

Dharmsinh Desai University, Nadiad
Faculty of Technology
Department of Computer Engineering



Lab-8
Smart Device Programming

Name: Vashishth Shaileshbhai Patel

Roll No: CE105

ID: 20CEUBG080

Batch: B2

Git Repo: https://github.com/vasu-1/SDP_Labs/tree/main/lab8

TUTORIAL – 1

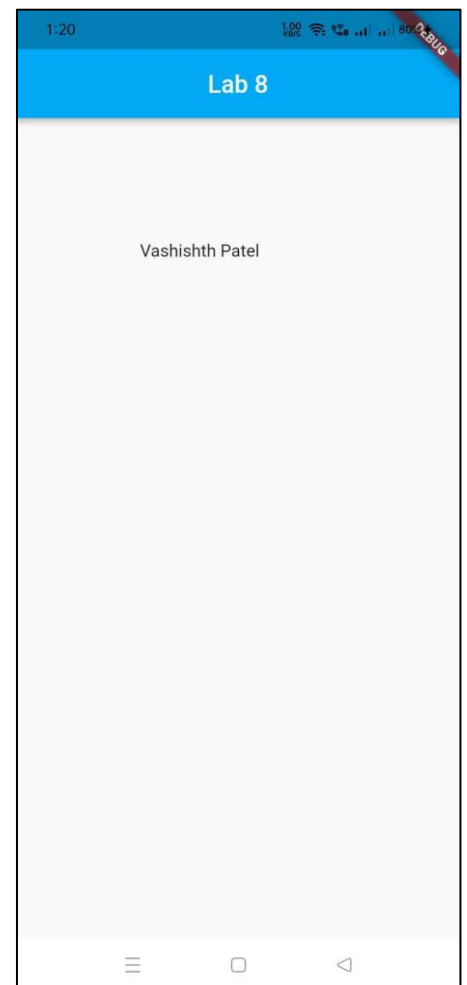
Padding Widget:

- 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.

Code:1

```
import "package:flutter/material.dart";
void main() {
  runApp(HomeScreen());
}
class HomeScreen extends StatelessWidget {
  const HomeScreen({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text("Lab 8"),
          centerTitle: true,
          backgroundColor: Colors.lightBlue,
        ),
        body: Container(
          child: Padding(
            padding: EdgeInsets.all(100),
            child: Text("Vashishth Patel"),
          ),
        ),
      ),
    );
  }
}
```



Code 2:

The Row Widget

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 is equal to 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.

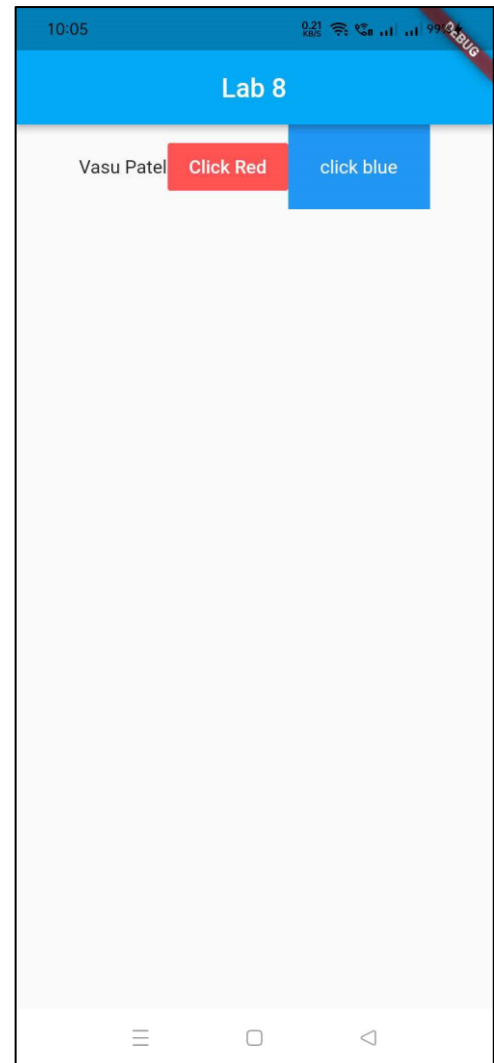
Row's mainAxis is horizontal and cross Axis to Row's main Axis is vertical. We can align children horizontally using MainAxisAlignment and vertically using CrossAxisAlignment in that row.

```
body: Row(  
  children: [  
    Text("Vasu Patel"),  
    FlatButton(  
      onPressed: () {},  
      color: Colors.redAccent,  
      child: Text("Click Red",  
        style: TextStyle(  
          color: Colors.white,  
        ),  
      ),  
    ),  
    Container(  
      color: Colors.blue,  
      padding: EdgeInsets.all(24),  
      child: Text("click blue",  
        style: TextStyle(  
          color: Colors.white,  
        ),  
      ),  
    ),  
  ],  
)
```



Code 3: Alignment of Row

```
body: Row(  
  mainAxisAlignment:  
    MainAxisAlignment.center,  
  crossAxisAlignment:  
    CrossAxisAlignment.center,  
  children: [
```



The Column Widget:

Here Width of the column is equal to maximum (Width of all components)

- It creates a vertical array of children.

- Alignment Properties:

- o In this also we have mainAxisAlignment and crossAxisAlignment. In column, children are aligned from top to bottom.

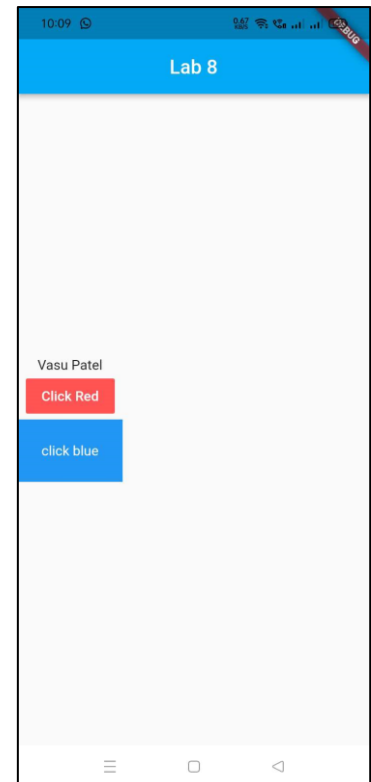
- o Main Axis is vertical and the Cross Axis is horizontal.

- o MainAxisAlignment aligns its children vertically and CrossAxisAlignment aligns horizontally in that Column

```

body: Column(
  mainAxisAlignment: MainAxisAlignment.center,
  crossAxisAlignment: CrossAxisAlignment.center,
  children: [
    Text("Vasu Patel"),
    FlatButton(
      onPressed: () {},
      color: Colors.redAccent,
      child: Text("Click Red",
        style: TextStyle(
          color: Colors.white,
        ),
      ),
    ),
    Container(
      color: Colors.blue,
      padding: EdgeInsets.all(24),
      child: Text("click blue",
        style: TextStyle(
          color: Colors.white,
        ),
      ),
    ),
  ],
),

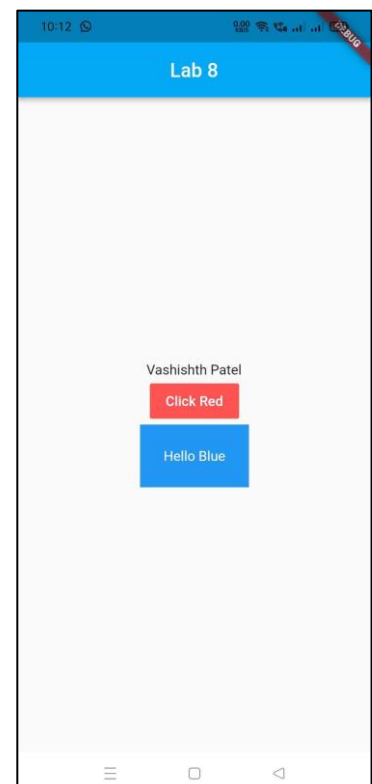
```



Fixing the alignment:

- Here, we added one extra child Row(), the height of row is zero here but its width is 100%. Hence, as the width of the column is equal to the width of its widest child, the width of the column will become 100% and will fix the alignment issue.

- body: Column(
 mainAxisAlignment: MainAxisAlignment.center,
 crossAxisAlignment:
 CrossAxisAlignment.center,
 children: [
 Row(),
 Text("Vashishth Patel"),
 FlatButton(
 onPressed: () {},
 color: Colors.redAccent,
 child: Text("Click Red",
 style: TextStyle(
 color: Colors.white,
),
),
),
],
),

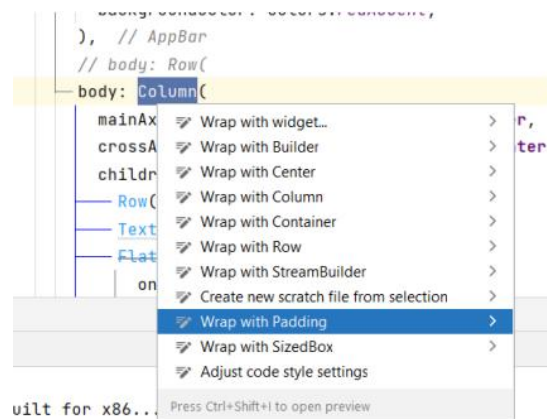


```

    ),
  ),
),
Container(
  color: Colors.blue,
  padding: EdgeInsets.all(24),
  child: Text("Hello Blue",
    style: TextStyle(
      color: Colors.white,
    ),
  ),
),
]
),

```

- Wrap an element instantly by right click > show context actions.

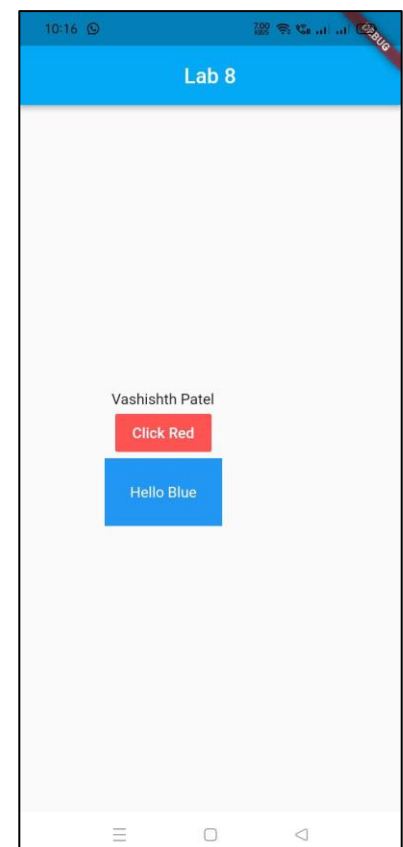


After clicking “Wrap with Padding”,

```

body: Padding(
  padding: const EdgeInsets.all(80.0),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    crossAxisAlignment: CrossAxisAlignment.center,
    children: [
      // Row(),
      Text("Vashishth Patel"),
      FlatButton(
        onPressed: () {},
        color: Colors.redAccent,
        child: Text("Click Red",
          style: TextStyle(
            color: Colors.white,
          ),
        ),
      ),
    ],
  ),
  Container(
    color: Colors.blue,
    padding: EdgeInsets.all(24),

```



```

        child: Text("Hello Blue",
          style: TextStyle(
            color: Colors.white,
          ),
        ),
      ],
    ),
  ),
),

```

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.

```

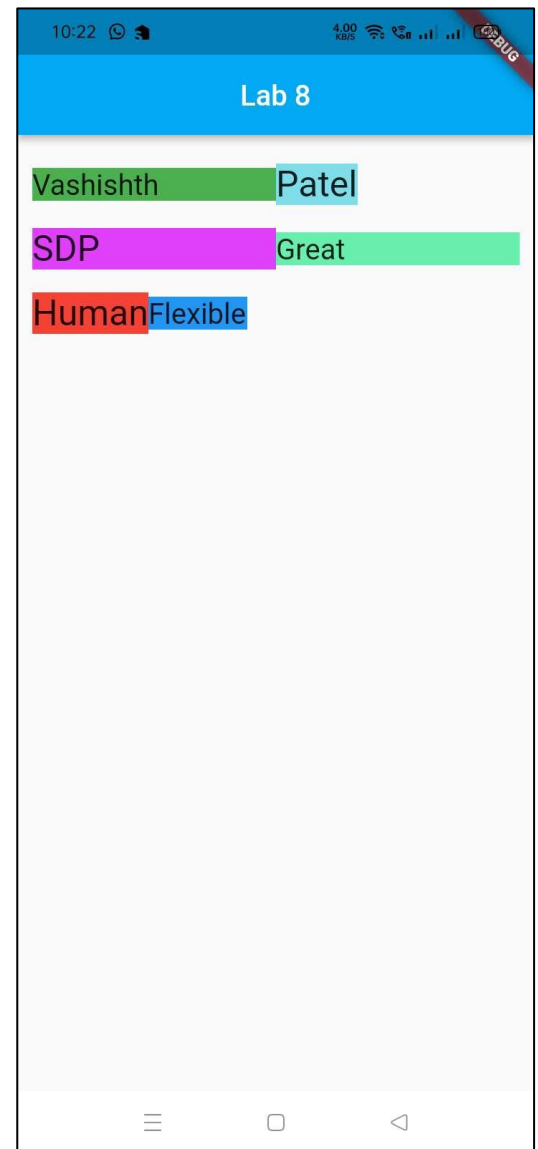
body: Padding(
  padding: const EdgeInsets.fromLTRB(10,20,10,20),
  child: Container(
    child: Column(
      children: [
        Row(
          children: [
            Expanded(
              child: Container(
                color: Colors.green,
                child: Text(
                  'Vashishth',
                  style: TextStyle(fontSize: 20),
                ),
              ),
            ),
          ],
        ),
        Flexible(child: Container(
          color: Colors.cyan[200],
          child: Text(
            'Patel',
            style: TextStyle(fontSize: 25),
          ),
        ),),
      ],
    ),
  ),
  SizedBox(
    height: 16.0,
  ),
),

```

```

Row(
  children: [
    Expanded(
      child: Container(
        color: Colors.purpleAccent,
        child: Text(
          'SDP',
          style: TextStyle(fontSize: 25),
        ),
      ),
    ),
    Expanded(
      child: Container(
        color: Colors.greenAccent,
        child: Text(
          'Great',
          style: TextStyle(fontSize: 20),
        ),
      ),
    ),
  ],
),
 SizedBox(
  height: 16.0,
),
Row(
  children: [
    Flexible(
      child: Container(
        color: Colors.red,
        child: Text(
          'Human',
          style: TextStyle(fontSize: 25),
        ),
      ),
    ),
    Flexible(
      child: Container(
        color: Colors.blue,
        child: Text(
          'Flexible',
          style: TextStyle(fontSize: 20),
        ),
      ),
    ),
  ],
),
),

```



TUTORIAL – 2

Stateful Widgets:

```
import "package:flutter/material.dart";
void main() {
  runApp(HomeScreen2());
}
class HomeScreen2 extends StatefulWidget {
  const HomeScreen2({Key? key}) : super(key: key);
  @override
  State<HomeScreen2> createState() => _HomeScreen2State();
}
class _HomeScreen2State extends State<HomeScreen2> {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text("Lab 8"),
          centerTitle: true,
          backgroundColor: Colors.blueAccent,
        ),
        body: ProfilePage(),
      ),
    );
  }
}
class ProfilePage extends StatefulWidget {
  const ProfilePage({Key? key}) : super(key: key);
  @override
  State<ProfilePage> createState() => _ProfilePageState();
}
class _ProfilePageState extends State<ProfilePage> {
  num age = 19;
  @override
  Widget build(BuildContext context) {
    return Container(
      padding: EdgeInsets.all(16.0),
      child: Column(
        // crossAxisAlignment: CrossAxisAlignment.center,
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Flexible(
                child: Image(
```

```

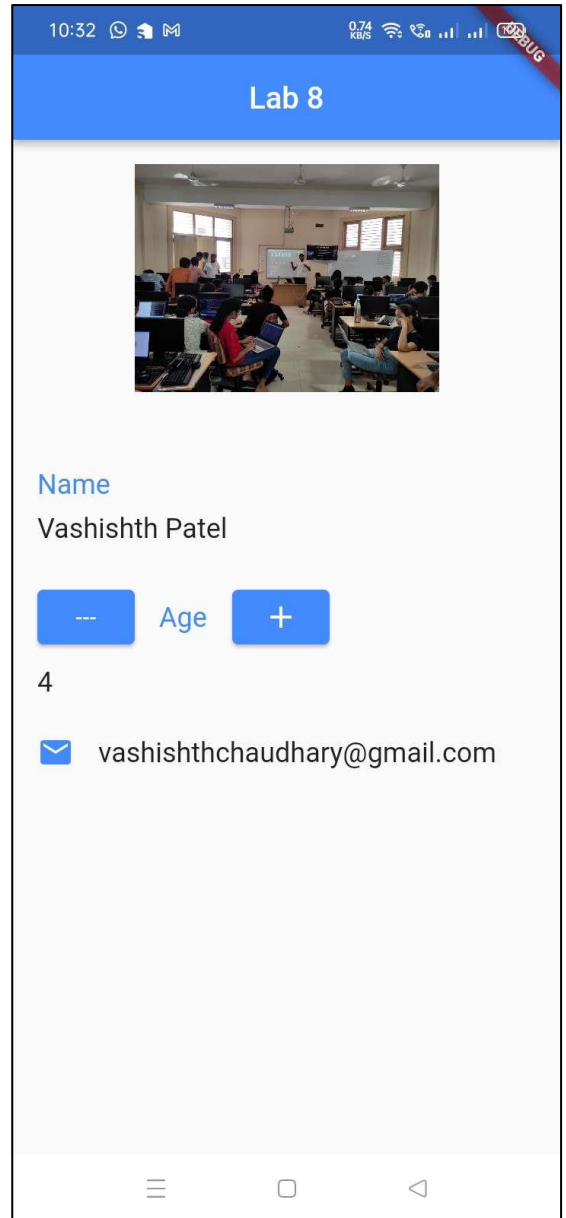
        image: AssetImage("images/snap.jpg"),
        width: 200,
      ),
    ),
  ],
),
PreferredSize(height: 50.0),
Container(
  child: Text(
    "Name",
    style: TextStyle(
      fontSize: 18,
      color: Colors.blueAccent,
    ),
  ),
),
PreferredSize(height: 8.0),
Container(
  child: Text(
    "Vashishth Patel",
    style: TextStyle(
      fontSize: 18.0,
    ),
  ),
),
PreferredSize(height: 24.0),
Row(
  children: [
    ElevatedButton(
      child: Text("----"),
      onPressed: () {
        setState(() {
          age -= 1;
        });
      },
      style: ElevatedButton.styleFrom(
        primary: Colors.blueAccent,
      ),
    ),
    Container(
      padding: EdgeInsets.fromLTRB(16, 0, 16, 0),
      child: Text(
        "Age",
        style: TextStyle(
          fontSize: 18.0,
          color: Colors.blueAccent,
        ),
      ),
    ),
    ElevatedButton(
      child: Icon(Icons.add),
      onPressed: () {
        setState(() {

```

```

        age += 1;
    });
},
style: ElevatedButton.styleFrom(
  primary: Colors.blueAccent,
),
)
],
),
SizedBox(height: 8.0),
Container(
  child: Text(
    age.toString(),
    style: TextStyle(
      fontSize: 18.0,
    ),
  ),
),
SizedBox(height: 24.0),
Row(
  children: [
    Icon(
      Icons.mail,
      color: Colors.blueAccent,
    ),
    Padding(
      padding: EdgeInsets.fromLTRB(16.0, 0, 0, 0),
      child: Text(
        "vashishthchaudhary@gmail.com",
        style: TextStyle(
          fontSize: 18.0,
        ),
      ),
    ],
  ),
),
),
);
}
}

```



Here, when we click on the (+) button, it calls the `setState()` method and hence the counter gets increased by 1. As this is a stateful widget, its state gets changed and so is the change in the value reflected on the screen.

Here, when we click on the (-) button, it calls the `setState()` method and hence the counter gets increased by 1. As this is a stateful widget, its state gets changed and so is the change in the value reflected on the screen.

Quotes Application:

Main.dart

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

void main() => runApp(MaterialApp(
  home: EchoList(),
));

class EchoList extends StatefulWidget {
  // const EchoList({Key? key}) : super(key: key);

  @override
  State<EchoList> createState() => _EchoListState();
}

class _EchoListState extends State<EchoList> {
  List<Quote> quotes = [
    Quote(text: 'The truth is really pure', author: 'Vasu'),
    Quote(author: 'Vashishth', text: 'I love to see people'),
    Quote(author: 'Vashishth', text: 'I love to show people'),
    Quote(author: 'Vashishth', text: 'I love to play with people'),
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blue[200],
      appBar: AppBar(
        title: Text('Quotes'),
        centerTitle: true,
        backgroundColor: Colors.blueAccent[300],
      ),
      body: Column(
        // mainAxisAlignment: MainAxisAlignment.center,
        // crossAxisAlignment: CrossAxisAlignment.center,
        // children: quotes.map((quote) => quoteTemplate(quote)).toList(),
        children: quotes.map((quote) => QuoteCard(
          quote: quote,
          delete: () {
            setState(() {
              quotes.remove(quote);
            });
          },
        )).toList(),
      ),
    );
  }
}
```

quote.dart

```
class Quote {
  String text;
  String author;
  Quote({this.text='', this.author=''});
}
```

quote_card.dart

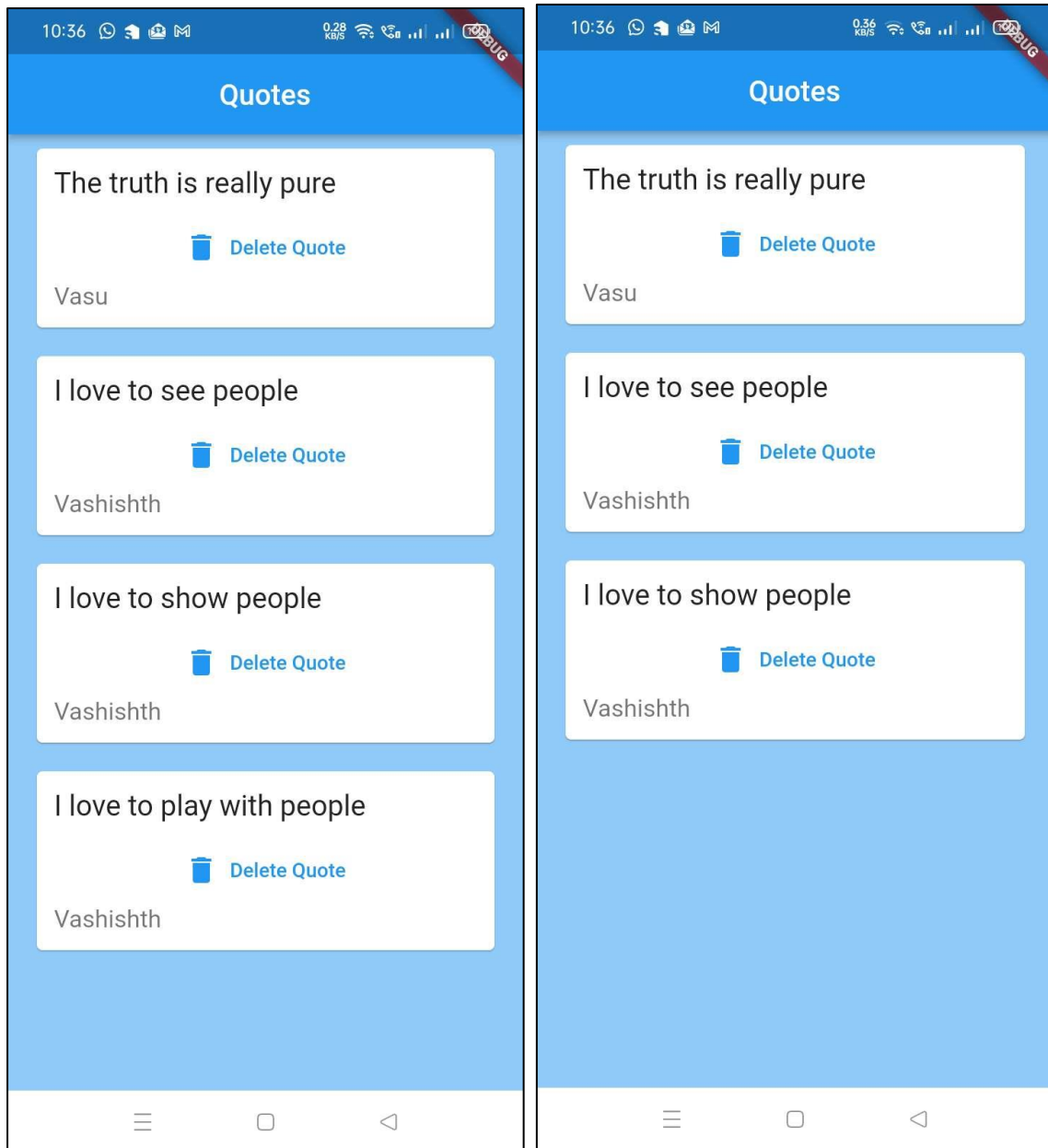
```
import 'package:flutter/material.dart';
import 'quote.dart';

class QuoteCard extends StatelessWidget {
  // const QuoteCard({
  //   Key? key,
  // }) : super(key: key);
  final Quote quote;
  final VoidCallback delete;

  QuoteCard({required this.quote, required this.delete});

  @override
  Widget build(BuildContext context) {
    return Card(
      margin: const EdgeInsets.fromLTRB(20.0, 10.0, 20.0, 10.0),
      child: Padding(
        padding: const EdgeInsets.all(12.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: [
            Text(
              quote.text,
              style: const TextStyle(
                fontSize: 20,
                color: Colors.black87,
              ),
            ),
            SizedBox(height: 10),
            TextButton.icon(
              /*
               our data doesn't exist inside this stateless widget class..
               data are in 'main.dart' file. ..
              */
              onPressed: delete,
              icon: Icon(Icons.delete),
              label: Text('Delete Quote'),
            ),
            Text(
              quote.author,
              style: TextStyle(
                fontSize: 17,
                color: Colors.black54,
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

```
} ;  
}  
}
```



THANK YOU