Dharmsinh Desai University, Nadiad

Faculty of Technology

Department of Computer Engineering

ASSIGNMENT 11

Subject: Smart Device Programming

Submitted To:

Prof. Jignesh Shah

CE Department

Submitted By:

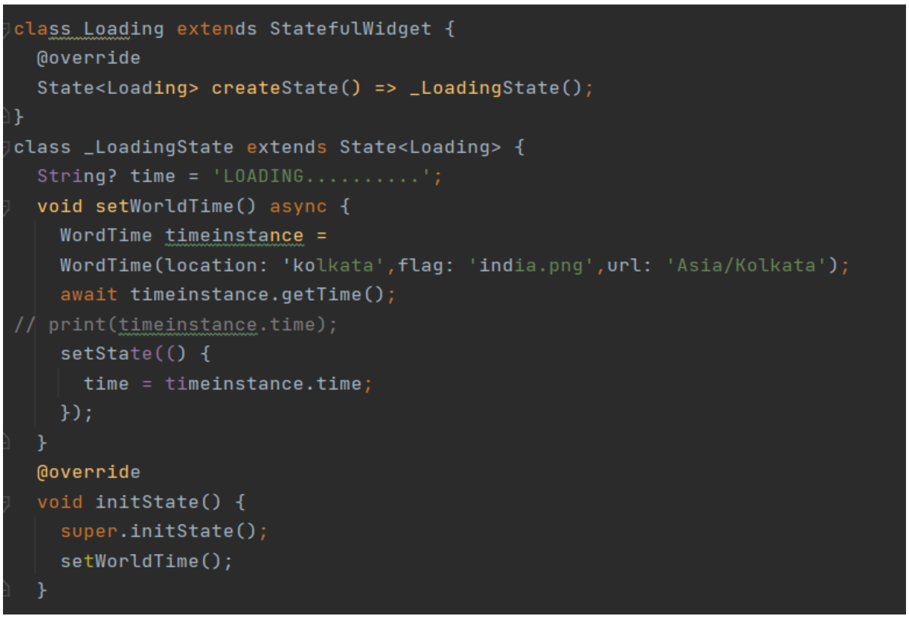
Sherathiya Dhruvi A.

Student ID:20CEUOS006

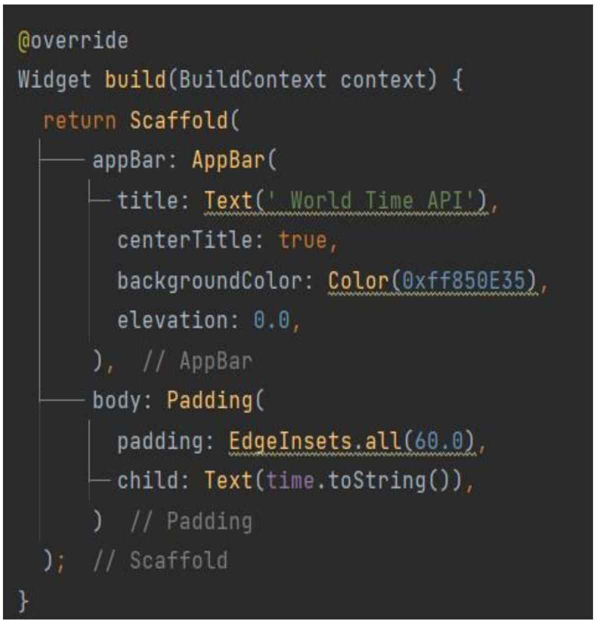
Roll No.: CE126

Code test 1: Handling Error: Try Catch

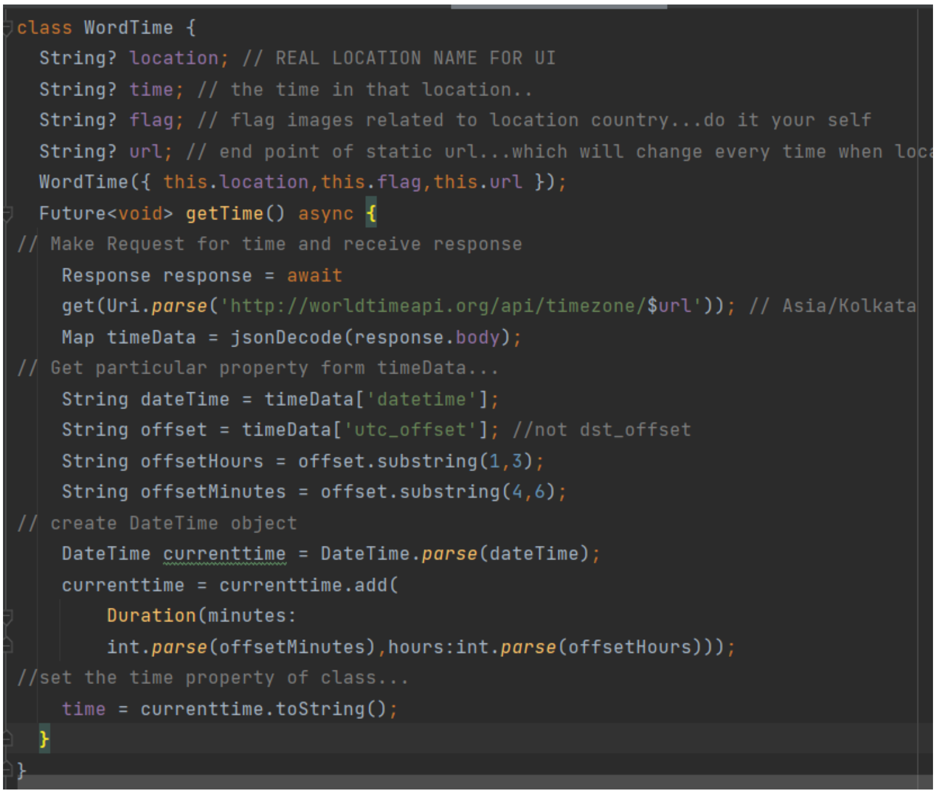
Using the world-time API, continue developing the app that we started in the last lab.



```
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();
  }
```

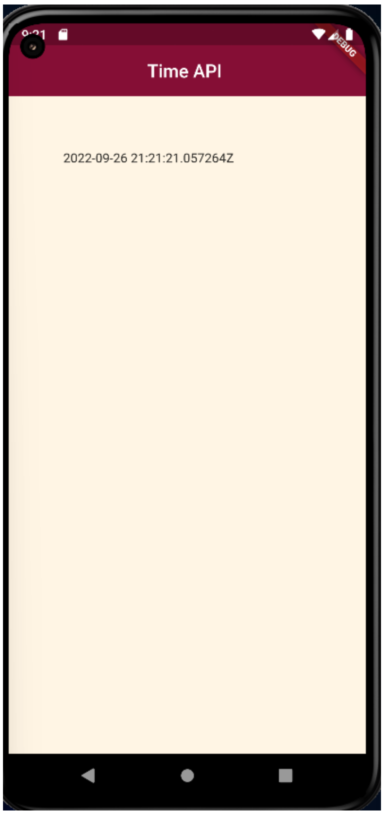


```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text(' World Time API'),
      centerTitle: true,
      backgroundColor: Color(0xff850E35),
      elevation: 0.0,
    ), // AppBar
    body: Padding(
      padding: EdgeInsets.all(60.0),
      child: Text(time.toString()),
    ) // Padding
  ); // Scaffold
}
```



```
class WordTime {
  String? location; // REAL LOCATION NAME FOR UI
  String? time; // the time in that location..
  String? flag; // flag images related to location country...do it your self
  String? url; // end point of static url...which will change every time when loca
  WordTime({ this.location,this.flag,this.url });
  Future<void> getTime() async {
// Make Request for time and receive response
    Response response = await
    get(Uri.parse('http://worldtimeapi.org/api/timezone/$url')); // Asia/Kolkata
    Map timeData = jsonDecode(response.body);
// Get particular property form timeData...
    String dateTime = timeData['datetime'];
    String offset = timeData['utc_offset']; //not dst_offset
    String offsetHours = offset.substring(1,3);
    String offsetMinutes = offset.substring(4,6);
// create DateTime object
    DateTime currenttime = DateTime.parse(dateTime);
    currenttime = currenttime.add(
        Duration(minutes:
        int.parse(offsetMinutes),hours:int.parse(offsetHours)));
//set the time property of class...
    time = currenttime.toString();
  }
}
```

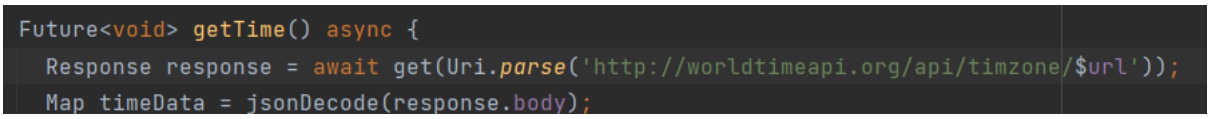
a



Time API

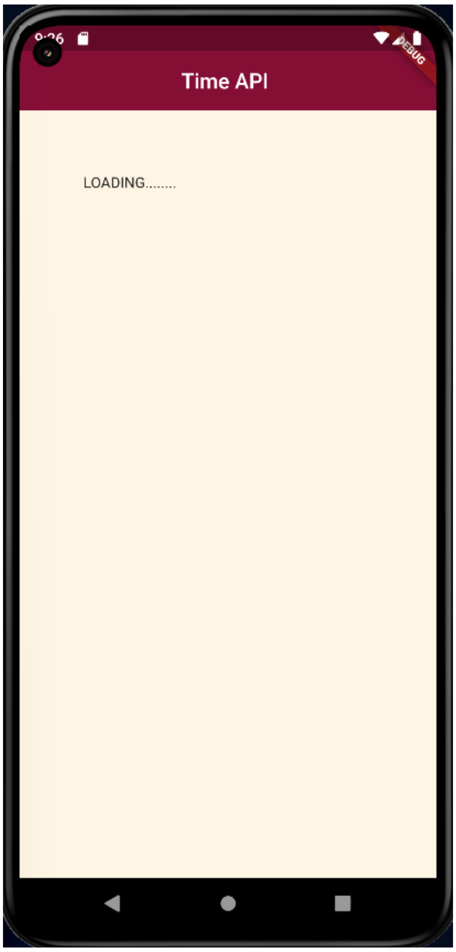
2022-09-26 21:21:21.057264Z

Now we will add error handling. For that first of all make some intentionality mistake. i.e. Remove one character from API string

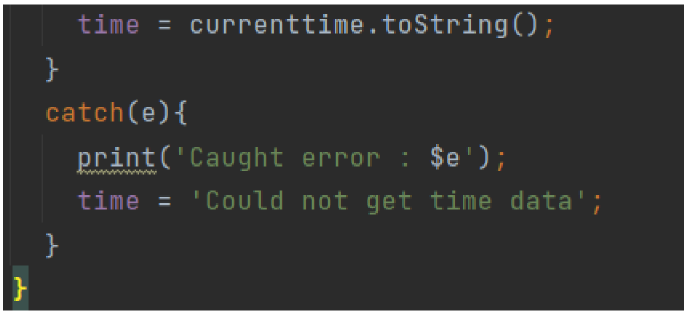


```
Future<void> getTime() async {
  Response response = await get(Uri.parse('http://worldtimeapi.org/api/timzone/$url'));
  Map timeData = jsonDecode(response.body);
```

Then output will be as shown below. Here in output we can see that loading is printing. So we need to do something that will display error accordingly. Here on error we are setting time value to some appropriate message. So that it can be displayed there.

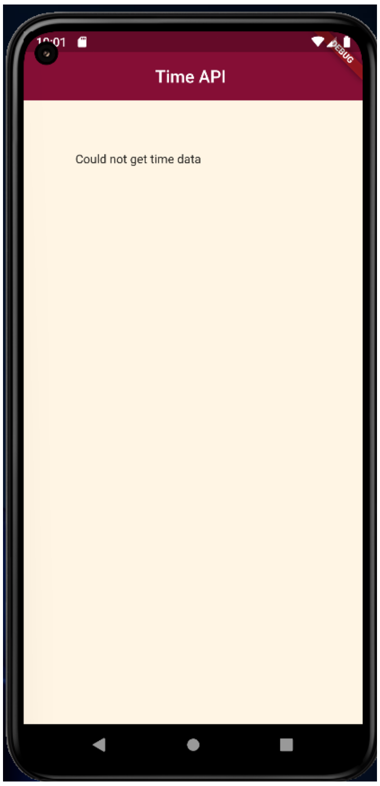


LOADING........



```
    time = currenttime.toString();
  }
  catch(e){
    print('Caught error : $e');
    time = 'Could not get time data';
  }
}
```

So if there is some issue in fetching time then we can set time value as shown above. So updated output would be …

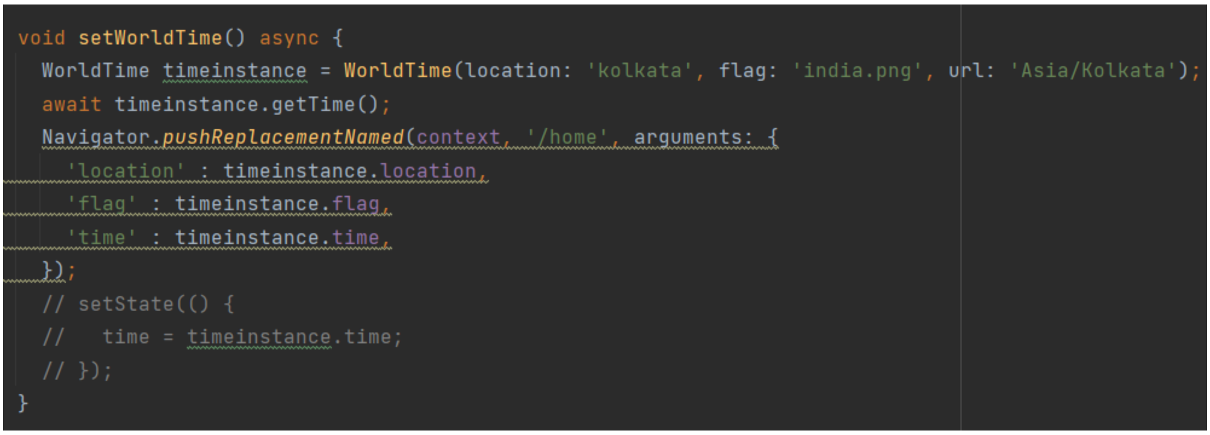


Could not get time data

Code test 2: Data passing between route

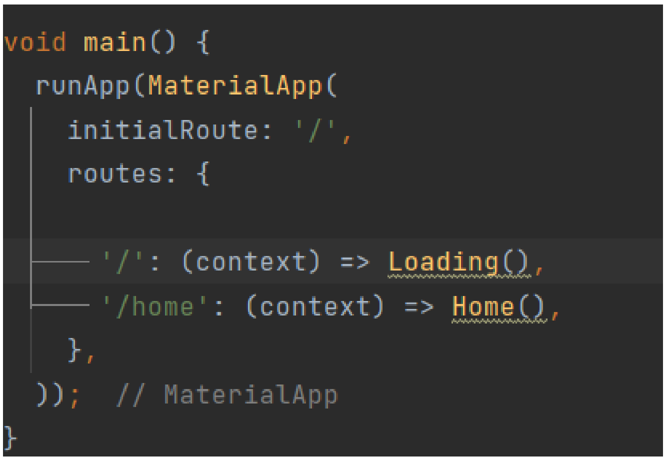
Code test 3: Formatting and showing data—dates

Do some modifications in loading.dart file as shown below.



```dart
void setWorldTime() async {
  WorldTime timeinstance = WorldTime(location: 'kolkata', flag: 'india.png', url: 'Asia/Kolkata');
  await timeinstance.getTime();
  Navigator.pushReplacementNamed(context, '/home', arguments: {
    'location' : timeinstance.location,
    'flag' : timeinstance.flag,
    'time' : timeinstance.time,
  });
  // setState(() {
  //   time = timeinstance.time;
  // });
}
```

Here we are using routing with passing some additional arguments into it. So in home.dart file we can access location, flag and time field.

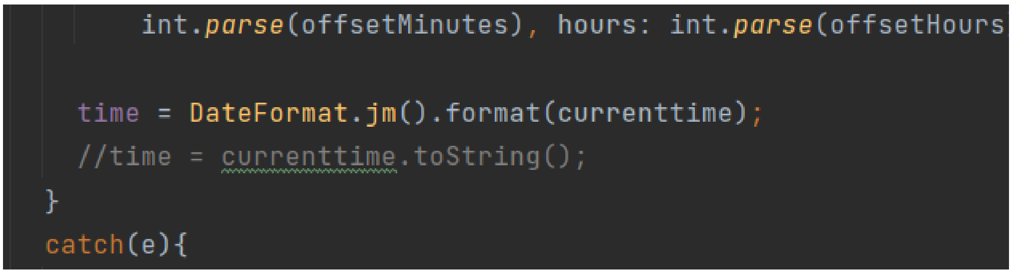


```dart
void main() {
  runApp(MaterialApp(
    initialRoute: '/',
    routes: {

      '/': (context) => Loading(),
      '/home': (context) => Home(),
    },
  )); // MaterialApp
}
```

Basic routing

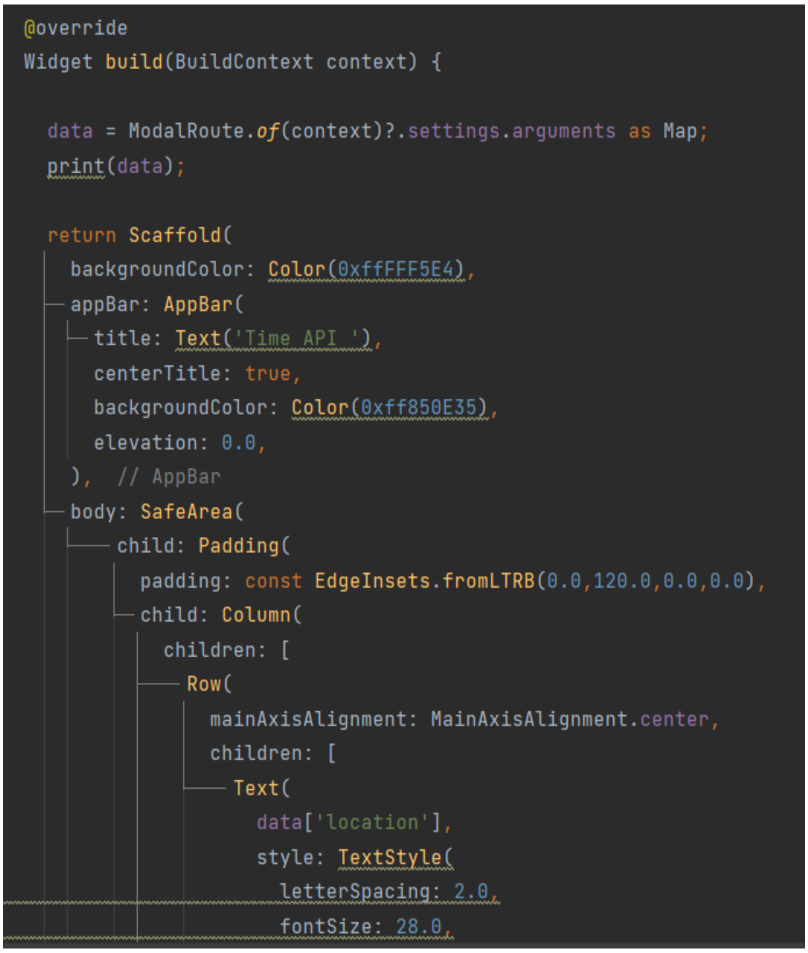
Now we will perform some formatting steps using intl. For that we need to add it into pubspec.yaml

Import intl in time.dart file and change accordingly that file.

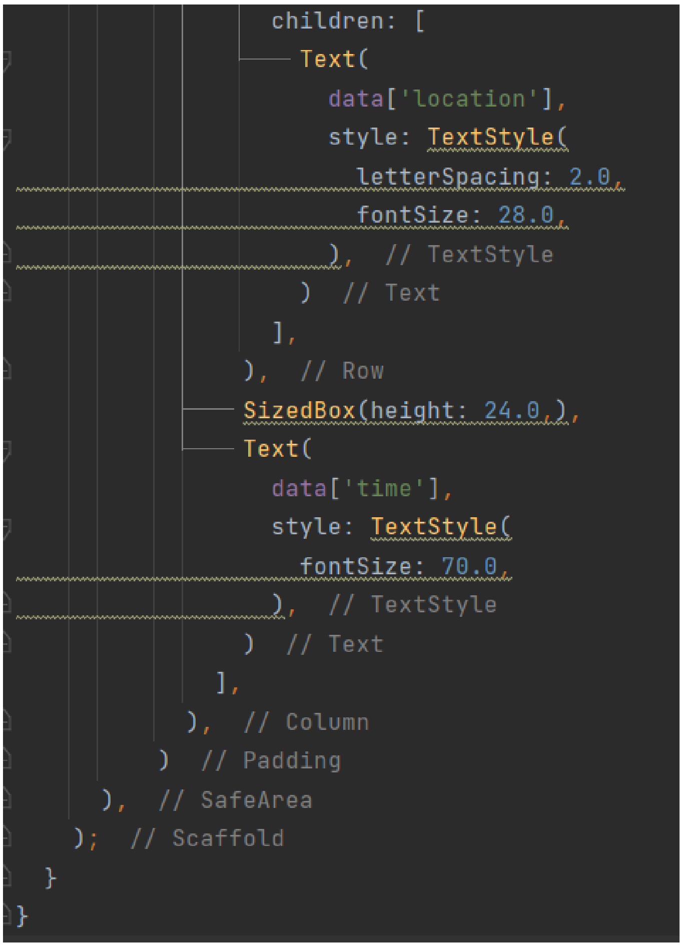


```
        int.parse(offsetMinutes), hours: int.parse(offsetHours

    time = DateFormat.jm().format(currenttime);
    //time = currenttime.toString();
  }
  catch(e){
```

Now home.dart file would contains below code.

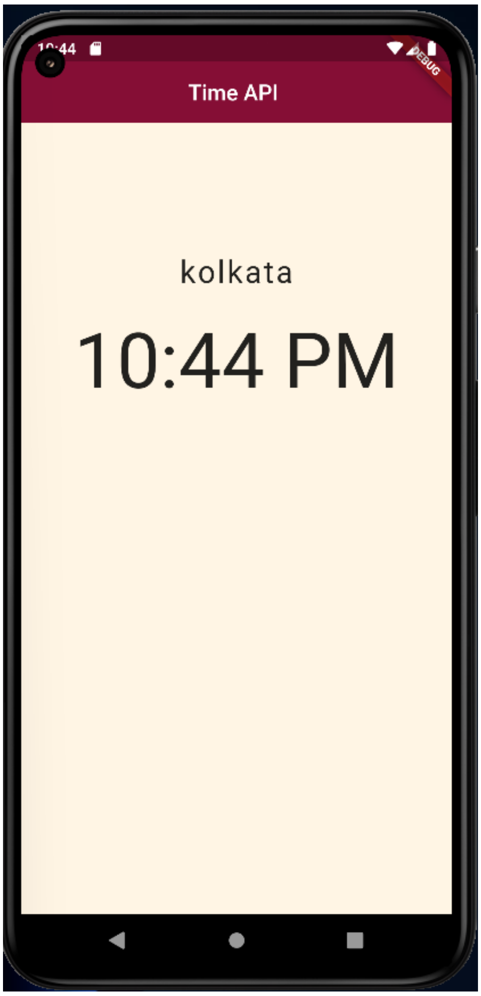


```
@override
Widget build(BuildContext context) {

  data = ModalRoute.of(context)?.settings.arguments as Map;
  print(data);

  return Scaffold(
    backgroundColor: Color(0xffFFF5E4),
    appBar: AppBar(
      title: Text('Time API '),
      centerTitle: true,
      backgroundColor: Color(0xff850E35),
      elevation: 0.0,
    ), // AppBar
    body: SafeArea(
        child: Padding(
          padding: const EdgeInsets.fromLTRB(0.0,120.0,0.0,0.0),
          child: Column(
            children: [
              Row(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
                  Text(
                    data['location'],
                    style: TextStyle(
                      letterSpacing: 2.0,
                      fontSize: 28.0,
```



```
                children: [
                  Text(
                    data['location'],
                    style: TextStyle(
                      letterSpacing: 2.0,
                      fontSize: 28.0,
                    ), // TextStyle
                  ) // Text
                ],
              ), // Row
              SizedBox(height: 24.0,),
              Text(
                data['time'],
                style: TextStyle(
                  fontSize: 70.0,
                ), // TextStyle
              ) // Text
            ],
          ), // Column
        ) // Padding
    ), // SafeArea
  ); // Scaffold
}
}
```

Final Output:



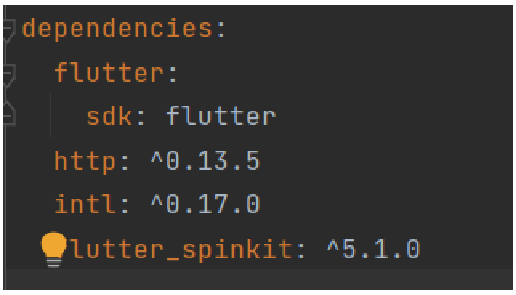
Time API

kolkata

10:44 PM

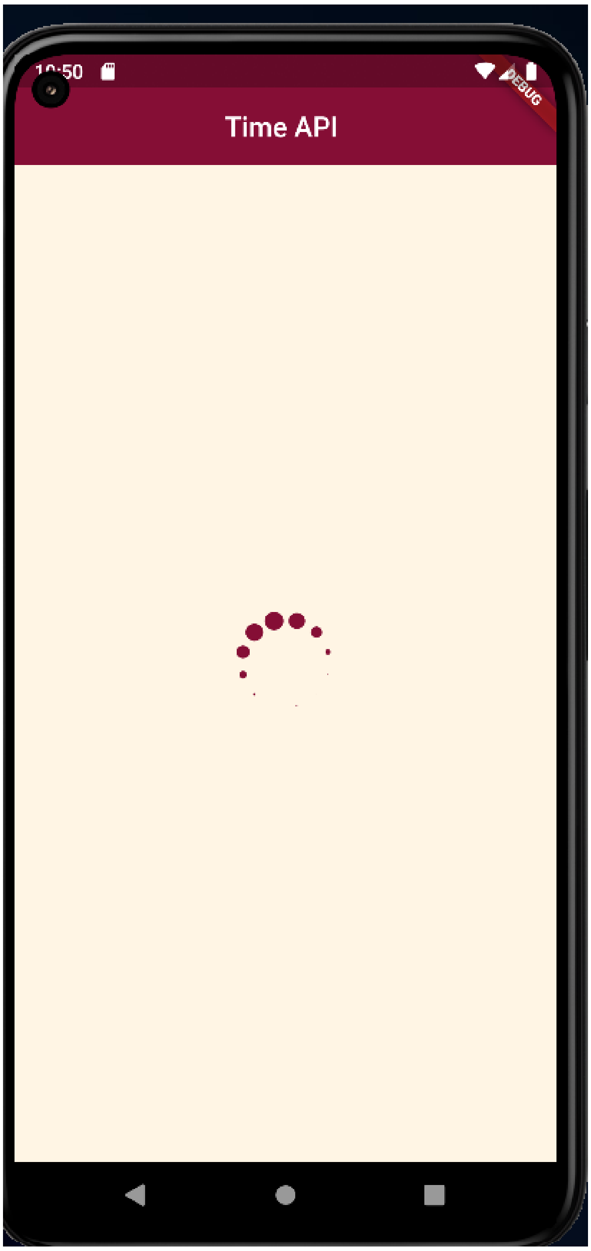
Code test 4: Using spinners for loading page

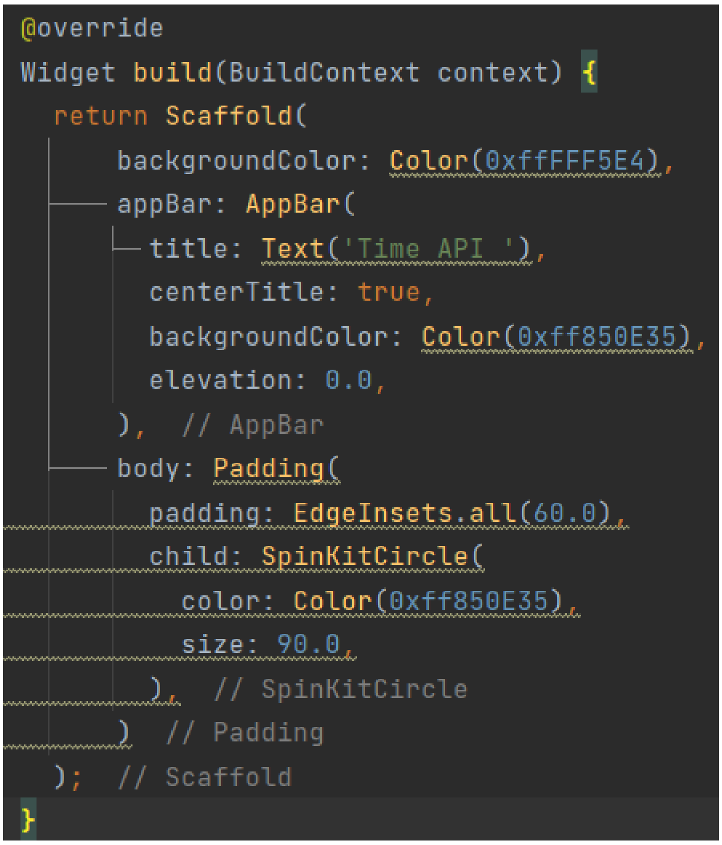
Add flutter spinkit into pubspec.yaml file and import it into loading file.



```
dependencies:
  flutter:
    sdk: flutter
  http: ^0.13.5
  intl: ^0.17.0
  lutter_spinkit: ^5.1.0
```

Now we will use flutter spinner for loading and after data fetched current time and location will be displayed.





```
@override
Widget build(BuildContext context) {
  return Scaffold(
      backgroundColor: Color(0xffFFF5E4),
      appBar: AppBar(
        title: Text('Time API '),
        centerTitle: true,
        backgroundColor: Color(0xff850E35),
        elevation: 0.0,
      ),  // AppBar
      body: Padding(
        padding: EdgeInsets.all(60.0),
        child: SpinKitCircle(
          color: Color(0xff850E35),
          size: 90.0,
        ),  // SpinKitCircle
      )  // Padding
  );  // Scaffold
}
```

GitHub repository link:

https://github.com/DhruviSherathiya/SmartDeviceProgramming/tree/main/Lab11