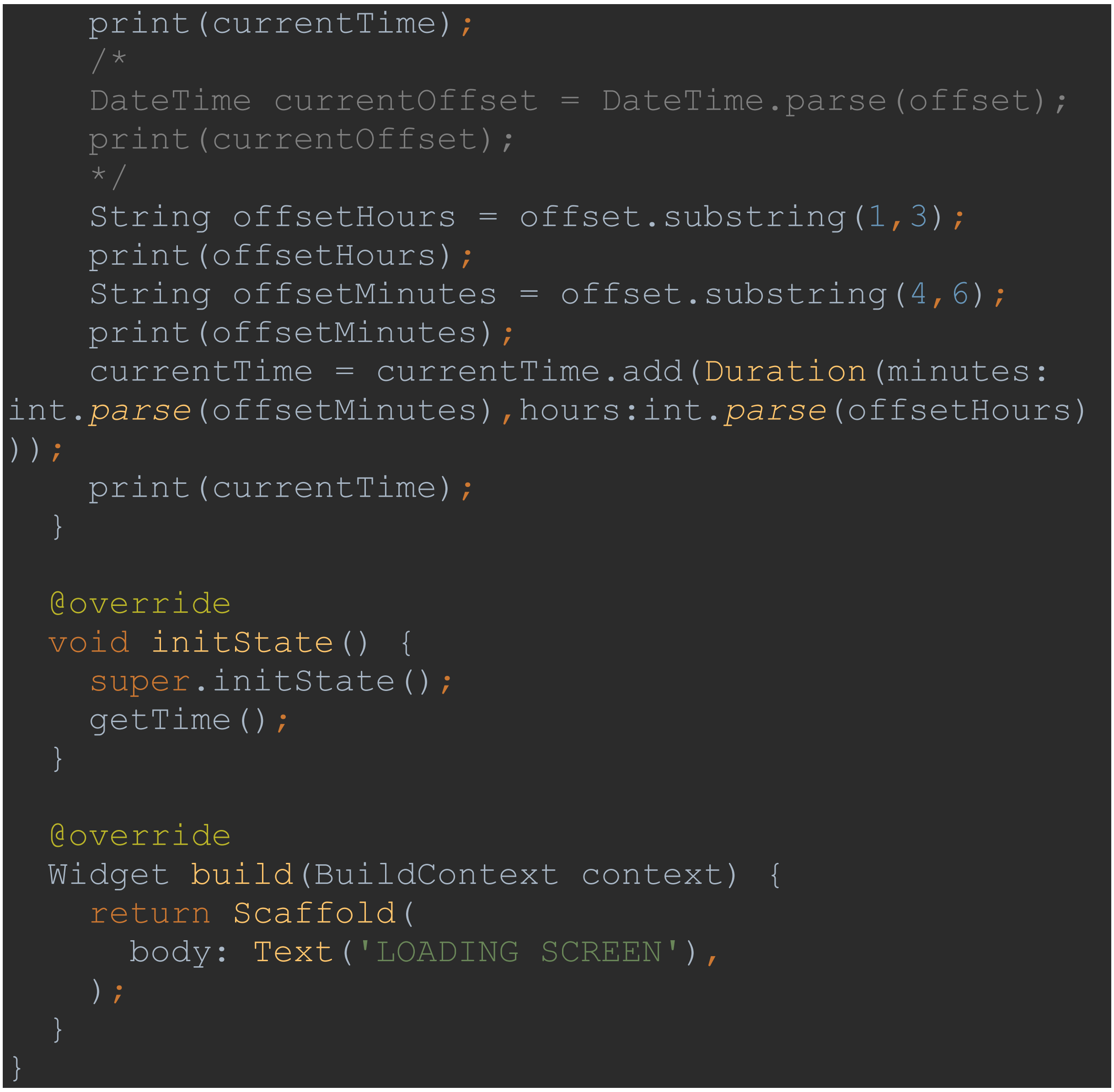
```
I/flutter (21559): {
I/flutter (21559):   "userId": 1,
I/flutter (21559):   "id": 1,
I/flutter (21559):   "title": "quidem molestiae enim"
I/flutter (21559): }
I/flutter (21559): {userId: 1, id: 1, title: quidem molestiae enim}
I/flutter (21559): quidem molestiae enim
```

Tutorial-2

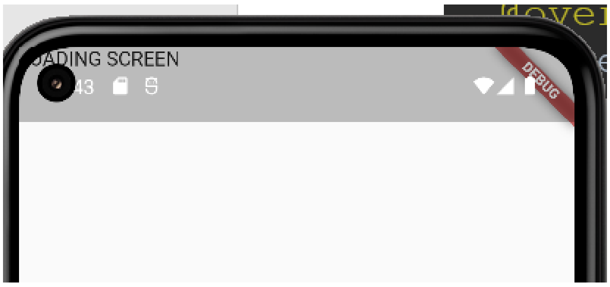
```
import 'package:flutter/material.dart';
import './pages/loading.dart';

void main() => runApp(MaterialApp(
    initialRoute: '/',
    routes: {
      '/': (context) => Loading(),
    }
));
```

In loading.dart:

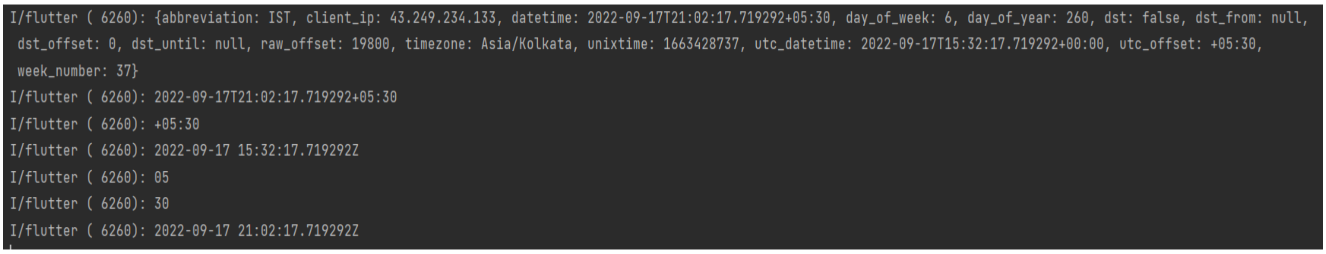
```
import 'package:flutter/material.dart';
import 'package:http/http.dart';
import 'dart:convert';
class Loading extends StatefulWidget {
  @override
  State<Loading> createState() => _LoadingState();
}
class _LoadingState extends State<Loading> {
  void getTime() async {
    // Make Request for time and receive response
    Response response = await

get(Uri.parse('http://worldtimeapi.org/api/timezone/A
sia/Kolkata'));
    Map timeData = jsonDecode(response.body);
    print(timeData);
    // Get particular property form timeData...
    String dateTime = timeData['datetime'];
    String offset = timeData['utc_offset']; //not
dst_offset
    print(dateTime);
    print(offset);
    DateTime currentTime = DateTime.parse(dateTime);
```



```
    print(currentTime);
    /*
    DateTime currentOffset = DateTime.parse(offset);
    print(currentOffset);
    */
    String offsetHours = offset.substring(1,3);
    print(offsetHours);
    String offsetMinutes = offset.substring(4,6);
    print(offsetMinutes);
    currentTime = currentTime.add(Duration(minutes:
int.parse(offsetMinutes),hours:int.parse(offsetHours)
));
    print(currentTime);
  }

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

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



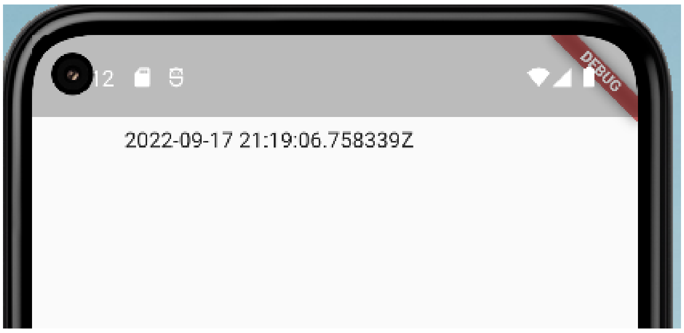
OADING SCREEN



```
I/flutter ( 6260): {abbreviation: IST, client_ip: 43.249.234.133, datetime: 2022-09-17T21:02:17.719292+05:30, day_of_week: 6, day_of_year: 260, dst: false, dst_from: null,
 dst_offset: 0, dst_until: null, raw_offset: 19800, timezone: Asia/Kolkata, unixtime: 1663428737, utc_datetime: 2022-09-17T15:32:17.719292+00:00, utc_offset: +05:30,
 week_number: 37}
I/flutter ( 6260): 2022-09-17T21:02:17.719292+05:30
I/flutter ( 6260): +05:30
I/flutter ( 6260): 2022-09-17 15:32:17.719292Z
I/flutter ( 6260): 05
I/flutter ( 6260): 30
I/flutter ( 6260): 2022-09-17 21:02:17.719292Z
```

In loading.dart:

```
import 'package:flutter/material.dart';
import 'package:http/http.dart';
import 'dart:convert';
import 'package:lab10_2/services/world_time.dart';

class Loading extends StatefulWidget {
  @override
  State<Loading> createState() => _LoadingState();
}
class _LoadingState extends State<Loading> {
  String? time = 'LOADING..........';

  void setWorldTime() async {
    WordTime timeinstance =
    WordTime(location: 'kolkata', flag: 'india.png',
url: 'Asia/Kolkata');
    await timeinstance.getTime();
    // print(timeinstance.time);
    setState(() {
      time = timeinstance.time;
    });
  }

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
        body: Padding(
          padding: EdgeInsets.all(60.0),
          child: Text(time.toString()),
        )
    );
  }
}
```



```
2022-09-17 21:19:06.758339Z
```