LAB-8

OBJECTIVE: Introduction of Padding, Column with multiple row, card, BoxDecoration Class, SizedBox Class

Submitted by:

PATELWALA HARSH MITESH

ROLL NO: CE108 ID NO: 20CEUBS072

For the partial fulfilment of the award of degree

BACHELOR OF TECHNOLOGY

(COMPUTER ENGINEERING)



Department of Computer Engineering

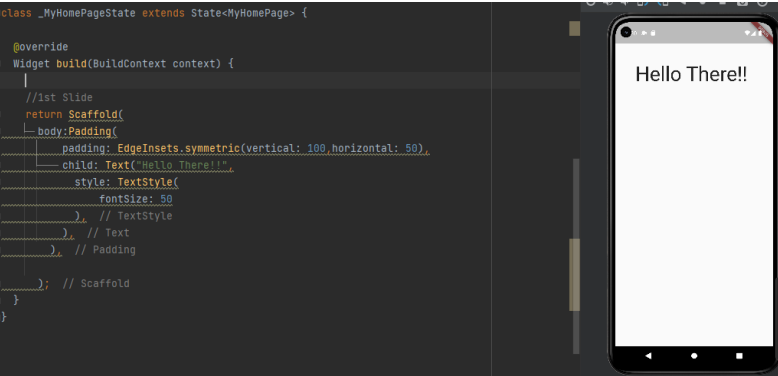
Faculty of Technology

# DHARMSINH DESAI UNIVERSITY

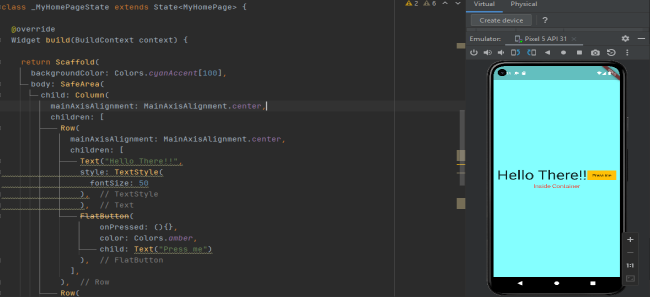
NAAC ACCREDUTED, ISO 9001:2008 CERTIFIED

NADIAD – 387001

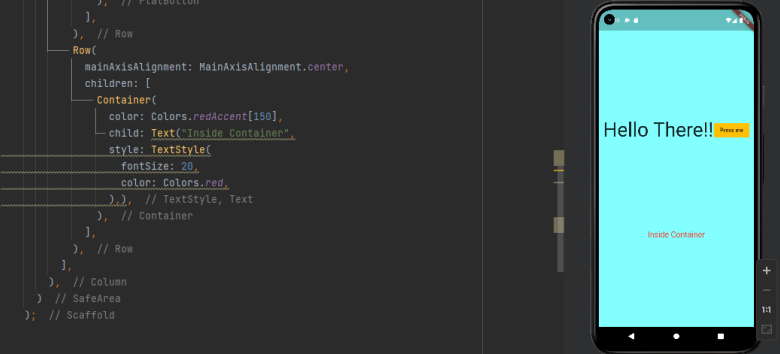
* **Padding:**
* When passing layout constraints to its child, padding shrinks the constraints by the given padding, causing the child to layout at a smaller size.
* Padding allows us to define padding but Container is the collection of small classes like Padding that makes the job easy.



* **Column with multiple row:**
* Column allows us to define multiple classes to be stored in a vertical way.
* Rows allows us to define multiple classes to be stored in a horizontal way.



* Rows and Column classes allow us to arrange the elements contained in them in different ways on two axis namely MainAxis and CrossAxis.



* **Card:**
* A Material Design card: a panel with slightly rounded corners and an elevation shadow. A card is a sheet of Material used to represent some related information, for example an album, a geographical location, a meal, contact details, etc.
* **BoxDecoration Class:**
* An immutable description of how to paint a box. The BoxDecoration class provides a variety of ways to draw a box. The box has a border, a body, and may cast a boxShadow. The shape of the box can be a circle or a rectangle. If it is a rectangle, then the borderRadius property controls the roundness of the corners.
* The body of the box is painted in layers. The bottom-most layer is the color, which fills the box. Above that is the gradient, which also fills the box. Finally there is the image, the precise alignment of which is controlled by the DecorationImage class.
* **SizedBox Class:**
* A box with a specified size. If either the width or height is null, this widget will try to size itself to match the child's size in that dimension. If the child's size depends on the size of its parent, the height and width must be provided.
* If given a child, this widget forces it to have a specific width and/or height. These values will be ignored if this widget's parent does not permit them. For example, this happens if the parent is the screen (forces the child to be the same size as the parent), or another SizedBox (forces its child to have a specific width and/or height). This can be remedied by wrapping the child SizedBox in a widget that does permit it to be any size up to the size of the parent, such as Center or Align.

import 'package:flutter/material.dart';

void main() {

  runApp(const MyApp());

}

class MyApp extends StatelessWidget {

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

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      home: const MyHomePage(title: 'Flutter Demo Home Page'),

    );

  }

}

class MyHomePage extends StatefulWidget {

  const MyHomePage({Key? key, required this.title}) : super(key: key);

  final String title;

  @override

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

}

class \_MyHomePageState extends State<MyHomePage> {

  @override

  Widget build(BuildContext context) {

    return Scaffold(

  );

// 2nd Slide

// return Scaffold(

//  backgroundColor: Colors.cyanAccent[100],

//  body: SafeArea(

//    child: Column(

//      mainAxisAlignment: MainAxisAlignment.spaceEvenly,

//      children: [

//        Row(

//          mainAxisAlignment: MainAxisAlignment.center,

//          children: [

//            Text("Hello There !",

//            style: TextStyle(

//            fontSize: 50

//            ),),

//            FlatButton(

//              onPressed: (){},

//              color: Colors.amber,

//              child: Text("Press me")

//            ),

//          ],

//        ),

//        Row(

//          mainAxisAlignment: MainAxisAlignment.center,

//          children: [

//            Container(

//              color: Colors.redAccent[150],

//              child: Text("Inside Container",

//              style: TextStyle(

//              fontSize: 20,

//              color: Colors.red,

//              ),),

//            ),

//          ],

//        ),

//      ],

//    ),)

// );

// 1st Slide

// return Scaffold(

//  body:Padding(

//    padding: EdgeInsets.symmetric(vertical: 100,horizontal: 50),

//    child: Text("Hello There !",

//    style: TextStyle(

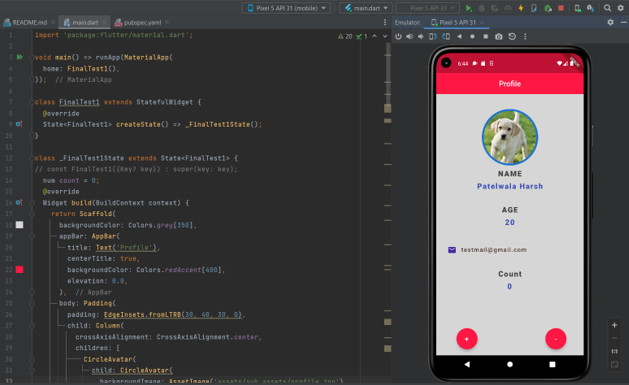
//    fontSize: 50

//    ),),

//  ),

// );

* **Profile Decoration:**

****

import 'package:flutter/material.dart';

void main() > runApp(MaterialApp(

  home: FinalTest1(),

));

class FinalTest1 extends StatefulWidget {

  @override

  State<FinalTest1> createState() > \_FinalTest1State();

}

class \_FinalTest1State extends State<FinalTest1> {

  const FinalTest1({Key? key}) : super(key: key);

  num count = 0;

  @override

Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: Colors.grey[350],

      appBar: AppBar(

      title: Text('Profile'),

      centerTitle: true,

      backgroundColor: Colors.redAccent[400],

      elevation: 0.0,

    ),

body: Padding(

      padding: EdgeInsets.fromLTRB(30, 40, 30, 0),

      child: Column(

        crossAxisAlignment: CrossAxisAlignment.center,

        children: [

          CircleAvatar(

            child: CircleAvatar(

              backgroundImage: AssetImage('assets/sub\_assets/dog4.jpg'),

              backgroundColor: Colors.redAccent[400],

              radius: 70,

            ),

            radius: 75,

          ),

          SizedBox(

            height: 10,

          ),

          Text(

            'NAME',

            style: TextStyle(

              fontWeight: FontWeight.w900,

              fontSize: 20,

              color: Colors.grey[800],

              letterSpacing: 2.0,

            ),

          ),

          SizedBox(height: 10),

          Text(

          'Patelwala Harsh',

            style: TextStyle(

              color: Colors.indigo[500],

              letterSpacing: 2.0,

              fontWeight: FontWeight.w900,

              fontSize: 20.0,

            ),

          ),

          SizedBox(height: 40),

          Text(

            'AGE',

            style: TextStyle(

              fontWeight: FontWeight.w900,

              fontSize: 20,

              color: Colors.grey[800],

              letterSpacing: 2.0,

            ),

          ),

SizedBox(height: 10),

          Text(

            '20',

            style: TextStyle(

              color: Colors.indigo[500],

              letterSpacing: 2.0,

              fontWeight: FontWeight.w900,

              fontSize: 20.0,

            ),

          ),

          SizedBox(height: 50),

          Row(

            children: [

              Icon(

                Icons.email\_rounded,

                color: Colors.deepPurple[800],

              ),

              SizedBox(width: 12.0),

              Text(

                'testmail@gmail.com',

                style: TextStyle(

                  color: Colors.brown[800],

                  fontWeight: FontWeight.w500,

                  fontSize: 16.0,

                  letterSpacing: 1.5,

                ),

              )

            ],

          ),

          SizedBox(height: 40,),

          Text(

            'Count',

            style: TextStyle(

              fontWeight: FontWeight.w900,

              fontSize: 20,

              color: Colors.grey[800],

              letterSpacing: 2.0,

            ),

          ),

SizedBox(height: 10),

          Text(

            "${count}",

            style: TextStyle(

              color: Colors.indigo[500],

              letterSpacing: 2.0,

              fontWeight: FontWeight.w900,

              fontSize: 20.0,

            ),

          ),

        ],

      ),

    ),

    floatingActionButton: Padding(

      padding: const EdgeInsets.fromLTRB(70, 0, 30, 0),

      child: Row(

        mainAxisAlignment: MainAxisAlignment.spaceBetween,

        children: [

          FloatingActionButton(

            onPressed: () {

              setState(() {

                count += 1;

                print(count);

              });

            },

            backgroundColor: Colors.redAccent[400],

            child: Text(

              "+",

              style: TextStyle(fontSize: 20),

            ),

          ),

          FloatingActionButton(

            onPressed: () {

              setState(() {

                count -= 1;

                print(count);

              });

            },

            backgroundColor: Colors.redAccent[400],

            child: Text(

            "-",

            style: TextStyle(fontSize: 20),

            ),

          ),

        ],

      ),

    ),

  );

}

}