# Widget Types in Dart (Flutter)

## 1. Stateless Widgets

Stateless widgets are immutable, meaning their properties can't change once they are built. They are ideal for UI components that do not require dynamic updates.  
  
Example:  
```dart  
import 'package:flutter/material.dart';  
  
class MyStatelessWidget extends StatelessWidget {  
 @override  
 Widget build(BuildContext context) {  
 return Center(  
 child: Text(  
 'Hello, World!',  
 style: TextStyle(fontSize: 24),  
 ),  
 );  
 }  
}  
  
void main() {  
 runApp(MaterialApp(home: MyStatelessWidget()));  
}```

## 2. Stateful Widgets

Stateful widgets maintain a mutable state. They are ideal for components that need to dynamically update based on user interaction or other changes.  
  
Example:  
```dart  
import 'package:flutter/material.dart';  
  
class MyStatefulWidget extends StatefulWidget {  
 @override  
 \_MyStatefulWidgetState createState() => \_MyStatefulWidgetState();  
}  
  
class \_MyStatefulWidgetState extends State<MyStatefulWidget> {  
 int \_counter = 0;  
  
 void \_incrementCounter() {  
 setState(() {  
 \_counter++;  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: Text('Stateful Widget Example')),  
 body: Center(  
 child: Text(  
 'Counter: $\_counter',  
 style: TextStyle(fontSize: 24),  
 ),  
 ),  
 floatingActionButton: FloatingActionButton(  
 onPressed: \_incrementCounter,  
 child: Icon(Icons.add),  
 ),  
 );  
 }  
}  
  
void main() {  
 runApp(MaterialApp(home: MyStatefulWidget()));  
}```

## 3. Other Widget Categories

### a. Layout Widgets  
Used to arrange other widgets on the screen.  
  
Examples:  
```dart  
Column(  
 children: [  
 Text('Item 1'),  
 Text('Item 2'),  
 Text('Item 3'),  
 ],  
)  
```  
  
### b. Input Widgets  
Used to get user input.  
  
Examples:  
```dart  
TextField(  
 decoration: InputDecoration(labelText: 'Enter your name'),  
)  
```  
  
### c. Interaction Widgets  
Allow users to interact with the app.  
  
Examples:  
```dart  
GestureDetector(  
 onTap: () {  
 print("Tapped!");  
 },  
 child: Text('Click Me'),  
)  
```  
  
### d. Styling Widgets  
Modify appearance and styles.  
  
Examples:  
```dart  
Padding(  
 padding: EdgeInsets.all(16.0),  
 child: Text('Padded Text'),  
)  
```  
  
### e. Animation Widgets  
Create animations and transitions.  
  
Examples:  
```dart  
AnimatedContainer(  
 duration: Duration(seconds: 1),  
 color: Colors.blue,  
 width: 100,  
 height: 100,  
)  
```

## Using Widgets in Visual Studio Code

1. \*\*Code IntelliSense\*\*: Use the built-in IntelliSense in VS Code for auto-completion and widget suggestions.  
2. \*\*Hot Reload\*\*: While running your app, make changes to the code and see updates instantly without restarting the app.  
3. \*\*Widget Inspector\*\*: Use the Flutter DevTools or the widget inspector in VS Code to explore widget hierarchies.