Code 4.1: Container widget without child

void main() {

runApp(

const HelloApp()

);

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: Container(

color: Colors.red,

),

)

);

}

}

Code 4.2: Container widget with a text widget child

void main() {

runApp(

const HelloApp()

);

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: Container(

color: Colors.red,

child: const Text("Hello User"),

),

)

);

}

}

Code 4.3: SafeArea example

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: SafeArea(

child: Container(

color: Colors.red,

child: const Text("Hello User")

),

),

)

);

}

Code 4.3: Container widget with margin and padding

void main() {  
 runApp(  
 const HelloApp()  
 );  
}  
  
class HelloApp extends StatelessWidget {  
 const HelloApp({super.key});  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home: Scaffold(  
 backgroundColor: Colors.green,  
 body: Container(  
 color: Colors.red,  
 margin: const EdgeInsets.symmetric(vertical: 40, horizontal: 20),  
 padding: const EdgeInsets.all(40),  
 child: const Text("Hello User")  
 ),  
 )  
 );  
 }  
}

Code 4.3: Center widget with Container child

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: Center(

child: Container(

color: Colors.red,

child: const Text("Hello User"),

),

),

),

);

}

Code 4.3: Center widget with Text child

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: Container(

color: Colors.red,

child: const Center(

child: Text("Hello User"),

),

),

),

);

}

Code 4.4: align widget

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: Center(

child: Container(

color: Colors.red,

width: 120,

height: 120,

child: const Align(

alignment: Alignment.bottomLeft,

child: Text("Hello User"),

),

),

),

),

);

}

Code 4.5:

const Align(

alignment: Alignment(1,1),

child: Text("Hello User"),

),

Code 4.6:

@override

Widget build(BuildContext context) {

return const MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: SafeArea(

child: Padding(

padding: EdgeInsets.all(20.0),

child: Text("Hello User"),

),

),

),

);

}

Code 4.10: Column Widget

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.green,

body: SafeArea(

child: Column(

verticalDirection: VerticalDirection.up,

mainAxisAlignment: MainAxisAlignment.spaceAround,

crossAxisAlignment: CrossAxisAlignment.end,

children: [

Container(

color: Colors.orange,

height: 100,

width: 100

),

const SizedBox(

height: 20.0,

),

Container(

color: Colors.blue,

height: 100,

width: 100

),

const SizedBox(

height: 20.0,

),

Expanded(

child: Container(

color: Colors.pink,

height: 100,

width: 100

),

),

Container(

width: double.infinity,

)

],

)),

),

);

}

Code 4.11: Solution for the assignment

import 'package:flutter/material.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.white,

body: SafeArea(

child: Row(

crossAxisAlignment: CrossAxisAlignment.end,

children: [

Container(

color: Colors.orange,

height: double.infinity,

width: 100

),

Expanded(

child: Column(

mainAxisAlignment: MainAxisAlignment.spaceEvenly,

children: [

Container(

color: Colors.blue,

height: 100,

width: 100

),

Container(

color: Colors.green,

height: 100,

width: 100

),

],

),

),

Container(

color: Colors.pink,

height: 100,

width: 100

),

],

),

),

),

);

}

}

Code 4.12: Stack widget example

import 'package:flutter/material.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

home: Scaffold(

backgroundColor: Colors.white,

body: SafeArea(

child: Center(

child: Stack(

alignment: AlignmentDirectional.bottomCenter,

children: [

Container(color: Colors.orange, height: 140, width: 140),

Positioned(

top: 50,

left: 50,

child: Container(

width: 100,

height: 100,

color: Colors.blue,

),

),

Container(color: Colors.pink, height: 60, width: 80),

],

),

),

),

),

);

}

}

Code 4.13: ListView example

SafeArea(

child: Center(

child: ListView(

padding: const EdgeInsets.all(8),

scrollDirection: Axis.horizontal,

children: <Widget>[

Container(

width: 230,

color: Colors.teal[600],

child: const Center(child: Text('Item 1')),

),

Container(

width: 230,

color: Colors.yellow[200],

child: const Center(child: Text('Item 2')),

),

Container(

width: 230,

color: Colors.blue[400],

child: const Center(child: Text('Item 3')),

),

],

)

),

),

Code 4.14: ListView.separated

ListView.separated(

padding: const EdgeInsets.all(8),

itemCount: 100, // The total number of items in the list

itemBuilder: (BuildContext context, int index) {

// itemBuilder is a callback function that builds each item in the list

// based on its index

// Return a widget for each item in the list

return Container(

height: 50,

color: Colors.teal[100],

child: Center(child: Text('Entry $index')),

);

},

separatorBuilder: (BuildContext context, int index) => const Divider(),

)

Code 4.15: slivers example

CustomScrollView(

slivers: [

const SliverAppBar(

title: Text('Sliver Example'),

expandedHeight: 200,

backgroundColor: Colors.teal,

),

SliverList(

delegate: SliverChildBuilderDelegate(

(BuildContext context, int index) {

return Container(

height: 50,

color: Colors.teal[100],

child: Padding(

padding: const EdgeInsets.all(8.0),

child: Text('Entry $index'),

),

);

},

childCount: 20,

),

),

],

),

Code 4.16: GridView example

GridView.count(

crossAxisCount: 2, // Number of columns in the grid

padding: const EdgeInsets.all(8),

crossAxisSpacing: 10,

mainAxisSpacing: 10,

children: List.generate(

10, // Total number of items in the grid

(index) {

return Container(

color: Colors.blueGrey, // Example widget, replace with your own

child: Center(

child: Text('Item $index'),

),

);

},

),

)

Code 4.17: image asset widget

Image.asset('assets/images/logo.png')

Code 4.18: full image path

assets:

- images/mastercard-logo.png

Code 4.19: images path

assets:

- images/

Code 4.20: network image widget

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

Code 4.21: stateless widget

import 'package:flutter/material.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatelessWidget {

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

@override

Widget build(BuildContext context) {

void submitButtonPressed() {

// Add button click logic here

print('Submit button pressed!');

}

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: const Text('Hello User'),

),

body: Padding(

padding: const EdgeInsets.all(16.0),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

const Text(

'Welcome User',

style: TextStyle(fontSize: 24),

),

const SizedBox(height: 16),

const TextField(

decoration: InputDecoration(

labelText: 'Enter your name',

),

),

const SizedBox(height: 16),

ElevatedButton(

onPressed: submitButtonPressed,

child: const Text('Submit'),

),

],

),

),

),

);

}

}

Code 4.22: Stateful widget

import 'package:flutter/material.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatefulWidget {

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

@override

State<HelloApp> createState() => \_HelloAppState();

}

class \_HelloAppState extends State<HelloApp> {

String userName = 'User';

final TextEditingController \_textEditingController = TextEditingController();

@override

void dispose() {

\_textEditingController.dispose();

super.dispose();

}

void submitButtonPressed() {

setState(() {

userName = \_textEditingController.text;

});

}

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: const Text('Hello User'),

),

body: Padding(

padding: const EdgeInsets.all(16.0),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

Text(

'Welcome $userName',

style: const TextStyle(fontSize: 24),

),

const SizedBox(height: 16),

TextField(

controller: \_textEditingController,

decoration: const InputDecoration(

labelText: 'Enter your name',

),

),

const SizedBox(height: 16),

ElevatedButton(

onPressed: submitButtonPressed,

child: const Text('Submit'),

),

],

),

),

),

);

}

}

Code 4.23: Material widget

import 'package:flutter/material.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: const Text('Material App'),

),

body: Center(

child: ElevatedButton(

child: const Text('Button'),

onPressed: () {

// Button press logic

},

),

),

),

);

}

}

Code 4.24: Cupertino widget

import 'package:flutter/cupertino.dart';

void main() {

runApp(const HelloApp());

}

class HelloApp extends StatelessWidget {

const HelloApp({super.key});

@override

Widget build(BuildContext context) {

return CupertinoApp(

home: CupertinoPageScaffold(

navigationBar: const CupertinoNavigationBar(

middle: Text('Cupertino App'),

),

child: Center(

child: CupertinoButton(

child: const Text('Button'),

onPressed: () {

// Button press logic

},

),

),

),

);

}

}

Code 4.25: Divider widget

void main() {

runApp(const MyWidget());

}

class MyWidget extends StatelessWidget {

const MyWidget({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Directionality(

textDirection: TextDirection.ltr, // Set the desired text direction

child: Scaffold(

appBar: AppBar(

title: const Text('My App'),

),

body: const Column(

children: [

Text('Section 1'),

Divider(),

Text('Section 2'),

],

),

),

),

);

}

}

Code 4.25: Slider widget

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: const Text('Slider Example'),

),

body: Center(

child: Material(

child: Slider(

value: \_sliderValue,

onChanged: (newValue) {

setState(() {

\_sliderValue = newValue;

});

},

),

),

),

),

);

}