Module-5

Quiz

1. What is Flutter?
a) A programming language
b) A mobile operating system
c) A UI toolkit for building native applications
d) A cloud computing platform

Answer: c) A UI toolkit for building native applications

2.Which feature in Flutter allows developers to see the changes in the app's UI instantly?
a) Hot reload
b) Hot restart
c) Debug mode
d) Live preview

Answer: a) Hot reload

3. What does "Hot restart" do in Flutter?
a) Reboots the device/emulator
b) Reloads the entire app from scratch
c) Restarts the app without losing the app state
d) Closes the app and clears the cache

Answer: c) Restarts the app without losing the app state

4.What is the purpose of a package name in Flutter?
a) It identifies the version of the app
b) It determines the app's unique identifier
c) It specifies the location of the app's source code
d) It defines the app's layout and design

Answer: b) It determines the app's unique identifier

5.How can you create and run your first Flutter application?
a) Use the "flutter create" command in the terminal
b) Manually create the necessary files and folders
c) Copy an existing Flutter project and modify it
d) Install a Flutter template from the web

Answer: a) Use the "flutter create" command in the terminal

6.What is an AVD in Flutter development?
a) Audio Video Description
b) Android Virtual Device
c) App Version Descriptor
d) Automated Verification and Debugging

Answer: b) Android Virtual Device

7.In Flutter, what is the main source code flow?
a) main.dart -> runApp() -> build() -> Widget tree
b) runApp() -> main.dart -> build() -> Widget tree
c) build() -> runApp() -> main.dart -> Widget tree
d) Widget tree -> runApp() -> main.dart -> build()

Answer: a) main.dart -> runApp() -> build() -> Widget tree

8.Which class is used to define the root of a Flutter application?
a) Material App
b) Scaffold
c) Container
d) App

Answer: a) Material App

9.What does the FloatingActionButton widget represent in Flutter?
a) A button that appears at the bottom of the screen
b) A button that appears at the top of the screen
c) A button that appears in the centre of the screen
d) A button that floats above the other content

Answer: d) A button that floats above the other content

10.In Flutter, how can you add a leading icon/button to the app bar using Scaffold?
a) Use the "leading" property of the AppBar widget.
b) Add a IconButton as the first item in the "actions" property.
c) Use the "leadingIcon" property of the Scaffold widget.
d) You cannot add a leading icon/button to the app bar.

Answer: a) Use the "leading" property of the AppBar widget.

Live Test

Develop a Flutter app with a single screen that includes the following features:

AppBar: Customize the AppBar's background color, elevation, center-aligned title, and add at least two icons to the actions section.

Drawer: Include a banner with a custom message or image, and add two menu items using ListTile widgets.

Body: Customize the Scaffold's body with your preferred content, such as text, images, or any other widget.

Floating Action Button (FAB): Position the FAB in the bottom left corner, increase its size, and change its color to a custom choice.

Implement the above features to create a visually appealing and functional Flutter app with a single screen.

Solution:

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

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: MyScreen(),
    );
  }
}

class MyScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Colors.blue,
        elevation: 4.0,
        centerTitle: true,// Adjust the elevation
        title: const Text(
```

```
          'My App', // Center-aligned title
          style: TextStyle(
            fontSize: 20.0,
            fontWeight: FontWeight.bold,
          ),
        ),
        actions: [
          // Add at least two icons to the actions section
          IconButton(
            icon: const Icon(Icons.search),
            onPressed: () {
              // Perform search action
            },
          ),
          IconButton(
            icon: const Icon(Icons.settings),
            onPressed: () {
              // Open settings page
            },
          ),
        ],
      ),
      drawer: Drawer(
        child: Column(
          children: [
            // Include a banner with a custom message or image
            Container(
              height: 200,
              color: Colors.blue,
              child: Center(
                child: Text(
                  'My App Drawer',
                  style: TextStyle(
                    fontSize: 24.0,
                    color: Colors.white,
                  ),
                ),
              ),
            ),
            ListTile(
              title: Text('Menu Item 1'),
              onTap: () {
                // Handle menu item 1 tap
              },
            ),
            ListTile(
              title: Text('Menu Item 2'),
              onTap: () {
```

```
                // Handle menu item 2 tap
              },
            ),
          ],
        ),
      ),
      body: Center(
        child: Text(
          'Welcome to My App!', // Customize the body content
          style: TextStyle(fontSize: 24.0),
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          // Handle FAB tap
        },
        child: Icon(Icons.add),
        backgroundColor: Colors.red, // Change the FAB color
        elevation: 6.0, // Increase the size of the FAB
      ),
      floatingActionButtonLocation:
FloatingActionButtonLocation.startFloat,
    );
  }
}
```

Assignment

Create a Flutter app with three pages arranged in a tab view screen. Each page will have a different coloured floating action button and a centred text widget displaying the page number.

Requirements:

- Use MaterialApp and Scaffold widgets.
- Include an AppBar with a navigation drawer and a bottom navigation tab bar.
- The navigation drawer should allow users to navigate to each page.
- The bottom navigation tab bar should enable users to switch between the pages.
- Each page should have a unique coloured floating action button.
- Centre a text widget on each page displaying the page number.
- Ensure navigation from the drawer and tab bar leads to the correct pages.
- Use TabBarView and TabBar widgets for tabbed navigation.

Submission:

- Submit 3 screenshots of the app (3 screenshot of 3 pages and 1 screenshot for drawer)
- Submit github repo link of the complete assignment.

Solution:

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

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: MyTabbedScreen(),
    );
  }
}

class MyTabbedScreen extends StatefulWidget {
  @override
   _MyTabbedScreenState createState() => _MyTabbedScreenState();
}

class _MyTabbedScreenState extends State<MyTabbedScreen> with
SingleTickerProviderStateMixin {
  late TabController _tabController;

  @override
  void initState() {
    super.initState();
    _tabController = TabController(length: 3, vsync: this);
  }

  @override
  void dispose() {
    _tabController.dispose();
    super.dispose();
  }

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

```
      appBar: AppBar(
        title: const Text('Tabbed App'),
      ),
      drawer: Drawer(
        child: ListView(
          children: [
            ListTile(
              title:const  Text('Page 1'),
              onTap: () {
                _tabController.animateTo(0);
                Navigator.pop(context);
              },
            ),
            ListTile(
              title: const Text('Page 2'),
              onTap: () {
                _tabController.animateTo(1);
                Navigator.pop(context);
              },
            ),
            ListTile(
              title:const  Text('Page 3'),
              onTap: () {
                _tabController.animateTo(2);
                Navigator.pop(context);
              },
            ),
          ],
        ),
      ),
      body: TabBarView(
        controller: _tabController,
        children: const [
          Page(color: Colors.red, number: 1),
          Page(color: Colors.green, number: 2),
          Page(color: Colors.blue, number: 3),
        ],
      ),
      bottomNavigationBar: TabBar(
        controller: _tabController,
        tabs: const [
           Tab(icon: Icon(Icons.looks_one)),
           Tab(icon: Icon(Icons.looks_two)),
           Tab(icon: Icon(Icons.looks_3)),
        ],
      ),
    );
  }
```

```
}

class Page extends StatelessWidget {
  final Color color;
  final int number;

  const Page({required this.color, required this.number});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      floatingActionButton: FloatingActionButton(
        onPressed: () {},
        backgroundColor: color,
        child: Text(
          '$number',
          style: const TextStyle(
            fontSize: 18,
            fontWeight: FontWeight.bold,
          ),
        ),
      ),
      body: Center(
        child: Text(
          'Page $number',
          style: TextStyle(fontSize: 24),
        ),
      ),
    );
  }
}
```