

Duhok Polytechnic University

Technical College of Zakho

Department of CIS



LAB 7: Visual Identity

MOBILE APPLICATION

Objectives

Separators

Indicators

Iconography

Asset Registration

Image Sources

Visual Scaling

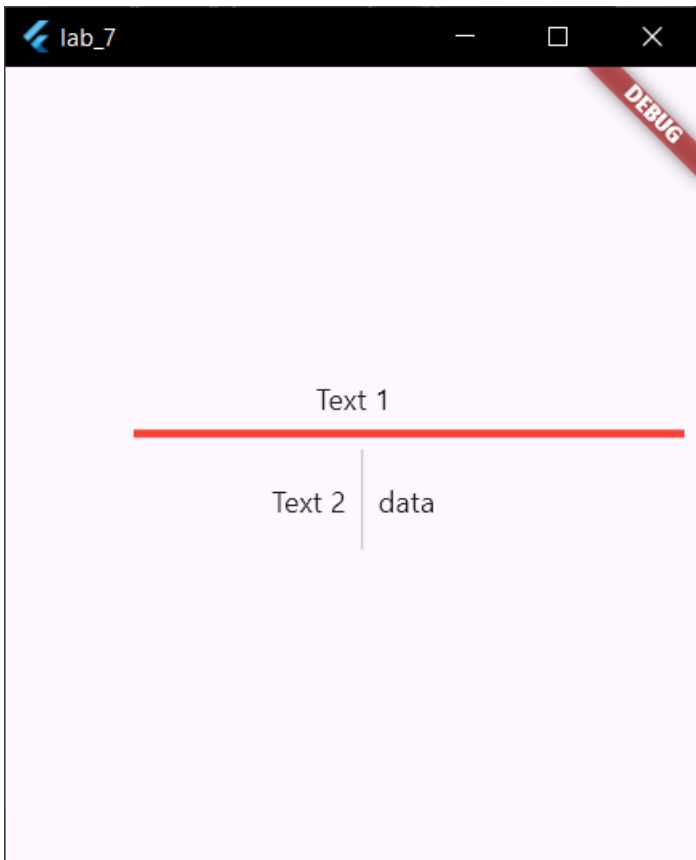
Dividers

When building lists or organizing complex screens, we often need to draw a physical line between elements to group them logically. Flutter provides the `Divider` (for horizontal lines in a `Column`) and the `VerticalDivider` (for vertical lines in a `Row`).

Key Attributes:

- `thickness`: How thick the actual drawn line is (in pixels).
- `color`: The color of the line (e.g., `Colors.grey.shade300`).
- `indent`: The amount of empty space before the line starts (left side for `Divider`, top side for `VerticalDivider`).
- `endIndent`: The amount of empty space after the line ends.

Dividers Example



```
1 Column(  
2   mainAxisAlignment: MainAxisAlignment.center,  
3   children: [  
4     Text("Text 1"),  
5     Divider(thickness: 4, color: Colors.red, indent: 64, endIndent: 8),  
6     SizedBox(  
7       height: 50,  
8       child: Row(  
9         mainAxisAlignment: MainAxisAlignment.center,  
10        children: [  
11          Text("Text 2"),  
12          VerticalDivider(),  
13          Text("data")  
14        ],  
15      ),  
16    ),  
17  ],  
18 )
```

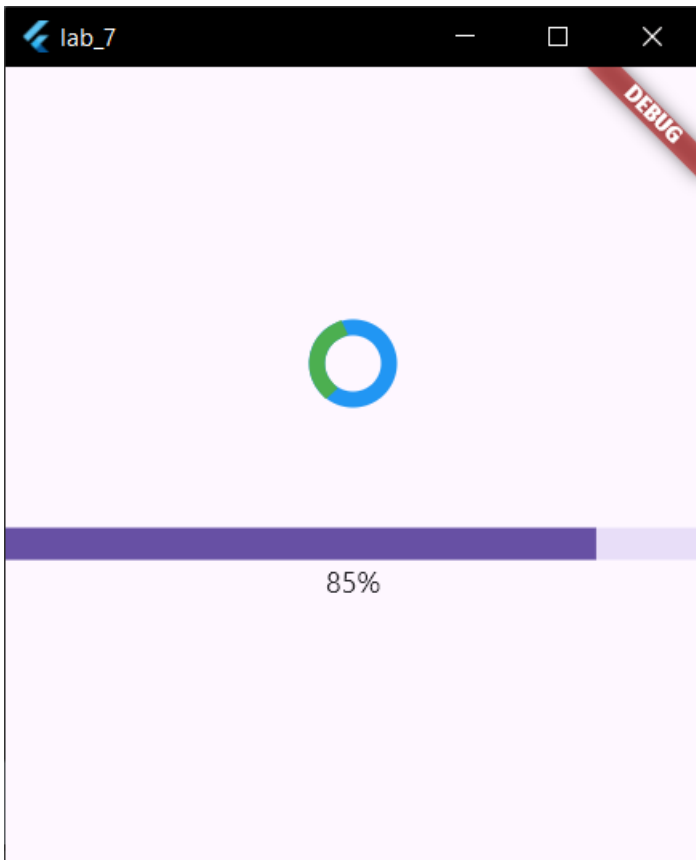
Progress Indicators

Whenever your app is fetching data, saving a file, or processing information, you must show the user that the app hasn't frozen. Flutter provides two main widgets for this: the `CircularProgressIndicator` (a spinning wheel) and the `LinearProgressIndicator` (a loading bar).

Key Attributes:

- `color`: The color of the active, moving part of the indicator.
- `backgroundColor`: The color of the track behind the moving part.
- `value`: This is the magic attribute!
 - If value is null (which is the default), the indicator spins or slides infinitely (called Indeterminate).
 - If value is a decimal between 0.0 and 1.0, it acts as a precise completion bar (called Determinate). E.g., value: 0.5 means 50% complete.
- `strokeWidth`: (Circular only) How thick the spinning circle is.
- `minHeight`: (Linear only) How thick the loading bar is.

Indicators Example



```
1 Column(  
2   mainAxisAlignment: MainAxisAlignment.center,  
3   children: [  
4     CircularProgressIndicator(  
5       strokeWidth: 8,  
6       backgroundColor: Colors.blue,  
7       color: Colors.green,  
8     ),  
9     SizedBox(height: 64),  
10    LinearProgressIndicator(value: .85, minHeight: 16),  
11    Text("85%"),  
12  ],  
13 )
```

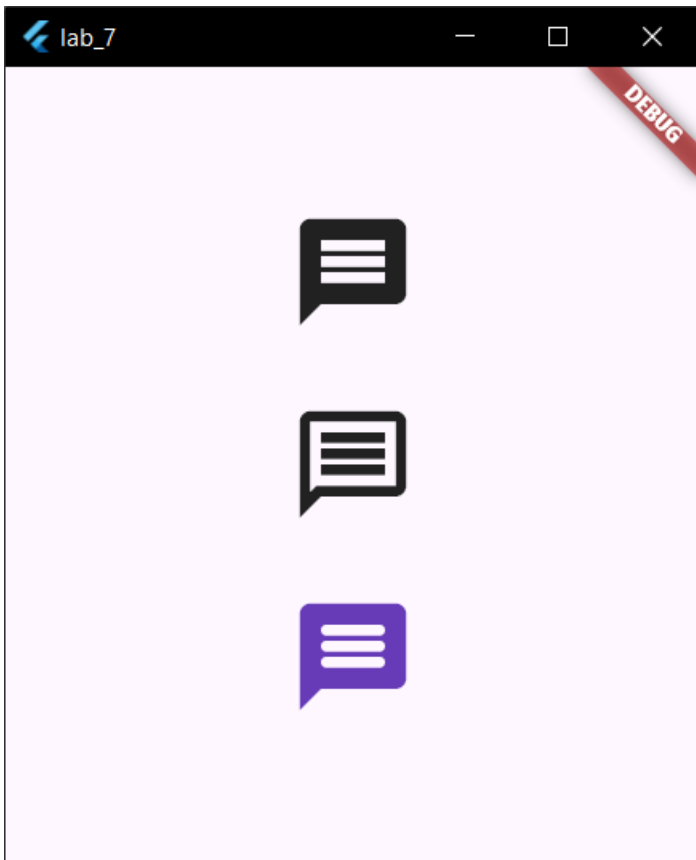
Icon

Icons are lightweight, scalable vector graphics that act as a universal visual language for your users. Instead of downloading separate images for things like a "Home" button or a "Search" glass, Flutter includes the entire Material Design Icon Library completely out of the box. Because they are drawn like text fonts, they scale perfectly without ever becoming blurry.

Key Attributes:

- **icon:** This is the actual shape you want to display. It is the only required positional argument. You access the library using the Icons class (e.g., `Icons.favorite`, `Icons.settings`, `Icons.home`).
- **color:** Changes the color of the icon. If you don't set this, it defaults to the icon theme's color (usually black or white depending on dark mode).
- **size:** A double value that determines how large the icon is in logical pixels (the default is usually 24.0).

Icon Example



```
1 Center(  
2   child: Column(  
3     mainAxisAlignment: MainAxisAlignment.center,  
4     children: [  
5       Icon(Icons.message, size: 64),  
6       SizedBox(height: 32),  
7       Icon(Icons.message_outlined, size: 64),  
8       SizedBox(height: 32),  
9       Icon(  
10        Icons.message_rounded,  
11        color: Colors.deepPurple,  
12        size: 64,  
13      ),  
14    ],  
15  ),  
16 )
```

Image

The Image widget is used to display pictures in your application. While you can display images from memory or files, the two most common methods for mobile apps are fetching an image directly from the internet (Network Image) or loading an image bundled directly inside your app's code (Asset Image).

Key Constructors & Attributes


- `Image.network('url')`: Loads an image from a web address. It requires an active internet connection but keeps your app's file size small.
- `Image.asset('path')`: Loads a local image stored in your project (e.g., `assets/logo.png`). Must be registered in `pubspec.yaml` first!
- `width / height`: Forces the image to be a specific size. If left blank, the image tries to display at its original resolution.
- `fit`: Uses the `BoxFit` class to determine how the image should resize if it doesn't perfectly match the width/height of its container.
 - `BoxFit.cover`: Stretches the image to fill the box, cropping the edges if necessary.
 - `BoxFit.contain`: Shrinks the image so the entire picture is visible, leaving empty space if the aspect ratio doesn't match.

Configuring Assets

Before `Image.asset` can display anything, Flutter needs to know which files to permanently bundle into your app. You do this by registering the files in your project's master configuration file: `pubspec.yaml`. YAML files rely entirely on exact spacing—if your indentation is off by even one space, the app will crash!

Setup:

1. Create the Folder: Right-click your main project folder and create a new folder named `assets`. Place your images inside (e.g., `dpulogo.png`).
2. Open `pubspec.yaml`: Scroll down to find the `flutter:` section (usually around line 40-60).
3. Register the Path: Uncomment or add the `assets:` tag, and list the exact path to your images underneath it.



```
1 flutter:
2   uses-material-design: true
3
4   assets:
5     - assets/dpulogo.png
6     # or just assets/
```

Image Example



```
1 Center(  
2   child: Column(  
3     mainAxisAlignment: MainAxisAlignment.center,  
4     children: [  
5       Image.asset(  
6         'assets/dpulogo.png',  
7         width: 128,  
8         fit: BoxFit.contain,  
9       ),  
10      SizedBox(height: 16),  
11      Image.network(  
12        //Requires internet connection  
13        'https://picsum.photos/200',  
14        width: 150,  
15        height: 150,  
16        fit: BoxFit.cover,  
17      ),  
18    ],  
19  ),  
20 )
```

Any Questions?



Lab Work

