DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

**LAB SESSION 7**

**Lab Session on Buttons and Images in Dart**

**Objective:**

The objective of this lab session is to familiarize the students with the implementation of buttons and images in Dart programming language. By the end of the session, participants should be able to understand the concept of buttons, different types of buttons available in Dart, and various methods to incorporate images into Dart applications.

**Introduction:**

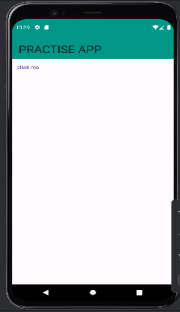
**Types of Buttons:**

User triggers the buttons to perform any action .buttons are basically graphical control elements that provide a user to trigger an event such as taking actions, making choices, searching things and many more. They can be placed anywhere in out UI like dialogues, forms, cards, tool bars etc.

**Text Button:** A basic button with text as its label. Text buttons are commonly used for simple actions. Previously it used to be a Flat button which is now deprecated.

TextButton(

onPressed: () {

// Add your action here 

},

child: Text('Click Me'),

)

**On Pressed:**

import 'package:flutter/material.dart';

void main() {

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState(); }

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

),

body:

TextButton(

child: Text("click me"),

onPressed: (){

print("button is pressed");

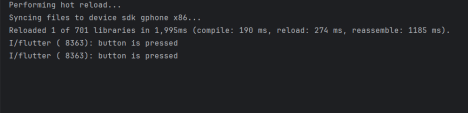
},

),

);

}

}



DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

**Elevated Button**: A button with elevation and a fill color when pressed. Elevated buttons are typically used for primary actions. Previously it used to be a Raised button which is now deprecated.

ElevatedButton(

onPressed: () {

// Add your action here

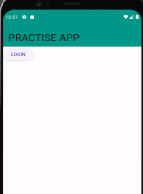
},

child: Text('Submit'),

) ElevatedButton(

import 'package:flutter/material.dart';

void main() {

runApp(const MyApp()); 

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState(); }

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

),

body:

ElevatedButton(

child: Text('LOGIN'),

onPressed: (){

print('elevated button is pressed');

},

)

);

}

}

**Outlined Button:** A button with an outline. Outlined buttons are useful for secondary actions or when less emphasis is required. Outlined button gives a ripple effect

OutlinedButton(

onPressed: () {

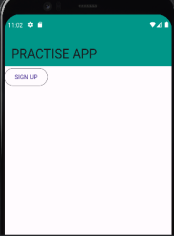
// Add your action here

},

child: Text('Cancel'),

)

import 'package:flutter/material.dart';

void main() { 

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState(); }

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

),

body:

OutlinedButton(

child: Text('SIGN UP'),

onPressed: (){

print("outlined button pressed");

},

)

);

}

}



**Icon Button:** A button that consists only of an icon. Icon buttons are suitable for actions that are represented by icons rather than text.

IconButton(

onPressed: () {

// Add your action here

},

icon: Icon(Icons.add),

)

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

**FloatingActionButton**: A circular button typically placed in a prominent position. Floating action buttons are used for primary or common actions in the application.

FloatingActionButton(

onPressed: () {

// Add your action here

},

child: Icon(Icons.add),

)

You can also accommodate tap or ink well widgets with buttons.

**Handling Button Presses:**

The on Pressed property of buttons in Dart specifies the action to be executed when the button is pressed. It typically takes a function as its value. However, there are alternative ways to handle button presses:

**On Long Pressed:**

In addition to handling regular button presses with the on Pressed property, Dart also provides the on Long Press property to handle long presses on buttons. Similar to on Pressed, on Long Press also takes a function as its value, but it triggers when the user holds the button for an extended period.

import 'package:flutter/material.dart';

void main() {

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState();

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

}

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

),

body:

TextButton(

child: Text("click me"),

onPressed: (){

print("button is pressed");

},

onLongPress: ()

{

print("button has been pressed for long"); },

),

);

}

}



**Images in Flutter:**

In Flutter, images can be incorporated into your application in various ways, including using assets, network images, and memory images. Here's an overview of each method:

Asset Images:

Asset images are static images that are bundled with your application. To use asset images, follow these steps:

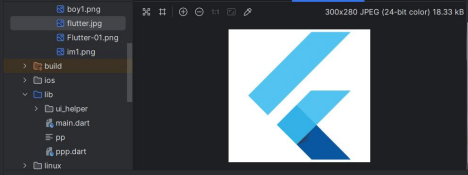
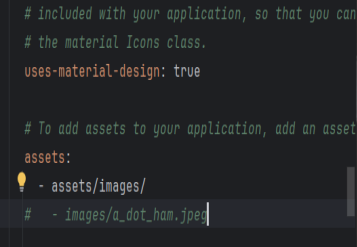
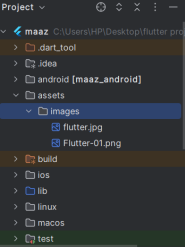
DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

∙ Place your image files in the assets directory of your Flutter project.

∙ Specify the asset in the pubspec.yaml file under the flutter section and run pub.get command. (process finished with exit code zero means pubget file has been updated without any error)

∙ Use the Image.asset() widget to display the asset image in your Flutter app.

∙ Run this command Image.asset('assets/images/image.png'),



import 'package:flutter/material.dart';

void main() {

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState(); }

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

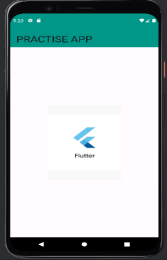
backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

),

body:

Center(child: Image.asset('assets/images/im1.png')) );

} 

}

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

import 'package:flutter/material.dart'; 

void main() {

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title:'flutter demo',

debugShowCheckedModeBanner: false,

theme:ThemeData(

primarySwatch:Colors.*lightBlue*,

),

home:const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key?key}):super(key:key);

@override

State<MyHomePage> createState() => \_MyHomePageState(); }

class \_MyHomePageState extends State<MyHomePage>

{

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

backgroundColor: Colors.*teal*,

title: Text('PRACTISE APP',style: TextStyle(fontSize: 30),),

),

body:

Container(

width:200,

height:200,

child: Image.asset('assets/images/im1.png')) );

}

}

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

**Network Images:**

Network images are dynamic images loaded from the internet. To use network images, provide the URL of the image to the Image.network() widget.Example:

Image.network('https://example.com/image.jpg'),

**Memory Images:**

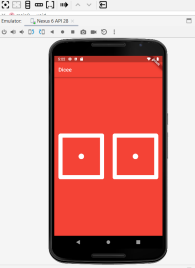
Memory images are images loaded directly from memory, such as byte arrays or Uint8Lists. This is useful for scenarios where you generate images dynamically.Example:

Image.memory(Uint8List.fromList(imageBytes)),

DEPARTMENT OF SOFTWARE ENGINEERING MOBILE APPLICATION DEVELOPMENT (SE-487)

**Exercise**

You're eager to play a game of Ludo, but alas, you don't have dice at hand. No worries, though! Let's leverage the power of Flutter to create a virtual solution. Imagine a Flutter application featuring two dice, each ready to roll at the tap of any button, generating random numbers just like real dice. Picture yourself and your friends huddled around your smartphone, taking turns rolling the dice virtually. How would you design and implement this Flutter application to simulate the excitement of rolling dice for your impromptu Ludo game session? Describe the key components and functionalities you would include in your Flutter application to replicate the thrill of rolling dice in the digital realm.



Code

//Musadiuqe Hussain SE-21031

//Muhammad Asim SE-21045

import 'package:flutter/material.dart';

import 'package:flutter/services.dart';

void main() {

  SystemChrome.setSystemUIOverlayStyle(SystemUiOverlayStyle(

    statusBarColor: Colors.red,

  ));

  runApp(MyApp());

}

class MyApp extends StatelessWidget {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      debugShowCheckedModeBanner: false,

      home: Scaffold(

        appBar: AppBar(

          backgroundColor: Colors.redAccent,

          title: Text('Dice'),

        ),

        body: Container(

          color: Colors.red,

          child: Center(

            child: Row(

              mainAxisAlignment: MainAxisAlignment.center,

              children: [

                Image.asset(

                  'assets/Images/dice1.png',

                  width: 160,

                  height: 160,

                ),

                SizedBox(width: 20),

                Image.asset(

                  'assets/Images/dice1.png',

                  width: 160,

                  height: 160,

                ),

              ],

            ),

          ),

        ),

      ),

    );

  }

}

Output

