

# TASK 3

## Abdul Moin Qureshi

### CU-1977-2021

---

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Widget Demo',
      home: Scaffold(
        appBar: AppBar(title: Text('Widget Demo')),
        body: DemoPage(),
      ),
    );
  }
}

class DemoPage extends StatefulWidget {
  @override
```

```
_DemoPageState createState() => _DemoPageState();  
}
```

```
class _DemoPageState extends State<DemoPage> {
```

```
  bool _isVisible = true;
```

```
  double _opacityValue = 1.0;
```

```
  void _toggleVisibility() {
```

```
    setState(() {
```

```
      _isVisible = !_isVisible;
```

```
    });
```

```
  }
```

```
  void _updateOpacity(double value) {
```

```
    setState(() {
```

```
      _opacityValue = value;
```

```
    });
```

```
  }
```

```
@override
```

```
Widget build(BuildContext context) {
```

```
  return SingleChildScrollView(
```

```
    padding: EdgeInsets.all(16.0),
```

```
    child: Column(
```

```
      crossAxisAlignment: CrossAxisAlignment.start,
```

```
      children: <Widget>[
```

```

// Section for Positioned Widget
Padding(
  padding: const EdgeInsets.only(bottom: 16.0),
  child: Text(
    'Positioned Widget',
    style: Theme.of(context).textTheme.headline6,
  ),
),
Container(
  color: Colors.grey[200],
  height: 200,
  child: Stack(
    children: <Widget>[
      Positioned(
        left: 20,
        top: 40,
        child: Container(
          color: Colors.blue,
          width: 100,
          height: 50,
          child: Center(child: Text('Positioned', style: TextStyle(color: Colors.white))),
        ),
      ),
    ],
  ),
),

```

```
Divider(),
```

```
// Section for Aligned Widget
```

```
Padding(
```

```
padding: const EdgeInsets.only(bottom: 16.0, top: 16.0),
```

```
child: Text(
```

```
  'Aligned Widget',
```

```
  style: Theme.of(context).textTheme.headline6,
```

```
),
```

```
),
```

```
Container(
```

```
color: Colors.grey[200],
```

```
height: 100,
```

```
child: Align(
```

```
alignment: Alignment.bottomRight,
```

```
child: Container(
```

```
color: Colors.green,
```

```
width: 100,
```

```
height: 50,
```

```
child: Center(child: Text('Aligned', style: TextStyle(color: Colors.white))),
```

```
),
```

```
),
```

```
),
```

```
Divider(),
```

```
// Section for Spacer Widget
```

```

Padding(
  padding: const EdgeInsets.only(bottom: 16.0, top: 16.0),
  child: Text(
    'Spacer Widget',
    style: Theme.of(context).textTheme.headline6,
  ),
),
Row(
  children: <Widget>[
    Container(color: Colors.red, width: 50, height: 50, child: Center(child: Text('Box 1', style:
TextStyle(color: Colors.white)))),
    Spacer(),
    Container(color: Colors.orange, width: 50, height: 50, child: Center(child: Text('Box 2',
style: TextStyle(color: Colors.white)))),
  ],
),
Divider(),

```

// Section for Visibility Widget

```

Padding(
  padding: const EdgeInsets.only(bottom: 16.0, top: 16.0),
  child: Text(
    'Visibility Widget',
    style: Theme.of(context).textTheme.headline6,
  ),
),
Visibility(

```

```
visible: _isVisible,
child: Container(
  color: Colors.purple,
  width: 100,
  height: 50,
  child: Center(child: Text('Visible Widget', style: TextStyle(color: Colors.white))),
),
),
SizedBox(height: 8),
ElevatedButton(
  onPressed: _toggleVisibility,
  child: Text(_isVisible ? 'Hide Widget' : 'Show Widget'),
),
Divider(),
```

```
// Section for Opacity Widget
```

```
Padding(
  padding: const EdgeInsets.only(bottom: 16.0, top: 16.0),
  child: Text(
    'Opacity Widget',
    style: Theme.of(context).textTheme.headline6,
  ),
),
Opacity(
  opacity: _opacityValue,
  child: Container(
```

```
        color: Colors.blueAccent,
        width: 100,
        height: 50,
        child: Center(child: Text('Opacity Widget', style: TextStyle(color: Colors.white))),
      ),
    ),
    SizedBox(height: 8),
    Text('Opacity: ${_opacityValue.toStringAsFixed(1)}'),
    Slider(
      min: 0.0,
      max: 1.0,
      value: _opacityValue,
      onChanged: _updateOpacity,
    ),
  ],
),
);
}
}
```

---