Flutter Scaffold Tutorial with Examples

1. Scaffold
2. Examples
3. floatingActionButton
4. floatingActionButtonLocation
5. floatingActionButtonAnimator
6. drawer
7. endDrawer
8. drawerDragStartBehavior
9. bottomNavigationBar
10. persistentFooterButtons
11. bottomSheet
12. backgroundColor
13. resizeToAvoidBottomPadding
14. resizeToAvoidBottomInset
15. primary

1. Scaffold

**Scaffold**class is an expandable widget that fills the available space or the screen. It provides an **API**to display the main widgets of the application such as **Drawer, SnackBar, Bottom-Sheet, FloatingActionButton, AppBar,**and **BottomNavigationBar**, etc.

**Scaffold** Constructor:

Scaffold constructor

**const** Scaffold({

Key key,

PreferredSizeWidget appBar,

Widget body,

Widget floatingActionButton,

FloatingActionButtonLocation floatingActionButtonLocation,

FloatingActionButtonAnimator floatingActionButtonAnimator,

List<Widget> persistentFooterButtons,

Widget drawer,

Widget endDrawer,

Widget bottomNavigationBar,

Widget bottomSheet,

Color backgroundColor,

bool resizeToAvoidBottomPadding,

bool resizeToAvoidBottomInset,

bool primary: **true**,

DragStartBehavior drawerDragStartBehavior: DragStartBehavior.down

})

2. Examples

main.dart (ex1)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'o7planning.org',

debugShowCheckedModeBanner: **false**,

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(title: 'Flutter Scaffold Example'),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key, **this**.title}) : **super**(key: key);

**final** String title;

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text(**this**.title),

),

body: Center(

child:

Text (

'Hello World',

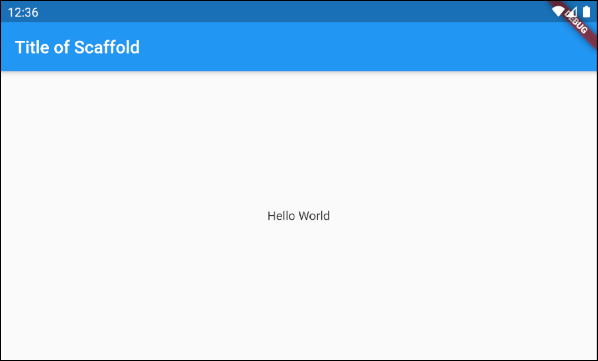
)

),

);

}

}



In this example, we have created a **Scaffold**with two parameters, **appBar** and **body**.

Scaffold (appBar + body)

// Create a Scaffold with 2 parameters: appBar, body.

Scaffold(

appBar: AppBar(

title: Text('Flutter Scaffold Example'),

),

body: Center(

child:

Text(

'Hello World',

)

),

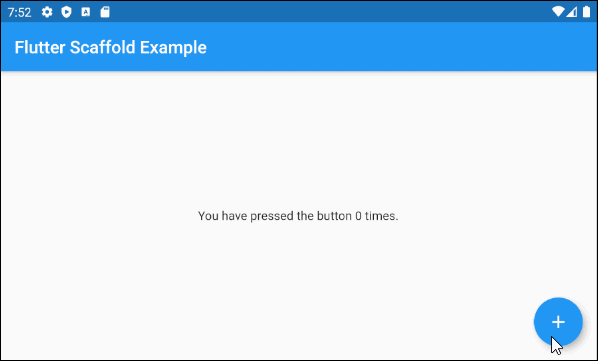
);

* [Flutter AppBar](https://o7planning.org/12851/flutter-appbar)

3. floatingActionButton

**floatingActionButton** is a button floating to the surface of the **body**. By default, it will float at the bottom right corner of the screen. You can specify its location by the **floatingActionButtonLocation**property.

Widget floatingActionButton



main.dart (floatingActionButton ex1)

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

**void** main() => runApp(MyApp());

// This Widget is the main application widget.

**class** **MyApp** **extends** **StatelessWidget** {

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: "o7planning.org",

debugShowCheckedModeBanner: **false**,

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatefulWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

MyHomePageState createState() => MyHomePageState();

}

**class** **MyHomePageState** **extends** **State**<**MyHomePage**> {

int \_count = 0;

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text('Flutter Scaffold Example'),

),

body: Center(

child: Text('You have pressed the button $\_count times.')

),

floatingActionButton: FloatingActionButton(

onPressed: () {

setState(() => **this**.\_count++);

},

tooltip: 'Increment Counter',

child: **const** Icon(Icons.add),

),

);

}

}

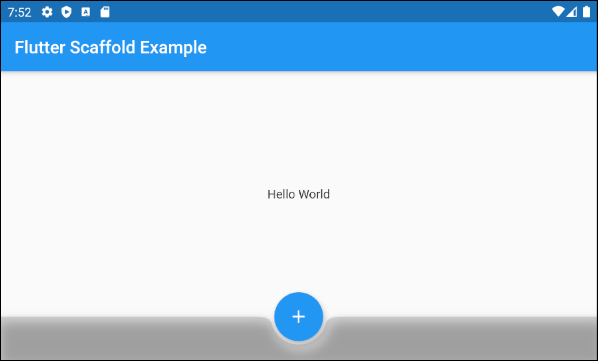
* Flutter FloatingActionButton

4. floatingActionButtonLocation

**floatingActionButtonLocation** property is used to specify the display position of the **floatingActionButton**. Its default value is **FloatingActionButtonLocation*.*endFloat**.

FloatingActionButtonLocation floatingActionButtonLocation

For example:



Scaffold (floatingActionButtonLocation ex1)

Scaffold(

appBar: AppBar(

title: Text('Flutter Scaffold Example'),

),

body: Center(

child: Text('Hello World')

),

bottomNavigationBar: BottomAppBar(

shape: CircularNotchedRectangle(),

color: Colors.black12,

child: Container(

height: 50.0,

),

),

floatingActionButton: FloatingActionButton(

onPressed: () {},

tooltip: 'Increment Counter',

child: Icon(Icons.add),

),

floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

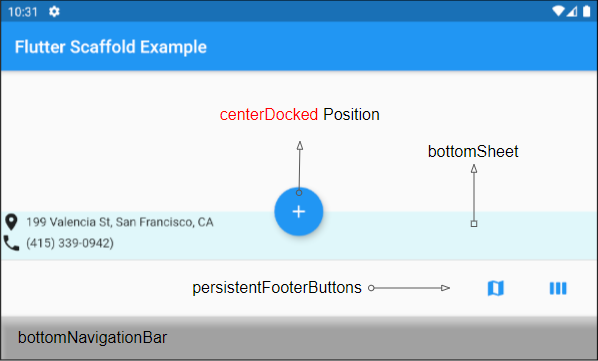
);

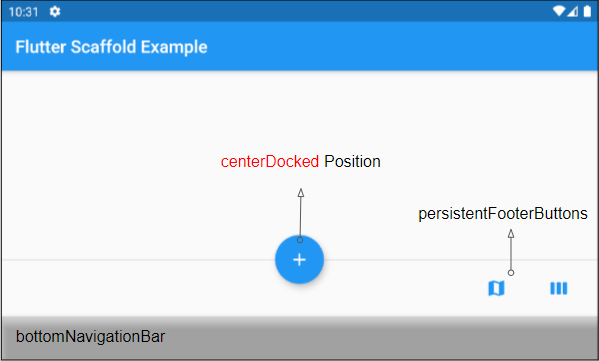
**FloatingActionButtonLocation**class allows you to define the display position of the **FloatingActionButton**. Here are several pre-defined positions as static constants of this class:

* FloatingActionButtonLocation.**centerDocked**
* FloatingActionButtonLocation.**centerFloat**
* FloatingActionButtonLocation.**centerTop**
* FloatingActionButtonLocation.**endDocked**
* FloatingActionButtonLocation.**endFloat**
* FloatingActionButtonLocation.**endTop**
* FloatingActionButtonLocation.**miniCenterDocked**
* FloatingActionButtonLocation.**miniCenterFloat**
* FloatingActionButtonLocation.**miniCenterTop**
* FloatingActionButtonLocation.**miniEndDocked**
* FloatingActionButtonLocation.**miniEndFloat**
* FloatingActionButtonLocation.**miniEndTo**
* FloatingActionButtonLocation.**miniStartDocked**
* FloatingActionButtonLocation.**miniStartFloat**
* FloatingActionButtonLocation.**miniStartTop**
* FloatingActionButtonLocation.**startDocked**
* FloatingActionButtonLocation.**startFloat**
* FloatingActionButtonLocation.**startTop**

\*Docked

**startDocked, centerDocked, endDocked, miniStartDocked, miniCenterDocked**and **miniEndDocked** constants allow **Scaffold.floatingActionButton** to be displayed on the **Scaffold.body**surface so that the center of the button is aligned with the top border of **Scaffold.bottomSheet** or **Scaffold.persistentFooterButtons**or **Scaffold.bottomNavigationBar** (In such the order of priority)





If **Scaffold.bottomSheet** and **Scaffold.persistentFooterButtons** are **null**, and **Scaffold.bottomNavigationBar** is a **BottomAppBar** object, **Scaffold.floatingActionButton** will create a notch on the surface of the **Scaffold.bottomNavigationBar**.

If Scaffold.bottomSheet and Scaffold.persistentFooterButtons are null, and Scaffold.bottomNavigationBar is a BottomAppBar object, Scaffold.floatingActionButton will create a notch on the surface of the Scaffold.bottomNavigationBar.

Flutter BottomAppBar Tutorial with Examples

1. BottomAppBar
2. child
3. shape
4. color
5. clipBehavior

1. BottomAppBar

A bottom menu is a traditional style of **iOS** apps. In **Flutter**, you can do this with **BottomAppBar**. Besides, the **BottomAppBar**also has a handy feature that allows you to attach a **FloatingActionButton**to it.

BottomAppBar Constructor:

BottomAppBar constructor

**const** BottomAppBar(

{Key key,

Color color,

double elevation,

NotchedShape shape,

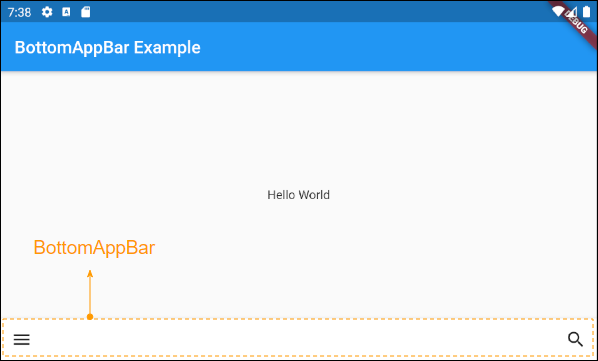
Clip clipBehavior: Clip.none,

double notchMargin: 4.0,

Widget child}

)

The **BottomAppBar**is usually placed in a **Scaffold**through the **AppBar.bottomNavigationBar**property, and it will appear at the bottom of the **Scaffold**.

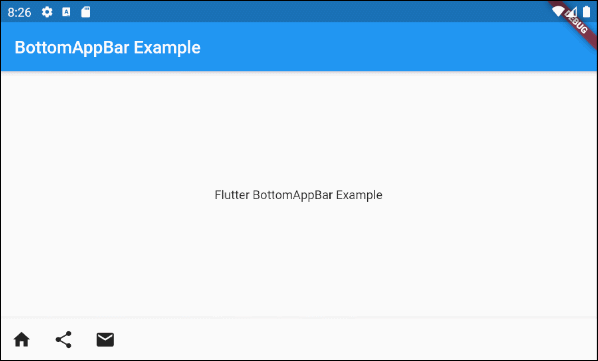


## 2. child

**child**is the most important property of the **BottomAppBar**. In most use cases, it will be a **Row**, and this **Row**object contains one or more child **Widget**(s)such as **IconButton**, **PopupMenuButton**,etc.

Widget child

An example of a **BottomAppBar** whose **action**(s) are **IconButton**(s) and **PopupMenuButton**(s).



main.dart (child - ex1)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

bottomNavigationBar: BottomAppBar(

child: **new** Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.start,

children: <Widget>[

IconButton(icon: Icon(Icons.home), onPressed: () {},),

PopupMenuButton(

icon: Icon(Icons.share),

itemBuilder: (context) => [

PopupMenuItem(

value: 1,

child: Text("Facebook"),

),

PopupMenuItem(

value: 2,

child: Text("Instagram"),

),

],

),

IconButton(icon: Icon(Icons.email), onPressed: () {},),

],

),

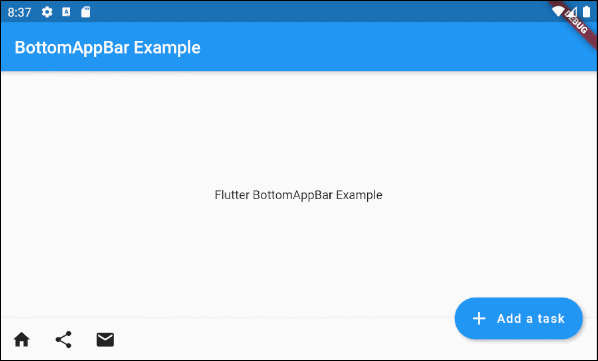
)

);

}

}

To extend the example above, we attach a **FloatingActionButton** to the right side of the **BottomAppBar**.



main.dart (child ex2)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

floatingActionButton: FloatingActionButton.extended (

elevation: 4.0,

icon: **const** Icon(Icons.add),

label: **const** Text('Add a task'),

onPressed: () {},

),

floatingActionButtonLocation: FloatingActionButtonLocation.endDocked,

bottomNavigationBar: BottomAppBar(

child: **new** Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.start,

children: <Widget>[

IconButton(icon: Icon(Icons.home), onPressed: () {},),

PopupMenuButton(

icon: Icon(Icons.share),

itemBuilder: (context) => [

PopupMenuItem(

value: 1,

child: Text("Facebook"),

),

PopupMenuItem(

value: 2,

child: Text("Instagram"),

),

],

),

IconButton(icon: Icon(Icons.email), onPressed: () {},),

],

),

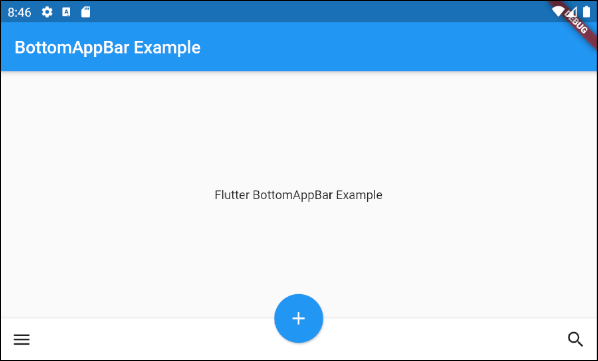
)

);

}

}

For example: A**BottomAppBar** along with a **FloatingActionButton** is anchored in the middle.



main.dart (child ex3)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

floatingActionButton: FloatingActionButton(

onPressed: () { },

tooltip: 'Increment',

child: Icon(Icons.add),

elevation: 2.0,

),

floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

bottomNavigationBar: BottomAppBar(

child: **new** Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: <Widget>[

IconButton(icon: Icon(Icons.menu), onPressed: () {},),

IconButton(icon: Icon(Icons.search), onPressed: () {},),

],

),

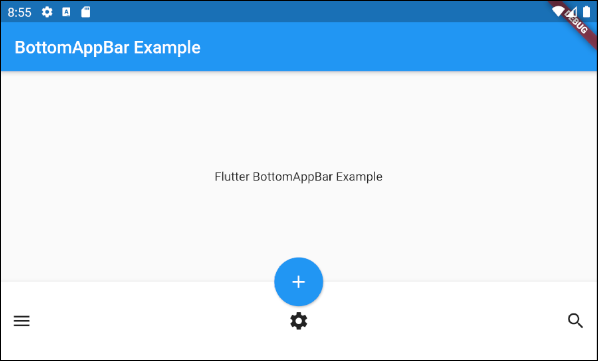
)

);

}

}

Example: Customize the height of the **BottomAppBar**:



main.dart (child ex4)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

floatingActionButton: FloatingActionButton(

onPressed: () { },

tooltip: 'Increment',

child: Icon(Icons.add),

elevation: 2.0,

),

floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

bottomNavigationBar: BottomAppBar(

child: Container(

height: 90.0,

child: **new** Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: <Widget>[

IconButton(icon: Icon(Icons.menu), onPressed: () {},),

IconButton(icon: Icon(Icons.settings), onPressed: () {},),

IconButton(icon: Icon(Icons.search), onPressed: () {},),

],

),

)

)

);

}

}

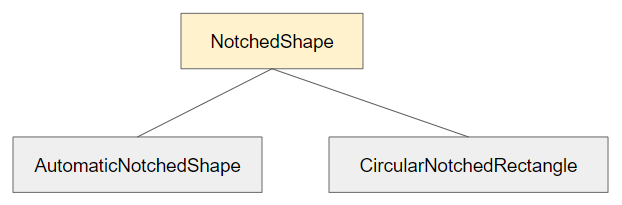
## 3. shape

**shape** property is used to define the shape of the notch when the **FloatingActionButton** is placed on a **BottomAppBar**.

NotchedShape shape

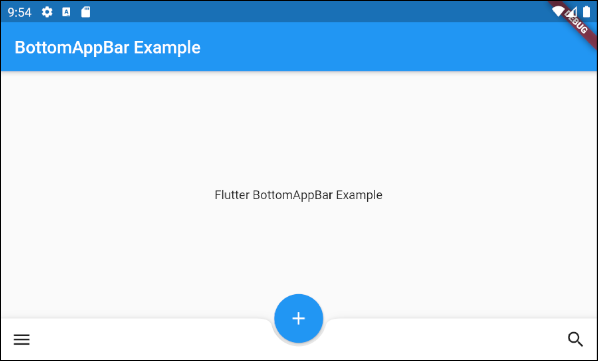
**NotchedShape** is an abstract class, it has two subclasses **CircularNotchedRectangle** and **AutomaticNotchedShape**.

* The **CircularNotchedRectangle** class helps you create a circular notch that fit the circular **FloatingActionButton**(s).
* The **AutomaticNotchedShape** class helps you create custom notches that fit the different shapes of the **FloatingActionButton**.



* CircularNotchedRectangle
* AutomaticNotchedShape

Now take a look at an example of using **CircularNotchedRectangle** to create a circular notch:



main.dart (shape ex1)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

floatingActionButton: FloatingActionButton(

onPressed: () { },

tooltip: 'Increment',

child: Icon(Icons.add),

elevation: 2.0,

),

floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

bottomNavigationBar: BottomAppBar(

child: Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: <Widget>[

IconButton(icon: Icon(Icons.menu), onPressed: () {},),

IconButton(icon: Icon(Icons.search), onPressed: () {},),

],

),

shape: CircularNotchedRectangle(),

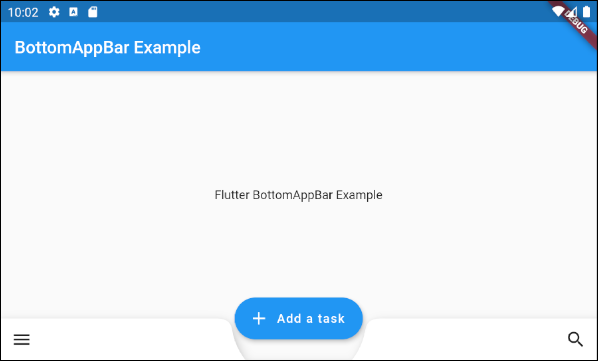
)

);

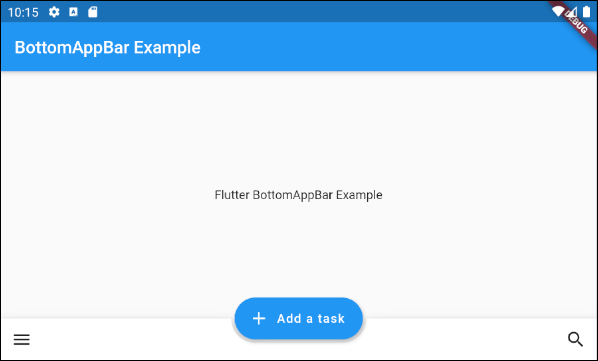
}

}

Note: Do not use the **CircularNotchedRectangle** with a **FloatingActionButton** which is not a circle. Otherwise, you will get a unsatisfactory result:



For example, use an**AutomaticNotchedShape** with a rectangular **FloatingActionButton**:



main.dart (shape ex3)

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

**void** main() {

runApp(MyApp());

}

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Title of Application',

theme: ThemeData(

primarySwatch: Colors.blue,

visualDensity: VisualDensity.adaptivePlatformDensity,

),

home: MyHomePage(),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key}) : **super**(key: key);

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text("BottomAppBar Example"),

),

body: Center(

child: Text(

'Flutter BottomAppBar Example',

)

),

floatingActionButton: FloatingActionButton.extended (

elevation: 4.0,

icon: **const** Icon(Icons.add),

label: **const** Text('Add a task'),

onPressed: () {},

),

floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

bottomNavigationBar: BottomAppBar(

child: Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: <Widget>[

IconButton(icon: Icon(Icons.menu), onPressed: () {},),

IconButton(icon: Icon(Icons.search), onPressed: () {},),

],

),

shape: AutomaticNotchedShape(

RoundedRectangleBorder(),

StadiumBorder(side: BorderSide())

),

)

);

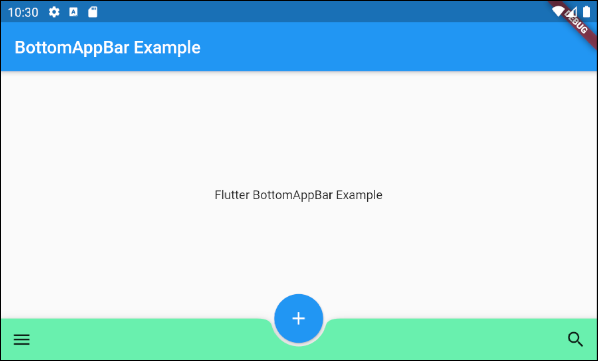
}

}

## 4. color

The **color** property is used to specify the color of **BottomAppBar**.

Color color



color (ex1)

BottomAppBar(

child: Row(

mainAxisSize: MainAxisSize.max,

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: <Widget>[

IconButton(icon: Icon(Icons.menu), onPressed: () {},),

IconButton(icon: Icon(Icons.search), onPressed: () {},),

],

),

shape: CircularNotchedRectangle(),

color: Colors.greenAccent

)

5. clipBehavior

Clip clipBehavior: Clip.none

* Flutter Clipping