

Mobile Application Flutter Layouts

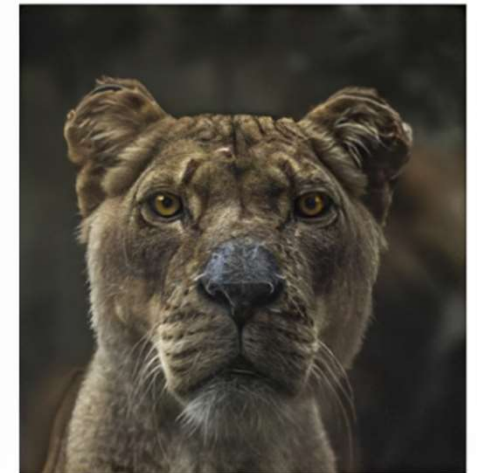
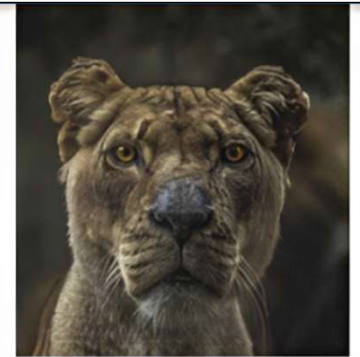
Chapter-6
Part-2

AspectRatio: This widget allows you to keep the size of the child widget to a specified aspect ratio.

```
AspectRatio(  
  aspectRatio: 5/3, // 5 =width and 3= height  
  child: Container(  
    color: Colors.blue,  
  ),  
),
```

Example

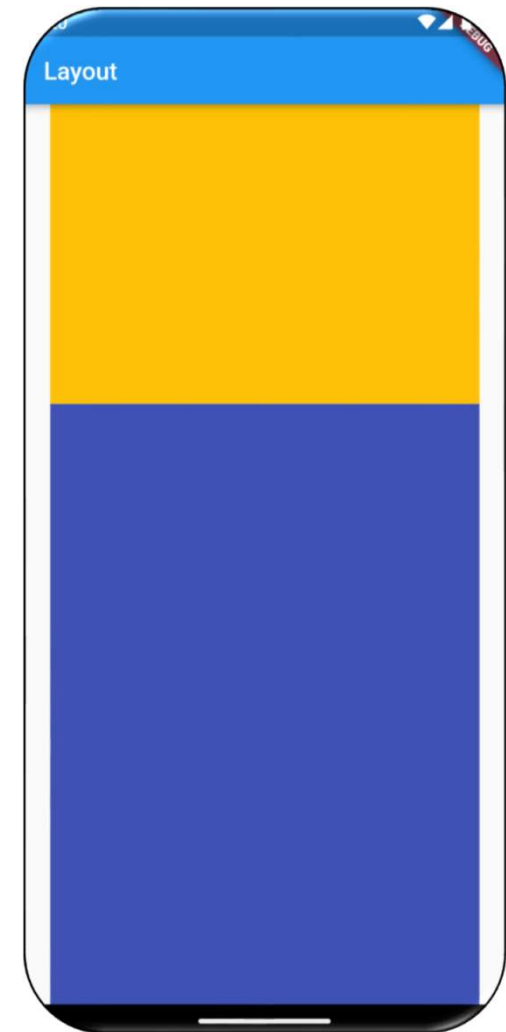
```
Column(  
  children: [  
    AspectRatio(  
      aspectRatio: 16 / 9,  
      child:  
Image.network('https://fastly.picsum.photos/id/1074/400/400.jpg?hmac=eH904qH8NQGitzB3QaCq9jrbDZr7KQkaW_w17w0uoGM'),  
    ),  
    SizedBox(height: 20), // Adding spacing between the two  
    AspectRatio widgets  
    AspectRatio(  
      aspectRatio: 4 / 3,  
      child:  
Image.network('https://fastly.picsum.photos/id/1074/400/400.jpg?hmac=eH904qH8NQGitzB3QaCq9jrbDZr7KQkaW_w17w0uoGM'),  
    ),  
  ],  
)
```



Expanded

- **Expanded:** It allows to make the children of a Row and Column widget to occupy the maximum possible area.

```
body: Center(  
  child: Column(  
    children: [  
      Expanded(  
        flex: 1,  
        child: Container(  
          height: 100,  
          width: 350,  
          color: Colors.amber,  
        ), ),  
      Expanded(  
        flex: 2,  
        child: Container(  
          height: 200,  
          width: 350,  
          color: Colors.indigo,  
        ), ),  
    ],  
  ),  
)
```



FittedBox

FittedBox is a very useful widget that scales and positions its child within itself according to **fit** and **alignment**.

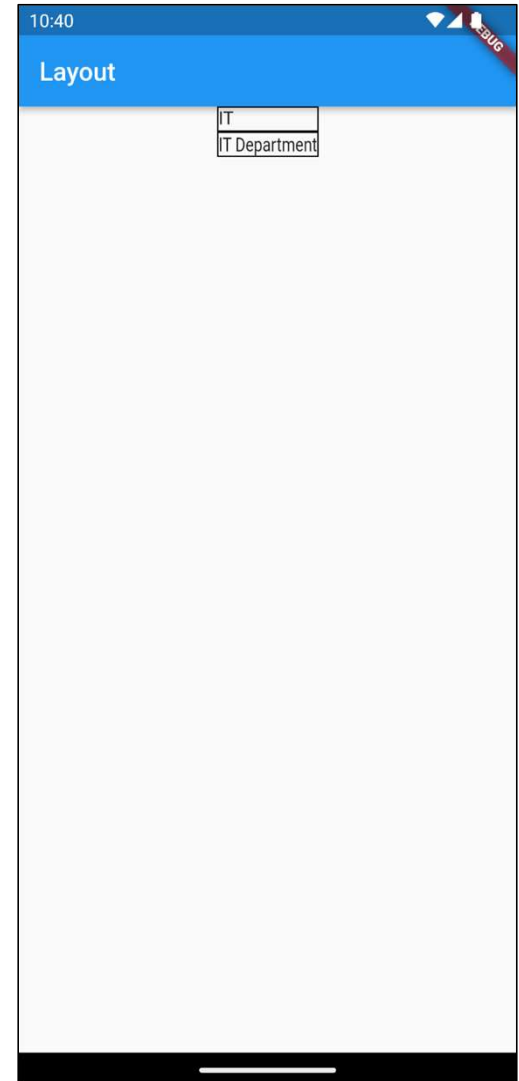
Properties:

1- Fit: BoxFit.(fill , cover , none , fitHeight , fitWidth, contain)

2- alignment: Alignment.(bottomCenter, bottomLeft, bottomRight, center, centerLeft, centerRight, topCenter, topLeft, topRight)

Example

```
body: Center(  
  child: Container(  
    child: Column(  
      children: [  
  
        Container( height: 20, width: 80,  
          child: Text("IT Department"),  
          decoration: BoxDecoration(  
            border: Border.all(),  
          ),),  
  
        Container( height: 20, width: 80,  
          child: FittedBox(  
            alignment: Alignment.topCenter,  
            fit: BoxFit.fill,  
            child: Text("IT Department")),  
          decoration: BoxDecoration(  
            border: Border.all(),  
          ),  
        ),  
      ],),),),),
```



Stack and Positioned

Stack

The stack is a widget in Flutter that contains a list of widgets and positions them on top of the other. In other words, the stack allows developers ***to overlap multiple widgets into a single screen*** and renders them from bottom to top.

- The child widget in a stack can be either **positioned** or **non-positioned**.
- The non-positioned child widgets are aligned itself. It displays on the screen based on the stack's alignment. The default position of the children is in the top left corner.
- We can use the **alignment** attribute to change the alignment of the widgets.

Stack and Positioned

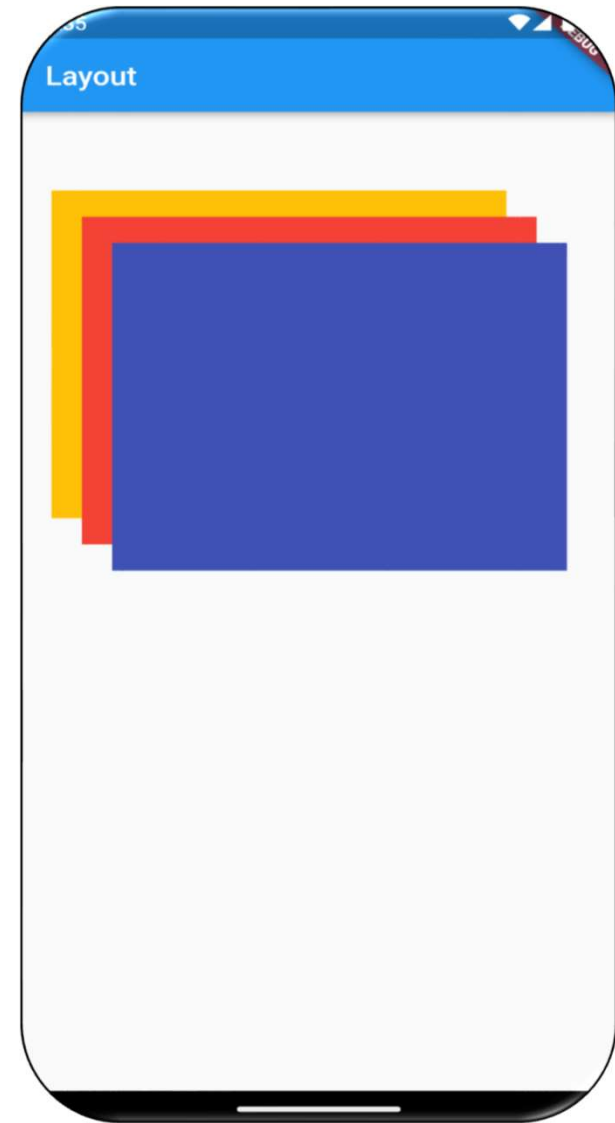
Positioned

It is not the stack parameter but can be used in the stack to locate the children widgets. The following are the constructor of the positioned stack:

```
Positioned({  
  Key key,  
  this.left,  
  this.top,  
  this.right,  
  this.bottom,  
  this.width,  
  this.height,  
  @required Widget child,  
})
```


Example

```
body: Stack(  
  children: [  
    Positioned(  
      top: 60,  
      left: 20,  
      height: 250,  
      width: 300,  
      child: Container(  
        color: Colors.amber,  
      )),  
    Positioned(  
      top: 80,  
      left: 40,  
      height: 250,  
      width: 300,  
      child: Container(  
        color: Colors.red,  
      )),  
    Positioned(  
      top: 100,  
      left: 60,  
      height: 250,  
      width: 300,  
      child: Container(  
        color: Colors.indigo,  
      )),  
  ],  
),
```



GridView

GridView is a widget in Flutter that displays the items in a **2-D array** (two-dimensional rows and columns). As the name suggests, it will be used when we want to show items in a Grid. We can select the desired item from the grid list by tapping on them. This widget can contain **text, images, icons**, etc. to display in a grid layout depending on the user requirement.

Grid

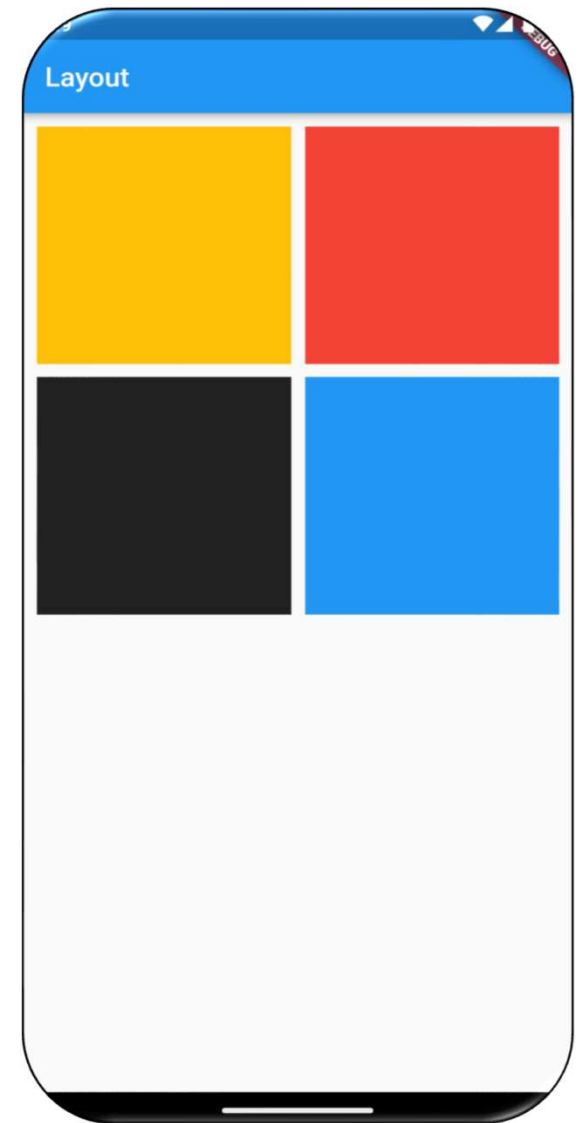
- **crossAxisCount:** It is used to specify the number of columns in a grid view.
- **crossAxisSpacing:** It is used to specify the number of pixels between each child widget listed in the cross axis.
- **mainAxisSpacing:** It is used to specify the number of pixels between each child widget listed in the main axis.
- **padding(EdgeInsetsGeometry):** It is used to specify the space around the whole list of widgets.
- **scrollDirection:** It is used to specify the direction in which the items on GridView scrolls. By default, it scrolls in a vertical direction.

```
scrollDirection: Axis.horizontal,
```

- **reverse:** If it is true, it will reverse the list in the opposite direction along the main axis.

Example 1

```
body: Container(  
  child: GridView(  
    padding: EdgeInsets.all(10),  
    //scrollDirection: Axis.horizontal,  
    // reverse: true,  
  
    children: [  
      Container(color: Colors.amber,),  
      Container(color: Colors.red,),  
      Container(color: Colors.black87,),  
      Container(color: Colors.blue,),  
    ],  
    gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(  
  
      crossAxisCount: 2,  
      mainAxisSpacing: 10,  
      crossAxisSpacing: 10,  
    ),  
  ),  
),
```



Example 2

```
body: Container(  
  child: GridView(  
    padding: EdgeInsets.all(10),  
    children: [  
      Container(  
        decoration: BoxDecoration(  
          color: Colors.black87,  
          borderRadius: BorderRadius.circular(25),  
        ),  
        child: Column(  
          mainAxisAlignment: MainAxisAlignment.center,  
          children: [  
            Icon(  
              Icons.mail,  
              color: Colors.greenAccent[200],  
              size: 70,  
            ),  
            Text("Mail", style: TextStyle(  
              fontSize: 24, color: Colors.greenAccent[200]),  
            ),  
          ],  
        ),  
      ),  
    ],  
  ),  
),
```

