

Name: Varun Khubani

Division: D15A

Roll No: 30

Batch: B

Experiment No 6

Aim: To Connect Flutter UI with Firebase

Theory:

FlutterFire is a set of Flutter plugins that enable Flutter developers to integrate their applications with various Firebase services. Firebase is a comprehensive mobile and web application development platform provided by Google. FlutterFire is specifically designed to provide Flutter developers with a seamless way to interact with Firebase services.

Key features of FlutterFire include:

1. **Firebase Authentication:** FlutterFire provides plugins to easily integrate Firebase Authentication, allowing developers to implement user sign-up, sign-in, and password recovery features in their Flutter applications. Firebase supports various authentication methods, including email/password, Google Sign-In, Facebook Sign-In, and more.
2. **Cloud Firestore and Realtime Database:** FlutterFire supports both Cloud Firestore and Firebase Realtime Database, enabling developers to store and retrieve data in real-time. Firestore is a NoSQL document database, while Realtime Database is a JSON-based database.
3. **Cloud Functions:** Developers can deploy serverless functions using Cloud Functions for Firebase, and FlutterFire allows Flutter apps to trigger and interact with these functions.
4. **Cloud Storage:** FlutterFire supports Firebase Cloud Storage, allowing developers to upload, download, and manage files in the cloud. This is useful for handling user-generated content, such as images or videos.
5. **Firebase Cloud Messaging (FCM):** FCM enables developers to send push notifications to their Flutter applications. FlutterFire provides plugins for integrating FCM and handling push notifications.
6. **Firebase Performance Monitoring:** Developers can monitor the performance of their Flutter applications using Firebase Performance Monitoring. This includes measuring app

startup time, screen rendering, and network performance.

7. Firebase Analytics: FlutterFire includes plugins for integrating Firebase Analytics, enabling developers to gain insights into user behavior and app usage.
8. Firebase Remote Config: FlutterFire supports Firebase Remote Config, allowing developers to remotely configure app behavior without publishing updates. This is useful for A/B testing and feature toggling.
9. Firebase Crashlytics: FlutterFire includes support for Firebase Crashlytics, providing real-time crash reporting to help developers identify and fix issues quickly.
10. Firebase AdMob: FlutterFire includes AdMob plugins for integrating advertisements into Flutter applications using Firebase AdMob.

Firebase API code:

```
import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
import 'package:flutter/foundation.dart'
    show defaultTargetPlatform, kIsWeb, TargetPlatform;

/// Default [FirebaseOptions] for use with your Firebase apps.
///
/// Example:
/// ```dart
/// import 'firebase_options.dart';
/// // ...
/// await Firebase.initializeApp(
///   options: DefaultFirebaseOptions.currentPlatform,
/// );
/// ```
class DefaultFirebaseOptions {
  static FirebaseOptions get currentPlatform {
    if (kIsWeb) {
      return web;
    }
    switch (defaultTargetPlatform) {
      case TargetPlatform.android:
        return android;
      case TargetPlatform.iOS:
        return ios;
      case TargetPlatform.macos:
        return macos;
      case TargetPlatform.windows:
        throw UnsupportedError(
          'DefaultFirebaseOptions have not been configured for windows - '
          'you can reconfigure this by running the FlutterFire CLI again.',
        );
      case TargetPlatform.linux:
        throw UnsupportedError(
          'DefaultFirebaseOptions have not been configured for linux - '
          'you can reconfigure this by running the FlutterFire CLI again.',
        );
    }
  }
}
```

```
        default:
            throw UnsupportedError(
                'DefaultFirebaseOptions are not supported for this platform.',
            );
    }
}
```

```
static const FirebaseOptions web = FirebaseOptions(
    apiKey: 'AIzaSyDIVnkIATlnZWfZ-wyY0XdicW-5fxd-KQQ',
    appId: '1:288829673555:web:32b40b9fd993005f8c36a6',
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    authDomain: 'nike-shoes-ee9a3.firebaseio.com',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
    measurementId: 'G-JC4VTW0LP2',
);
```

```
static const FirebaseOptions android = FirebaseOptions(
    apiKey: 'AIzaSyCoMsxqQXu7PuHbBgz9gkWSkTcIrYR8s4M',
    appId: '1:288829673555:android:add9916c7fd73ee58c36a6',
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
);
```

```
static const FirebaseOptions ios = FirebaseOptions(
    apiKey: 'AIzaSyAAJ2iFZJycPwx-CW7GAVWa4bxQcyqpyqI',
    appId: '1:288829673555:ios:8bd54ee07f4b8a948c36a6',
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
    iosBundleId: 'com.example.kickBazaarApp',
);
```

```
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
  );

  static const FirebaseOptions ios = FirebaseOptions(
    apiKey: 'AIzaSyAAJ2iFZJycPWx-CW7GAVWa4bxQcyqpyqI',
    appId: '1:288829673555:ios:8bd54ee07f4b8a948c36a6',
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
    iosBundleId: 'com.example.kickBazaarApp',
  );

  static const FirebaseOptions macos = FirebaseOptions(
    apiKey: 'AIzaSyAAJ2iFZJycPWx-CW7GAVWa4bxQcyqpyqI',
    appId: '1:288829673555:ios:8b74ef2fa6b4e4e68c36a6',
    messagingSenderId: '288829673555',
    projectId: 'nike-shoes-ee9a3',
    storageBucket: 'nike-shoes-ee9a3.appspot.com',
    iosBundleId: 'com.example.kickBazaarApp.RunnerTests',
  );
}
```

SignUp Page:

```
1 import 'package:firebase_auth/firebase_auth.dart';
2 import 'package:flutter/material.dart';
3 import 'package:kick_bazaar_app/main.dart';
4
5 class MySignup extends StatefulWidget {
6   const MySignup({Key? key}) : super(key: key);
7
8   @override
9   State<MySignup> createState() => _MySignupState();
10 }
11
12 class _MySignupState extends State<MySignup> {
13   TextEditingController _emailController = TextEditingController();
14
15   // TextEditingController _emailTextEditingController = TextEditingController();
16   TextEditingController passwordTextEditingController = TextEditingController();
17   TextEditingController phoneNoEditingController =
18     TextEditingController();
19   TextEditingController package:kick_bazaar_app/signup.dart ();
20   TextEditingController phoneNoEditingController = TextEditingController();
21   // TextEditingController cityTextEditingController = TextEditingController();
22   // TextEditingController countryTextEditingController = TextEditingController();
23   // TextEditingController profileHeadingTextEditingController =
24     TextEditingController();
25   //
26   //
27   // TextEditingController heightTextEditingController = TextEditingController();
28   // TextEditingController weightTextEditingController = TextEditingController();
29   // TextEditingController bodytypeTextEditingController = TextEditingController();
30   // TextEditingController drinkTextEditingController = TextEditingController();
31   // TextEditingController smokeTextEditingController = TextEditingController();
32   // TextEditingController haveChildrenTextEditingController =
33     TextEditingController();
34   // TextEditingController noOfChildrenTextEditingController =
35     TextEditingController();
36   // TextEditingController professionTextEditingController =
37     TextEditingController();
```

```

void createAccount() async {
  String email = _emailController.text.trim();
  String password = passwordTextEditingController.text.trim();
  String cpassword = confirmPasswordTextEditingController.text.trim();

  if (email == "" || password == "" ) {
    print('Please fill all details');
  } else {
    try {
      UserCredential userCredential = await FirebaseAuth.instance
        .createUserWithEmailAndPassword(email: email, password: password);

      if (userCredential.user != null) {
        Navigator.push(
          context,
          MaterialPageRoute(builder: (context) => NavigationPage()),
        );
      }

      print('Account Created Succesfully');
    } on FirebaseAuthException catch (ex) {
      print(ex.code.toString());
    }
  }
}

@override
Widget build(BuildContext context) {
  return Container(
    decoration: const BoxDecoration(
      image: DecorationImage(
        image: AssetImage(
          "/Users/varunkhubani/Downloads/Kick-Bazaar-App/assets/signup.png"), // AssetIma
        fit: BoxFit.cover), // DecorationImage // BoxDecoration
    child: Scaffold(
      appBar: AppBar(

```

```

Widget build(BuildContext context) {
  return Container(
    decoration: const BoxDecoration(
      image: DecorationImage(
        image: AssetImage(
          "/Users/varunkhubani/Downloads/Kick-Bazaar-App/assets/signup.png"), // AssetImage
        fit: BoxFit.cover), // DecorationImage // BoxDecoration
    child: Scaffold(
      appBar: AppBar(
        elevation: 0,
        backgroundColor: Colors.transparent,
      ), // AppBar
      backgroundColor: Colors.transparent,
      body: Stack(
        children: [
          Container(
            padding: const EdgeInsets.only(left: 35, top: 70),
            child: const Text(
              "Create Account",
              style: TextStyle(
                color: Colors.white,
                fontSize: 33.0,
              ), // TextStyle
            ), // Text
          ), // Container
          SingleChildScrollView(
            child: Container(
              padding: EdgeInsets.only(
                top: MediaQuery.of(context).size.height * 0.25,
                left: 30.0,
                right: 30.0), // EdgeInsets.only
              child: Column(
                children: <Widget>[
                  TextField(
                    controller: UsernameTextEditingController,

```



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

```
  TextField(  
    controller: UsernameTextEditingController,  
    decoration: InputDecoration(  
      fillColor: Colors.white10,  
      filled: true,  
      border: OutlineInputBorder(  
        borderRadius: BorderRadius.all(Radius.circular(30)),  
      ), // OutlineInputBorder  
      hintText: "Username",  
      hintStyle: TextStyle(fontSize: 18.0),  
    ), // InputDecoration  
  ), // TextField  
  const SizedBox(  
    height: 25.0,  
  ), // SizedBox  
  TextField(  
    controller: _emailController,  
    decoration: InputDecoration(  
      fillColor: Colors.white10,  
      filled: true,  
      border: OutlineInputBorder(  
        borderRadius: BorderRadius.all(Radius.circular(30)),  
      ), // OutlineInputBorder  
      hintText: "Email",  
      hintStyle: TextStyle(fontSize: 18.0),  
    ), // InputDecoration  
  ), // TextField  
  const SizedBox(  
    height: 25.0,  
  ), // SizedBox  
  TextField(  
    controller: passwordTextEditingController,  
    decoration: InputDecoration(  
      fillColor: Colors.white10,  
      filled: true,
```

```

class _MySignupState extends State<MySignup> {
  Widget build(BuildContext context) {
    // CircleAvatar
  },
), // Row
const SizedBox(
  height: 25.0,
), // SizedBox
Column(
  children: <Widget>[
    const Text("Already have an Account? ",
      style: TextStyle(
        fontWeight: FontWeight.bold, fontSize: 20.0)), // TextStyle // Text
    const SizedBox(
      height: 10,
    ), // SizedBox
    TextButton(
      style: TextButton.styleFrom(
        padding: const EdgeInsets.only(
          left: 20, top: 10, right: 20, bottom: 10), // EdgeInsets.only
        backgroundColor: Colors.black26,
        onPressed: () {
          Navigator.pop(context);
        },
        child: const Text(
          "Login",
          style: TextStyle(
            fontSize: 20.0,
            color: Colors.white,
            decoration: TextDecoration.underline), // TextStyle
        ), // Text
      ), // TextButton
    ], // <Widget>[]
  ), // Column
), // Column
), // Container

```

forgot Password Page:

```
class ForgetPassword extends StatefulWidget {
  const ForgetPassword({Key? key}) : super(key: key);

  @override
  State<ForgetPassword> createState() => _ForgetPasswordState();
}

class _ForgetPasswordState extends State<ForgetPassword> {
  @override
  Widget build(BuildContext context) {
    return Container(
      decoration: const BoxDecoration(
        image: DecorationImage(
          image: AssetImage("/Users/varunkhubani/Downloads/Kick-Bazaar-App/assets/signup.png"), fit: BoxFit.cover)),
      child: Scaffold(
        appBar: AppBar(
          elevation: 0,
          backgroundColor: Colors.transparent,
        ), // AppBar
        backgroundColor: Colors.transparent,
        body: Stack(
          children: [
            Container(
              padding: const EdgeInsets.only(left: 35, top: 90),
              child: const Text(
                "Reset Password",
                style: TextStyle(
                  color: Colors.white,
                  fontSize: 33.0,
                ), // TextStyle
              ), // Text
            ), // Container
            Container(
              padding: const EdgeInsets.only(left: 35, top: 180),
              child: const Text(
```

```

Widget build(BuildContext context) {
  ), // Container
  Container(
    padding: const EdgeInsets.only(left: 35, top: 180),
    child: Text(
      "Enter the E-mail associated with your account and we will send you the instruction to reset your Password",
      style: TextStyle(
        color: Colors.black54,
        fontSize: 20.0,
        // fontStyle: FontStyle.italic,
        fontWeight: FontWeight.w400), // TextStyle
    ), // Text
  ), // Container
  SingleChildScrollView(
    child: Container(
      padding: EdgeInsets.only(
        top: MediaQuery.of(context).size.height * 0.4,
        left: 30.0,
        right: 30.0), // EdgeInsets.only
      child: Column(
        children: <Widget>[
          const TextField(
            decoration: InputDecoration(
              fillColor: Colors.white10,
              filled: true,
              border: OutlineInputBorder(
                borderRadius: BorderRadius.all(Radius.circular(30)),
              ), // OutlineInputBorder
              hintText: "Your Email",
              hintStyle: TextStyle(fontSize: 18.0),
            ), // InputDecoration
          ), // TextField
          const SizedBox(
            height: 25.0,
          ), // SizedBox
          Row(

```

login page:

```
login.dart / ...
// import 'package:day6_login_signup_page/dashboard.dart';
// import 'package:day6_login_signup_page/forgetpass.dart';
// import 'package:day6_login_signup_page/signup.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:kick_bazaar_app/main.dart';
import 'package:kick_bazaar_app/pages/sneaker_page.dart';

// import './dashboard.dart';
import './pages/sneaker_page.dart';
import './forgetpass.dart';
import './signup.dart';
import 'package:flutter/material.dart';

class MyLogin extends StatefulWidget {
  const MyLogin({Key? key}) : super(key: key);

  @override
  State<MyLogin> createState() => _MyLoginState();
}

class _MyLoginState extends State<MyLogin> {
  TextEditingController emailTextEditingController = TextEditingController();
  TextEditingController passwordTextEditingController = TextEditingController();
  bool showProgressBar = false;
  // final FirebaseFirestore _firestore = FirebaseFirestore.instance;

  //function for login with email and password
  void login() async {
    String email = emailTextEditingController.text.trim();
    String password = passwordTextEditingController.text.trim();

    if (email == "" || password == "") {
      print('please fill all details');
    } else {
      try {
        UserCredential userCredential = await FirebaseAuth.instance
          .signInWithEmailAndPassword(email: email, password: password);
      }
    }
  }
}
```

```

class _MyLoginState extends State<MyLogin> {
  void login() async {
    UserCredential userCredential = await FirebaseAuth.instance
      .signInWithEmailAndPassword(email: email, password: password);

    if (userCredential != null) {
      Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => NavigationPage()),
      );
    }
  } on FirebaseAuthException catch (ex) {
    print(ex.code.toString());
  }
}

@override
Widget build(BuildContext context) {
  return Container(
    decoration: const BoxDecoration(
      image: DecorationImage(
        image: AssetImage(
          "/Users/varunkhubani/Downloads/Kick-Bazaar-App/assets/login.png"), // AssetImage
        fit: BoxFit.cover), // DecorationImage // BoxDecoration
    child: Scaffold(
      backgroundColor: Colors.transparent,
      body: Stack(
        children: [
          Container(
            padding: const EdgeInsets.only(left: 35, top: 180),
            child: const Text(
              "Welcome Back!",
              style: TextStyle(
                color: Colors.white,
                fontSize: 33.0,
              ), // TextStyle

```

```

class _MyLoginState extends State<MyLogin> {
  Widget build(BuildContext context) {
    // BoxDecoration // BoxDecoration
    child: Scaffold(
      backgroundColor: Colors.transparent,
      body: Stack(
        children: [
          Container(
            padding: const EdgeInsets.only(left: 35, top: 180),
            child: const Text(
              "Welcome Back!",
              style: TextStyle(
                color: Colors.white,
                fontSize: 33.0,
              ), // TextStyle
            ), // Text
          ), // Container
          SingleChildScrollView(
            child: Container(
              padding: EdgeInsets.only(
                top: MediaQuery.of(context).size.height * 0.5,
                left: 30.0,
                right: 30.0), // EdgeInsets.only
              child: Column(
                children: [
                  TextField(
                    controller: emailTextEditingController,
                    decoration: InputDecoration(
                      fillColor: Colors.black12,
                      filled: true,
                      hintText: "Your Email",
                      hintStyle: const TextStyle(fontSize: 18.0),
                      border: OutlineInputBorder(
                        borderSide:
                          const BorderSide(width: 3, color: Colors.blue),
                        borderRadius: BorderRadius.circular(35),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ),
                ],
              ),
            ),
          ),
        ],
      ),
    ),
  ),
}

```

```

Widget build(BuildContext context) {
  const SizedBox(
    height: 30.0,
  ), // SizedBox
  TextField(
    controller: passwordTextEditingController,
    obscureText: true,
    decoration: InputDecoration(
      fillColor: Colors.black12,
      filled: true,
      hintText: "Your Password",
      hintStyle: const TextStyle(fontSize: 18.0),
      border: OutlineInputBorder(
        borderSide:
          const BorderSide(width: 3, color: Colors.blue),
        borderRadius: BorderRadius.circular(35),
      ), // OutlineInputBorder
    ), // InputDecoration
  ), // TextField
  const SizedBox(
    height: 30.0,
  ), // SizedBox
  Row(
    mainAxisAlignment: MainAxisAlignment.end,
    children: [
      // const Text(
      //   "Sign In",
      //   style: TextStyle(
      //     fontSize: 25, fontWeight: FontWeight.w700),
      // ),
      CircleAvatar(
        backgroundColor: Colors.black45,
        radius: 25.0,
        child: IconButton(
          icon: const Icon(Icons.login),
          color: Colors.white,

```



```

Widget build(BuildContext context) {
  ), // Row
  const SizedBox(
    height: 5.0,
  ), // SizedBox
  Column(
    mainAxisAlignment: MainAxisAlignment.spaceBetween,
    children: [
      Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          const Text("Not a Member?",
            style: TextStyle(
              fontWeight: FontWeight.bold,
              fontSize: 20.0)), // TextStyle // Text
          TextButton(
            onPressed: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) =>
                    const MySignup()), // MaterialPageRoute
              ),
            child: const Text(
              "Sign Up",
              style: TextStyle(
                decoration: TextDecoration.underline,
                fontSize: 20.0), // TextStyle
            )), // Text // TextButton
        ],
      ), // Row
      TextButton(
        onPressed: () {
          Navigator.push(
            context,
            MaterialPageRoute(

```

```

Widget build(BuildContext context) {
    ), // Text // TextButton
  ],
), // Row
TextButton(
  onPressed: () {
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) =>
          const ForgetPassword()), // MaterialPageRoute
      ),
    child: const Text(
      "Forget Password?",
      style: TextStyle(
        color: Colors.black54,
        decoration: TextDecoration.underline,
        fontSize: 20.0), // TextStyle
    )) // Text // TextButton
  ],
) // Column
),
), // Column
), // Container
) // SingleChildScrollView
],
), // Stack
), // Scaffold
); // Container
}
}

```

```

Widget build(BuildContext context) {
    duration: Duration(
      seconds:
        2), // Adjust the duration as needed // Duration
    ), // SnackBar
  );
  Future.delayed(const Duration(seconds: 1), () {
    Navigator.pop(context);
  }); // Future.delayed
},
), // ElevatedButton
) // SizedBox
: Padding(
  padding: EdgeInsets.only(
    top: MediaQuery.of(context).size.width * 0.74), // EdgeInsets.only
  child: const Center(
    child: Text(
      "Your basket is empty.",
      style: TextStyle(
        fontSize: 20, fontWeight: FontWeight.bold), // TextStyle
    ), // Text
  ), // Center
), // Padding // Padding
1,
), // Column
), // SingleChildScrollView
); // Scaffold
}
}

```

```

Widget build(BuildContext context) {
  ), // ListView.builder
  Padding(
    padding: const EdgeInsets.only(top: 20, bottom: 20),
    child: basket.isNotEmpty
      ? SizedBox(
        width: 140,
        child: ElevatedButton(
          style: ElevatedButton.styleFrom(
            backgroundColor: Colors.black),
          child: const Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              Icon(Icons.payment),
              Text("Checkout"),
            ],
          ), // Row
          onPressed: () {
            basket.clear();
            ScaffoldMessenger.of(context).showSnackBar(
              const SnackBar(
                backgroundColor: Colors.green,
                content: Text(
                  'Order placed successfully!',
                  style: TextStyle(
                    fontWeight: FontWeight.bold,
                    fontSize: 17), // TextStyle
                ), // Text
                duration: Duration(
                  seconds:
                    2), // Adjust the duration as needed // Duration
              ), // SnackBar
            );
            Future.delayed(const Duration(seconds: 1), () {
              Navigator.pop(context);
            }); // Future.delayed

```

checkoutPage

```
class _CheckoutPageState extends State<CheckoutPage> {
  Widget build(BuildContext context) {
    child: Column(
      children: [
        ListView.builder(
          physics: const BouncingScrollPhysics(),
          shrinkWrap: true,
          padding: const EdgeInsets.only(top: 15),
          itemCount: basket.length,
          itemBuilder: (context, int index) {
            return Container(
              decoration: const BoxDecoration(
                border: Border(
                  bottom: BorderSide(width: 1, color: Colors.grey)), // Border // BoxDecoration
              width: MediaQuery.of(context).size.width,
              height: 100,
              child: Padding(
                padding: const EdgeInsets.all(5),
                child: Row(
                  children: [
                    Image.asset(basket[index][2]),
                    Expanded(
                      child: Row(
                        children: [
                          Expanded(
                            child: Column(
                              mainAxisAlignment: MainAxisAlignment.center,
                              children: [
                                Text(
                                  basket[index][0],
                                  // ignore: prefer_const_constructors
                                  style: TextStyle(
                                    fontWeight: FontWeight.bold,
                                    fontSize: 16), // TextStyle
                                ), // Text
                                Text(
                                  basket[index][1],
```

```

import 'package:flutter/material.dart';

import 'lists.dart';

class CheckoutPage extends StatefulWidget {
  const CheckoutPage({Key? key}) : super(key: key);

  @override
  State<CheckoutPage> createState() => _CheckoutPageState();
}

class _CheckoutPageState extends State<CheckoutPage> {
  @override
  Widget build(BuildContext context) {
    ⚡ return Scaffold(
      appBar: AppBar(
        leading: IconButton(
          onPressed: () {
            Navigator.pop(context);
          },
          icon: const Icon(
            Icons.arrow_back_ios_new,
          ), // Icon
        ), // IconButton
        centerTitle: true,
        backgroundColor: Colors.black,
        title: const Text(
          "Kick Bazaar Basket",
          style: TextStyle(fontSize: 21),
        ), // Text
      ), // AppBar
      body: SingleChildScrollView(
        physics: const BouncingScrollPhysics(),
        child: Column(
          children: [
            ListView.builder(
              physics: const BouncingScrollPhysics(),

```

```

Widget build(BuildContext context) {
  text: 'size: ${sizes[selectedIndex]}',
  style: const TextStyle(color: Colors.black54),
), // TextSpan
],
), // TextSpan
), // RichText
duration: const Duration(
  seconds: 1), // Adjust the duration as needed // Duration
), // SnackBar
);
Future.delayed(const Duration(seconds: 1), () {
  Navigator.pop(context);
}); // Future.delayed
},
child: const Row(
  mainAxisAlignment: MainAxisAlignment.spaceBetween,
  children: [
    Icon(Icons.add_shopping_cart_rounded),
    Text("Add to Basket"),
  ],
), // Row
), // ElevatedButton
), // SizedBox
]), // Column
), // SingleChildScrollView
); // Scaffold
}
}

```

SneakerPage

```
class _SneakerPageState extends State<SneakerPage> {
  Widget build(BuildContext context) {
    SizedBox(
      height: 70,
      child: ListView.builder(
        physics: const BouncingScrollPhysics(),
        scrollDirection: Axis.horizontal,
        itemCount: sizes.length,
        itemBuilder: (context, int index) => Padding(
          padding: const EdgeInsets.only(left: 5),
          child: GestureDetector(
            onTap: () {
              setState(() {
                selectedIndex = index;
              });
            },
            child: CircleAvatar(
              backgroundColor:
                index == selectedIndex ? Colors.red : Colors.black,
              radius: 25,
              child: Text(
                sizes[index].toString(),
                style: const TextStyle(color: Colors.white),
              ), // Text
            ), // CircleAvatar
          ), // GestureDetector
        ), // Padding
      ), // ListView.builder
    ), // SizedBox
    const Divider(),
    Text(
      "₹ ${widget.price}",
      style:
        GoogleFonts.prompt(fontSize: 24, fontWeight: FontWeight.w500),
    ), // Text
    SizedBox(
      width: 150,
```



```
class _SneakPageState<SneakerPage> {  
  Widget build(BuildContext context) {  
  
    GoogleFonts.prompt(fontSize: 28, fontWeight: FontWeight.bold),  
  ), // Text  
) , // Align  
const Divider(),  
Padding(  
  padding: const EdgeInsets.only(left: 12, right: 12),  
  child: Text(  
    widget.description,  
    textAlign: TextAlign.start,  
    style: GoogleFonts.prompt(  
      fontSize: 16, fontWeight: FontWeight.w400),  
    )), // Text // Padding  
const Divider(),  
const Align(  
  alignment: Alignment.topLeft,  
  child: Padding(  
    padding: EdgeInsets.symmetric(horizontal: 13),  
    child: Text(  
      "Select your size :",  
      style: TextStyle(fontWeight: FontWeight.bold),  
    ), // Text  
  ), // Padding  
) , // Align  
// ListTile(  
//   leading: const Icon(Icons.logout),  
//   title: const Text("Logout"),  
//   onTap: () {  
//     Navigator.push(context, MaterialPageRoute(builder: (context) => const MyLogin()));  
//   },  
//   ),  
 SizedBox(  
  height: 70,  
  child: ListView.builder(  
    physics: const BouncingScrollPhysics(),  
    scrollDirection: Axis.horizontal,
```

```

class _SneakerPageState extends State<SneakerPage> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: SingleChildScrollView(
        physics: const BouncingScrollPhysics(),
        child: Column(children: [
          Padding(
            padding: const EdgeInsets.fromLTRB(10, 50, 5, 15),
            child: Container(
              decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(20),
                color: Colors.grey[200], // BoxDecoration
              ),
              width: MediaQuery.of(context).size.width,
              child: Image.asset(widget.imageUrl),
            ), // Container
          ), // Padding
          Align(
            alignment: Alignment.topCenter,
            child: Text(
              "${widget.brand} ${widget.model}",
              style:
                GoogleFonts.prompt(fontSize: 28, fontWeight: FontWeight.bold),
            ), // Text
          ), // Align
          const Divider(),
          Padding(
            padding: const EdgeInsets.only(left: 12, right: 12),
            child: Text(
              widget.description,
              textAlign: TextAlign.start,
              style: GoogleFonts.prompt(

```

```

import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';

import 'functions.dart';
import ' ../login.dart';

class SneakerPage extends StatefulWidget {
  const SneakerPage(
    {super.key,
    required this.index,
    required this.brand,
    required this.description,
    required this.model,
    required this.price,
    required this.imageUrl,
    required this.sneakerList});
  final int index;
  final String brand;
  final String model;
  final String description;
  final int price;
  final String imageUrl;
  final List sneakerList;

  @override
  State<SneakerPage> createState() => _SneakerPageState();
}

bool addedToBasket = false;
int selectedIndex = 4;
List<double> sizes = [
  7.5,
  8.0,
  8.5,
  9.0,
  9.5,
  10.0,

```

Output:

3:40

<

Create Account

Username

Email

Password

➔

Already have an Account?

Login

3:39



Welcome Back!



Not a Member? [Sign Up](#)

[Forget Password?](#)

3:45



Nike Shoes



Jordans

Nike
Air Jordans



The Original Jumpman

Air Force 1

Nike
Air Force 1



The King Of Streetwear

Dunks

3:45




Nike Air Jordans

Follow in the footsteps of greatness. The all-time classic AJ1 design gets a makeover with fresh pops of colour and mixed materials.

Select your size :



₹ 9000

 Add to Bag

RIGHT OVERFLOWED BY 17 PIXELS

3:45



Kick Bazaar Basket



Nike
Air Jordans

Size: 9.5
Price: €9000

 Checkout