Flutter mode

1. Develop a welcome screen for a shopping app using text, image, and button widgets.

import 'package:flutter/material.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: WelcomeScreen(),

);

}

}

class WelcomeScreen extends StatelessWidget {

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Shopping App'),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

'Welcome to ShopEasy!',

style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),

),

SizedBox(height: 20),

Image.network(

'https://via.placeholder.com/150', // Sample image from internet

height: 150,

),

SizedBox(height: 20),

ElevatedButton(

onPressed: () {

// You can add navigation here later

},

child: Text('Get Started'),

),

],

),

),

);

}

}

1. Build a Library Management App to add new books to the library and to show available vs. issued books using Icons

import 'package:flutter/material.dart';

void main() {

runApp(LibraryApp());

}

class LibraryApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Library Management',

home: LibraryHome(),

debugShowCheckedModeBanner: false,

);

}

}

class LibraryHome extends StatefulWidget {

@override

\_LibraryHomeState createState() => \_LibraryHomeState();

}

class \_LibraryHomeState extends State<LibraryHome> {

List<Book> books = [];

final TextEditingController titleController = TextEditingController();

final TextEditingController authorController = TextEditingController();

void \_addBook() {

showDialog(

context: context,

builder: (ctx) {

return AlertDialog(

title: Text('Add New Book'),

content: Column(

mainAxisSize: MainAxisSize.min,

children: [

TextField(

controller: titleController,

decoration: InputDecoration(labelText: 'Book Title'),

),

TextField(

controller: authorController,

decoration: InputDecoration(labelText: 'Author'),

),

],

),

actions: [

TextButton(

onPressed: () {

if (titleController.text.isNotEmpty && authorController.text.isNotEmpty) {

setState(() {

books.add(Book(

title: titleController.text,

author: authorController.text,

isIssued: false,

));

titleController.clear();

authorController.clear();

});

Navigator.of(ctx).pop();

}

},

child: Text('Add'),

),

],

);

},

);

}

void \_toggleIssued(int index) {

setState(() {

books[index].isIssued = !books[index].isIssued;

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Library Management'),

backgroundColor: Colors.deepPurple,

),

body: books.isEmpty

? Center(

child: Text(

'No books added yet!',

style: TextStyle(fontSize: 18),

),

)

: ListView.builder(

itemCount: books.length,

itemBuilder: (ctx, index) {

final book = books[index];

return Card(

elevation: 3,

margin: EdgeInsets.symmetric(horizontal: 10, vertical: 5),

child: ListTile(

leading: Icon(

book.isIssued ? Icons.close : Icons.check\_circle,

color: book.isIssued ? Colors.red : Colors.green,

),

title: Text(book.title),

subtitle: Text('by ${book.author}'),

trailing: IconButton(

icon: Icon(Icons.sync\_alt),

onPressed: () => \_toggleIssued(index),

tooltip: 'Toggle Issued/Available',

),

),

);

},

),

floatingActionButton: FloatingActionButton(

onPressed: \_addBook,

child: Icon(Icons.add),

backgroundColor: Colors.deepPurple,

),

);

}

}

class Book {

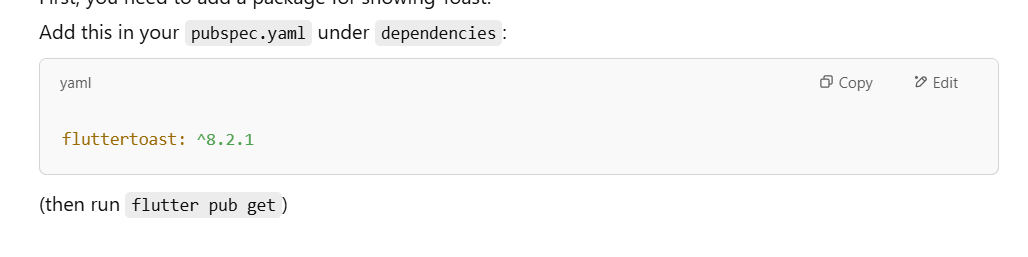
String title;

String author;

bool isIssued;

Book({required this.title, required this.author, this.isIssued = false});

}

1. Develop a login interface that shows a welcome toast after successful validation and invalid credentials show a "Login Failed" toast message. 

import 'package:flutter/material.dart';

import 'package:fluttertoast/fluttertoast.dart'; // Import fluttertoast

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: LoginScreen(),

);

}

}

class LoginScreen extends StatefulWidget {

@override

\_LoginScreenState createState() => \_LoginScreenState();

}

class \_LoginScreenState extends State<LoginScreen> {

TextEditingController usernameController = TextEditingController();

TextEditingController passwordController = TextEditingController();

void login() {

String username = usernameController.text;

String password = passwordController.text;

if (username == 'admin' && password == '1234') {

Fluttertoast.showToast(msg: "Welcome, $username!");

} else {

Fluttertoast.showToast(msg: "Login Failed");

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Login Page'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

TextField(

controller: usernameController,

decoration: InputDecoration(labelText: 'Username'),

),

TextField(

controller: passwordController,

obscureText: true,

decoration: InputDecoration(labelText: 'Password'),

),

SizedBox(height: 20),

ElevatedButton(

onPressed: login,

child: Text('Login'),

),

],

),

),

);

}

}

1. Develop a counter app with a "+" and "–" button to increase/decrease values.

import 'package:flutter/material.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: CounterScreen(),

);

}

}

class CounterScreen extends StatefulWidget {

@override

\_CounterScreenState createState() => \_CounterScreenState();

}

class \_CounterScreenState extends State<CounterScreen> {

int count = 0;

void increment() {

setState(() {

count++;

});

}

void decrement() {

setState(() {

count--;

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Counter App'),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

'Count:',

style: TextStyle(fontSize: 24),

),

Text(

'$count',

style: TextStyle(fontSize: 48, fontWeight: FontWeight.bold),

),

SizedBox(height: 20),

Row(

mainAxisAlignment: MainAxisAlignment.center,

children: [

ElevatedButton(

onPressed: decrement,

child: Text('-'),

),

SizedBox(width: 20),

ElevatedButton(

onPressed: increment,

child: Text('+'),

),

],

),

],

),

),

);

}

}

5) Build an app for counting the number of students entering a classroom using a floating action button to increase the count.

**Same**

**6)** Develop an app for adding and displaying a list of 5 books with title and author using text fields and list view

widgets.

import 'package:flutter/material.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: BookListScreen(),

);

}

}

class BookListScreen extends StatefulWidget {

@override

\_BookListScreenState createState() => \_BookListScreenState();

}

class \_BookListScreenState extends State<BookListScreen> {

List<Map<String, String>> books = [];

TextEditingController titleController = TextEditingController();

TextEditingController authorController = TextEditingController();

void addBook() {

if (titleController.text.isNotEmpty &&

authorController.text.isNotEmpty &&

books.length < 5) {

setState(() {

books.add({

'title': titleController.text,

'author': authorController.text,

});

titleController.clear();

authorController.clear();

});

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Book List App'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Column(

children: [

TextField(

controller: titleController,

decoration: InputDecoration(labelText: 'Book Title'),

),

TextField(

controller: authorController,

decoration: InputDecoration(labelText: 'Author'),

),

SizedBox(height: 20),

ElevatedButton(

onPressed: addBook,

child: Text('Add Book'),

),

SizedBox(height: 20),

Expanded(

child: ListView.builder(

itemCount: books.length,

itemBuilder: (context, index) {

return ListTile(

leading: Icon(Icons.book),

title: Text(books[index]['title']!),

subtitle: Text('by ${books[index]['author']}'),

);

},

),

),

],

),

),

);

}

}

7) Create an app for login authentication that shows a toast message for both successful and failed login attempts.

**Same**

**8)** Build an app for displaying an alert box to confirm user action before submitting a feedback form.

import 'package:flutter/material.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: FeedbackScreen(),

);

}

}

class FeedbackScreen extends StatefulWidget {

@override

\_FeedbackScreenState createState() => \_FeedbackScreenState();

}

class \_FeedbackScreenState extends State<FeedbackScreen> {

TextEditingController feedbackController = TextEditingController();

void submitFeedback() {

showDialog(

context: context,

builder: (context) => AlertDialog(

title: Text('Confirm Submission'),

content: Text('Are you sure you want to submit your feedback?'),

actions: [

TextButton(

onPressed: () {

Navigator.pop(context); // Close dialog

},

child: Text('Cancel'),

),

TextButton(

onPressed: () {

Navigator.pop(context); // Close dialog

ScaffoldMessenger.of(context).showSnackBar(

SnackBar(content: Text('Feedback Submitted!')),

);

feedbackController.clear(); // Clear text field

},

child: Text('Submit'),

),

],

),

);

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Feedback Form'),

),

body: Padding(

padding: EdgeInsets.all(16.0),

child: Column(

children: [

TextField(

controller: feedbackController,

decoration: InputDecoration(labelText: 'Enter your feedback'),

maxLines: 3,

),

SizedBox(height: 20),

ElevatedButton(

onPressed: submitFeedback,

child: Text('Submit'),

),

],

),

),

);

}

}

9) Develop an app for transferring money between two accounts using multiple screens and appropriate layouts

import 'package:flutter/material.dart';

void main() {

runApp(MaterialApp(

home: HomeScreen(),

debugShowCheckedModeBanner: false,

));

}

class HomeScreen extends StatelessWidget {

final String sender = 'Account 1';

final String receiver = 'Account 2';

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(title: Text('Bank App')),

body: Center(

child: ElevatedButton(

onPressed: () {

Navigator.push(

context,

MaterialPageRoute(

builder: (context) => TransferScreen(sender: sender, receiver: receiver),

),

);

},

child: Text('Transfer Money'),

),

),

);

}

}

class TransferScreen extends StatelessWidget {

final String sender;

final String receiver;

final TextEditingController amountController = TextEditingController();

TransferScreen({required this.sender, required this.receiver});

void showSuccessDialog(BuildContext context, String amount) {

showDialog(

context: context,

builder: (context) => AlertDialog(

title: Text('Success'),

content: Text('₹$amount transferred from $sender to $receiver!'),

actions: [

TextButton(

onPressed: () {

Navigator.pop(context); // Close dialog

Navigator.pop(context); // Go back to Home

},

child: Text('OK'),

),

],

),

);

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(title: Text('Transfer Money')),

body: Padding(

padding: EdgeInsets.all(20),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text('$sender ➡️ $receiver', style: TextStyle(fontSize: 20)),

SizedBox(height: 20),

TextField(

controller: amountController,

keyboardType: TextInputType.number,

decoration: InputDecoration(

labelText: 'Enter Amount',

border: OutlineInputBorder(),

),

),

SizedBox(height: 20),

ElevatedButton(

onPressed: () {

if (amountController.text.isNotEmpty) {

showSuccessDialog(context, amountController.text);

}

},

child: Text('Submit'),

),

],

),

),

);

}

}

10) Create an app for a student dashboard using custom widgets like AppBar with logo, Drawer menu, and BottomNavigationBar for navigation

import 'package:flutter/material.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: StudentDashboard(),

);

}

}

class StudentDashboard extends StatefulWidget {

@override

\_StudentDashboardState createState() => \_StudentDashboardState();

}

class \_StudentDashboardState extends State<StudentDashboard> {

int \_selectedIndex = 0;

// Simple pages for navigation

final List<Widget> \_pages = [

Center(child: Text('Home Page')),

Center(child: Text('Assignments Page')),

Center(child: Text('Profile Page')),

];

void \_onItemTapped(int index) {

setState(() {

\_selectedIndex = index; // Update selected page

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

// AppBar with title

appBar: AppBar(

title: Text('Student Dashboard'),

),

// Drawer Menu

drawer: Drawer(

child: ListView(

children: <Widget>[

DrawerHeader(

decoration: BoxDecoration(color: Colors.blue),

child: Text('Menu', style: TextStyle(color: Colors.white, fontSize: 24)),

),

ListTile(title: Text('Home')),

ListTile(title: Text('Assignments')),

ListTile(title: Text('Profile')),

],

),

),

// Display selected page

body: \_pages[\_selectedIndex],

// BottomNavigationBar for navigation

bottomNavigationBar: BottomNavigationBar(

items: const <BottomNavigationBarItem>[

BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),

BottomNavigationBarItem(icon: Icon(Icons.assignment), label: 'Assignments'),

BottomNavigationBarItem(icon: Icon(Icons.person), label: 'Profile'),

],

currentIndex: \_selectedIndex,

onTap: \_onItemTapped,

),

);

}

}