Lab 8 : Tutorial 1

Link for assignment : https://github.com/Diya-Lad/SDP

Code test 1 :

Padding Widget:

- It has padding property which use EdgeInsets widget which we can use to give either same or separate padding in all 4 directions.
  Padding widget in flutter does exactly what its name says, it adds padding or empty space around a widget or a bunch of widgets.
- We can apply padding around any widget by placing it as the child of the Padding widget.
- The size of the child widget inside padding is constrained by how much space is remaining after adding empty space around.
- The Padding widget adds empty space around any widget by using the abstract EdgeInsetsGeometry class.

```
body: Padding(
  padding: EdgeInsets.all(50),
  child: Text('Hello World'),
),
```

code test 2: Row widget : for multiple components :

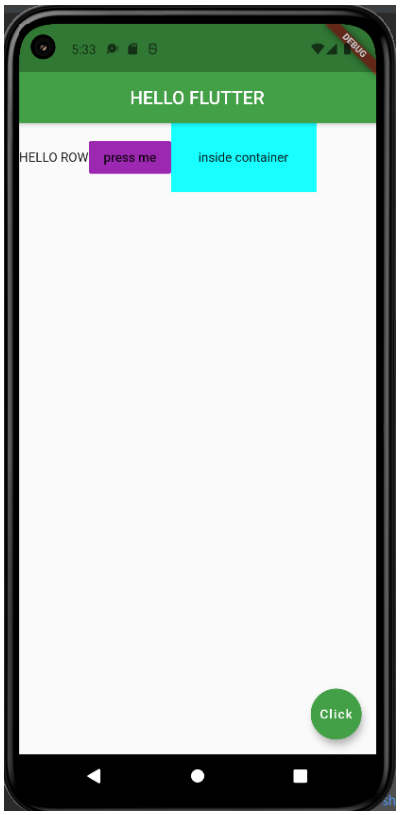
We have children property to the row widget where we can use set of widgets to place them in single row.

Here height of the row = maximum height (all widgets inside row)

- It creates a horizontal array of children.
- **Alignment Properties:**
  - We can align content as per our choice by using mainAxisAlignment and crossAxisAlignment.
  - Row's mainAxis is horizontal and cross Axis to Row's main Axis is vertical. We can align children horizontally using MainAxisAlignment and vertically using CrossAxisAlignment in that row.

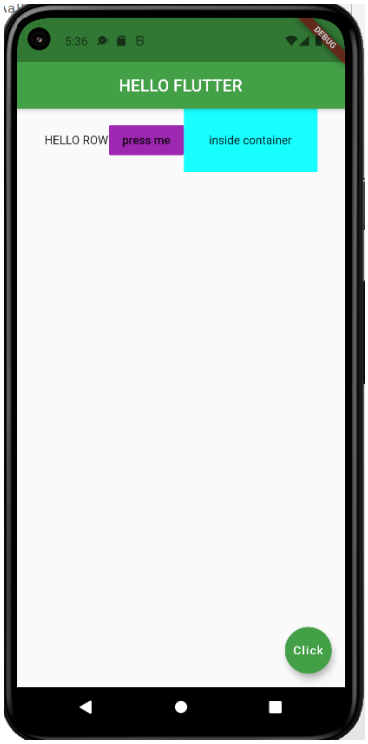
```
body: Row(
  children: [
    Text('HELLO ROW'),
    FlatButton(
      onPressed: () {},
      color: Colors.purple,
      child: Text('press me'),
    ),
    Container(
      color: Colors.cyanAccent,
      padding: EdgeInsets.all(30.0),
      child: Text('inside container'),
```

```
      ),
    ],
),
```



After adding these lines :

```
mainAxisAlignment: MainAxisAlignment.center,
crossAxisAlignment: CrossAxisAlignment.center,
```



Column widget:

Here Width of the column = maximum (Width of all components)

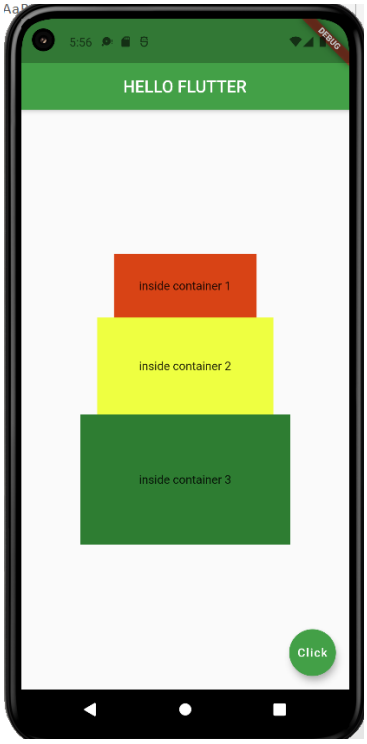
- It creates a vertical array of children.
- **Alignment Properties**:
  - In this also we have mainAxisAlignment and crossAxisAlignment. In column, children are aligned from top to bottom.
  - Main Axis is vertical and the Cross Axis is horizontal.
  - MainAxisAlignment aligns its children vertically and CrossAxisAlignment aligns horizontally in that Column.

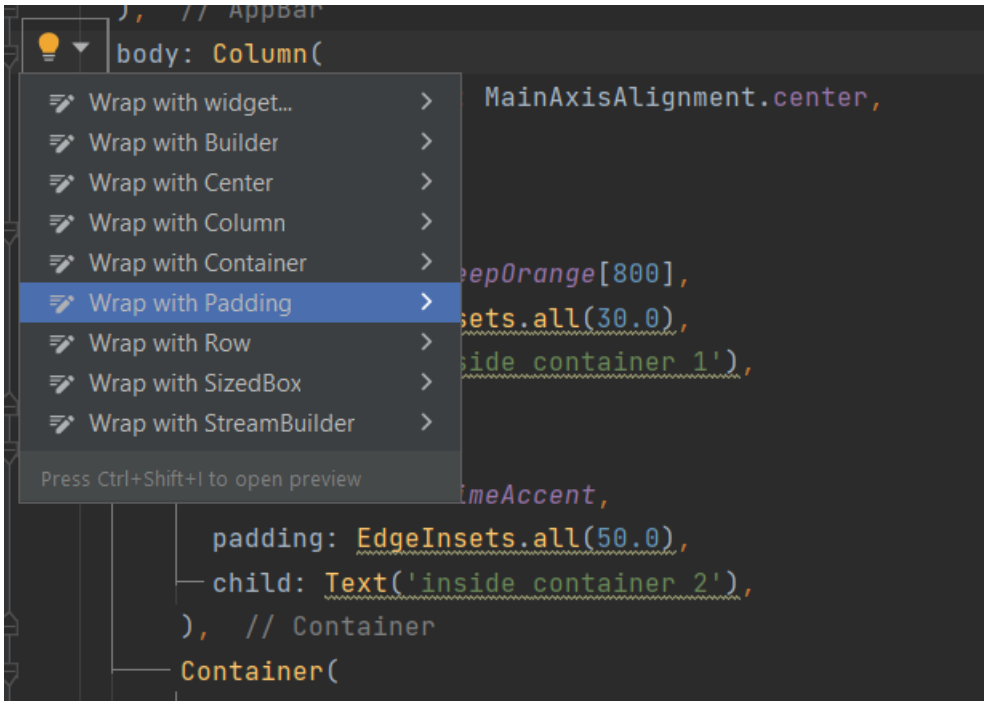
```
body: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    crossAxisAlignment: CrossAxisAlignment.center,
    children: [
      Container(
        color: Colors.deepOrange[800],
        padding: EdgeInsets.all(30.0),
        child: Text('inside container 1'),
      ),
      Container(
        color: Colors.limeAccent,
        padding: EdgeInsets.all(50.0),
        child: Text('inside container 2'),
      ),
      Container(
        color: Colors.green[800],
        padding: EdgeInsets.all(70.0),
```

```
        child: Text('inside container 3'),
      ),
  ],
),
```


```
body: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    // crossAxisAlignment: CrossAxisAlignment.center,
    children: [
      Row(),
      Container(
        color: Colors.deepOrange[800],
        padding: EdgeInsets.all(30.0),
        child: Text('inside container 1'),
      ),
      Container(
        color: Colors.limeAccent,
        padding: EdgeInsets.all(50.0),
        child: Text('inside container 2'),
      ),
      Container(
        color: Colors.green[800],
        padding: EdgeInsets.all(70.0),
        child: Text('inside container 3'),
      ),
  ],
),
```

Here we have added Row inside column whose width = screen size so this all container will be in middle because of mainAxisAlignment.

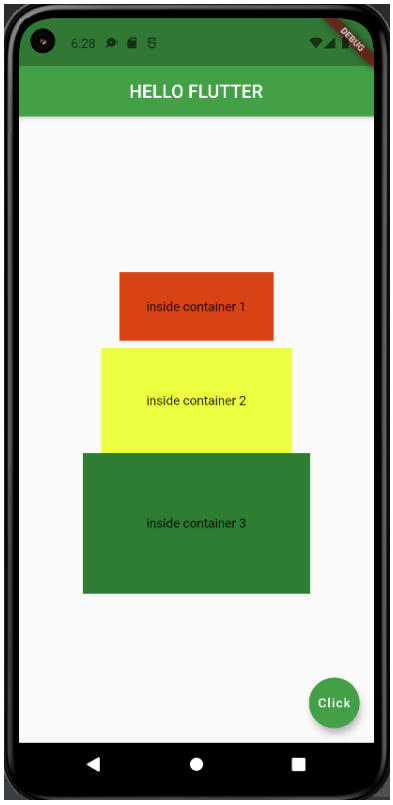




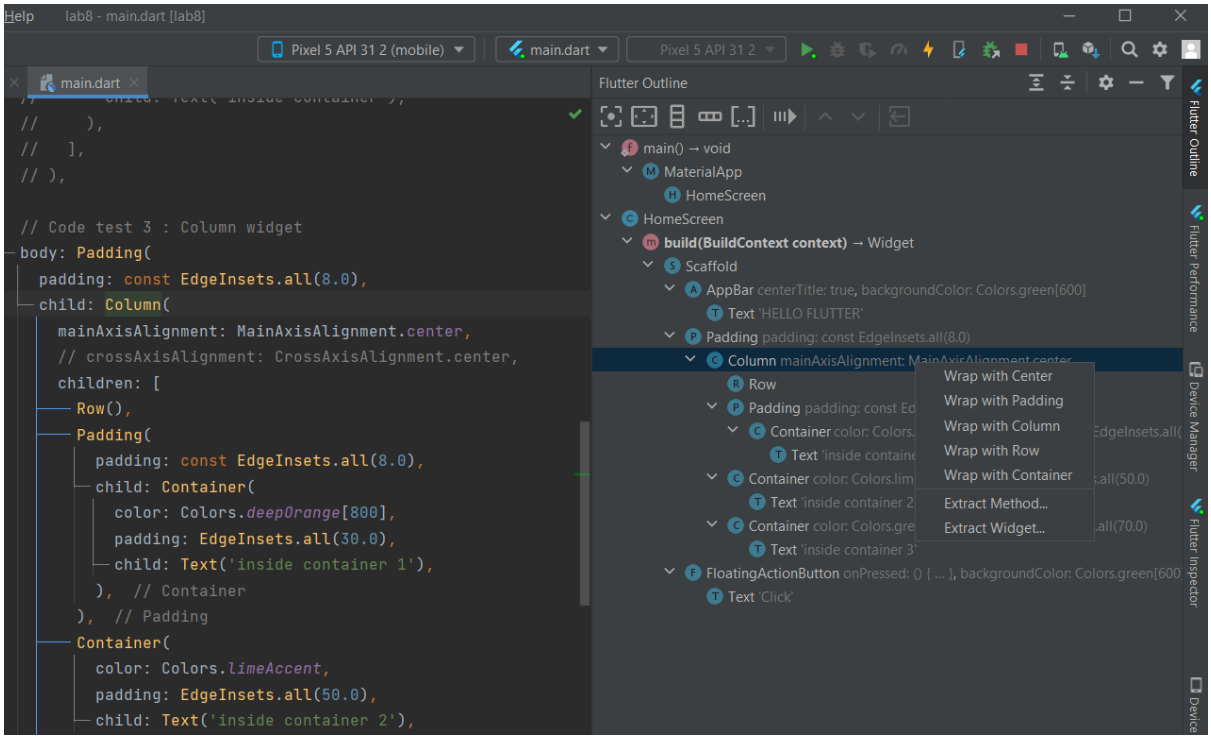
After clicking:

```
body: Padding(
  padding: const EdgeInsets.all(8.0),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Row(),
      Container(
        color: Colors.deepOrange[800],
        padding: EdgeInsets.all(30.0),
        child: Text('inside container 1'),
      ),  // Container
      Container(
        color: Colors.limeAccent,
        padding: EdgeInsets.all(50.0),
        child: Text('inside container 2'),
      ),  // Container
      Container(
```

```
body: Padding(
    padding: const EdgeInsets.all(8.0),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      // crossAxisAlignment: CrossAxisAlignment.center,
      children: [
        Row(),
        Padding(
          padding: const EdgeInsets.all(8.0),
          child: Container(
            color: Colors.deepOrange[800],
            padding: EdgeInsets.all(30.0),
            child: Text('inside container 1'),
          ),
        ),
        Container(
          color: Colors.limeAccent,
          padding: EdgeInsets.all(50.0),
          child: Text('inside container 2'),
        ),
        Container(
          color: Colors.green[800],
          padding: EdgeInsets.all(70.0),
          child: Text('inside container 3'),
        ),
    ],
),
  ),
```



We have another way of doing the same thing which we done above using flutter outline .



➢ Expanded and Flex widget:

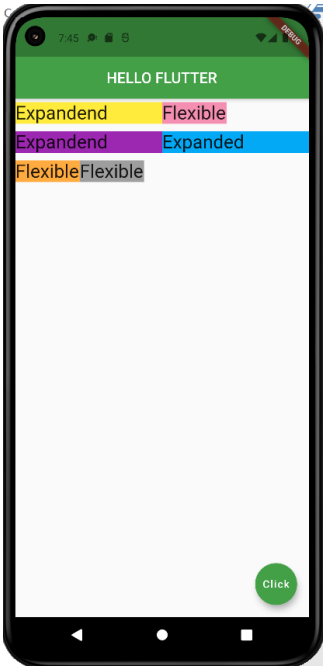
- Expanded widget : Creates a widget that expands a child of a Row, Column, or Flex so that the child fills the available space along the flex widget's main axis.

- Flex widget : Flexible is a built-in widget in flutter which controls how a child of base flex widgets that are Row, Column, and Flex will fill the space available to it.

Expanded widget will take fix place while flex will take place as per its size.

```
body: Padding(
  padding: const EdgeInsets.fromLTRB(0, 5, 0, 5),
  child: Container(
    child: Column(
      children: [
        Row(
          children: [
            Expanded(
              child: Container(
                color: Colors.yellow,
                child: Text(
                  'Expandend',
                  style: TextStyle(fontSize: 25),
                ),
              ),
            ),
            Flexible(child: Container(
              color: Colors.pink[200],
              child: Text(
                'Flexible',
                style: TextStyle(fontSize: 25),
              ),
            ),)
          ],
        ),
        SizedBox(
          height: 10.0,
        ),
        Row(
          children: [
            Expanded(
              child: Container(
                color: Colors.purple,
                child: Text(
                  'Expandend',
                  style: TextStyle(fontSize: 25),
                ),
```

```
              ),
            ),
            Expanded(
              child: Container(
                color: Colors.lightBlue,
                child: Text(
                  'Expanded',
                  style: TextStyle(fontSize: 25),
                ),
              ),)
          ],
        ),
        SizedBox(
          height: 10.0,
        ),
        Row(
          children: [
            Flexible(
              child: Container(
                color: Colors.orangeAccent,
                child: Text(
                  'Flexible',
                  style: TextStyle(fontSize: 25),
                ),
              ),
            ),
            Flexible(child: Container(
              color: Colors.grey,
              child: Text(
                'Flexible',
                style: TextStyle(fontSize: 25),
              ),
            ),)
          ],
        ),
      ],
    ),
  ),
),
```

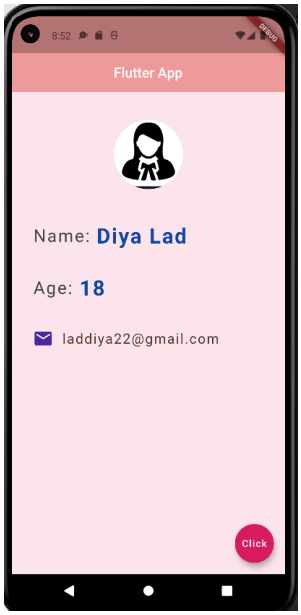


- Final custom design :

```
void main() {
  runApp(MaterialApp(
      home: Demo()
  ));
}


class Demo extends StatelessWidget {
// const FinalTest1({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.pink[50],
      appBar: AppBar(
        title: Text('Flutter App'),
        centerTitle: true,
        backgroundColor: Colors.red[200],
        elevation: 0.0,
      ),
      body: Padding(
        padding: EdgeInsets.fromLTRB(30, 40, 30, 0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Center(
              child:CircleAvatar(
                backgroundImage: AssetImage('asset/user.png'),
                radius: 50.0,
              ),
            ),
            SizedBox(height: 50,),
```

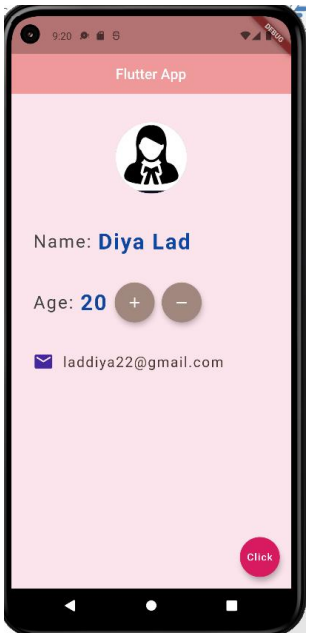
```
            Row(
              children: [
                Text(
                  'Name: ',
                  style: TextStyle(
                      color: Colors.grey[800],
                      letterSpacing: 2.0,
                      // fontWeight: FontWeight.bold,
                      fontSize: 25
                  ),
                ),
                Text(
                  'Diya Lad',
                  style: TextStyle(
                    color: Colors.blue[900],
                    letterSpacing: 2.0,
                    fontWeight: FontWeight.bold,
                    fontSize: 30.0,
                  ),
                ),
              ],
            ),
            SizedBox(height: 40),
            Row(
              children: [
                Text('Age: ',
                  style: TextStyle(
                    color: Colors.grey[800],
                    letterSpacing: 2.0,
                    fontSize: 25
                  ),
                ),
                Text('18',style: TextStyle(
                  color: Colors.blue[900],
                  letterSpacing: 2.0,
                  fontWeight: FontWeight.bold,
                  fontSize: 30.0,
                ),
                ),
              ],
            ),
            SizedBox(height: 40),
            Row(
              children: [
                Icon(
                  Icons.email_rounded,
                  color: Colors.deepPurple[800],
                  size: 30,
                ),
                SizedBox(width: 12.0),
                Text('laddiya22@gmail.com',
                  style: TextStyle(
                    color: Colors.brown[800],
                    fontSize: 20.0,
                    letterSpacing: 1.5,
                  ),
                )
              ],
            )
          ],
        ),
```

```
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {},
        child: Text('Click'),
        backgroundColor: Colors.pink[600],
      ),
    );
  }
}
```



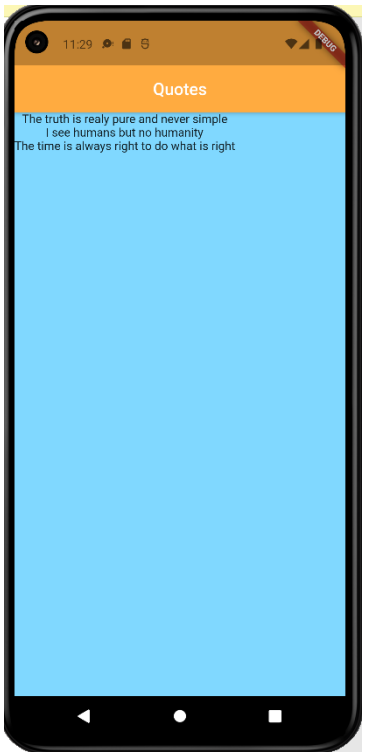

```
children: [
  Text('Age: ',
    style: TextStyle(
      color: Colors.grey[800],
      letterSpacing: 2.0,
      fontSize: 25
    ),
  ),
  Text(age.toString(),style: TextStyle(
    color: Colors.blue[900],
    letterSpacing: 2.0,
    fontWeight: FontWeight.bold,
    fontSize: 30.0,
  ),
  ),
  SizedBox(width: 10,),
  FloatingActionButton(onPressed: (){
    setState(() {
      age+=1;
    });
  },
    child: Icon(Icons.add),
    backgroundColor: Colors.brown[300],
  ),
  SizedBox(width: 10,),
  FloatingActionButton(
```

```
    onPressed: (){
      setState((){
        age -= 1;
      });
    },
    child: Icon(Icons.remove),
    backgroundColor: Colors.brown[300],
  ),

],
```




```
class EchoList extends StatefulWidget {
  const EchoList({Key? key}) : super(key: key);
  @override
  State<EchoList> createState() => _EchoListState();
}
class _EchoListState extends State<EchoList> {
  List<String> quotes = [
    'The truth is realy pure and never simple',
    'I see humans but no humanity',
    'The time is always right to do what is right'
  ];
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.lightBlueAccent[100],
      appBar: AppBar(
        title: Text('Quotes'),
        centerTitle: true,
        backgroundColor: Colors.orangeAccent,
      ),

      body: Column(
        children: quotes.map((quote) => Text(quote)).toList(),
      ),
    );
```

```
  }
}
```




```
List<Quote> quotes = [
  Quote(text: 'The truth is realy pure and never simple',author: 'xyz'),
  Quote(author: 'abc', text: 'I see humans but no humanity'),
  Quote(text: 'The time is always right to do what is right',author:
'pqr'),
];
Widget quoteTemplate(quote){
  return Card(
    margin: EdgeInsets.fromLTRB(20.0, 30.0, 40.0, 10.0),
    child: Padding(
      padding: const EdgeInsets.all(12.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.stretch,
        children: [
          Text(
            quote.text,
            style: TextStyle(
              fontSize: 20,
              color: Colors.deepPurple,
            ),
          ),
          SizedBox(height: 10),
          Text(
            quote.author,
            style: TextStyle(
              fontSize: 26,
              color: Colors.deepPurple,
            ),
          ),
```

```
        ],
      ),
    ),
  );
}
@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.lightBlueAccent[100],
    appBar: AppBar(
      title: Text('Quotes'),
      centerTitle: true,
      backgroundColor: Colors.orangeAccent,
    ),
    body: Column(
      children: quotes.map((quote) => quoteTemplate(quote)).toList(),
    ),
  );
}
```