day-8-LiB

Introduction to Flutter Development: A Beginner's Guide

Overview of Today

- [ ] Understanding Flutter and its architecture
- [ ] Widget system and types
- [ ] Basic app structure
- [ ] Essential widgets for UI building
- [ ] Building your first Flutter UI

Resources that'll help you

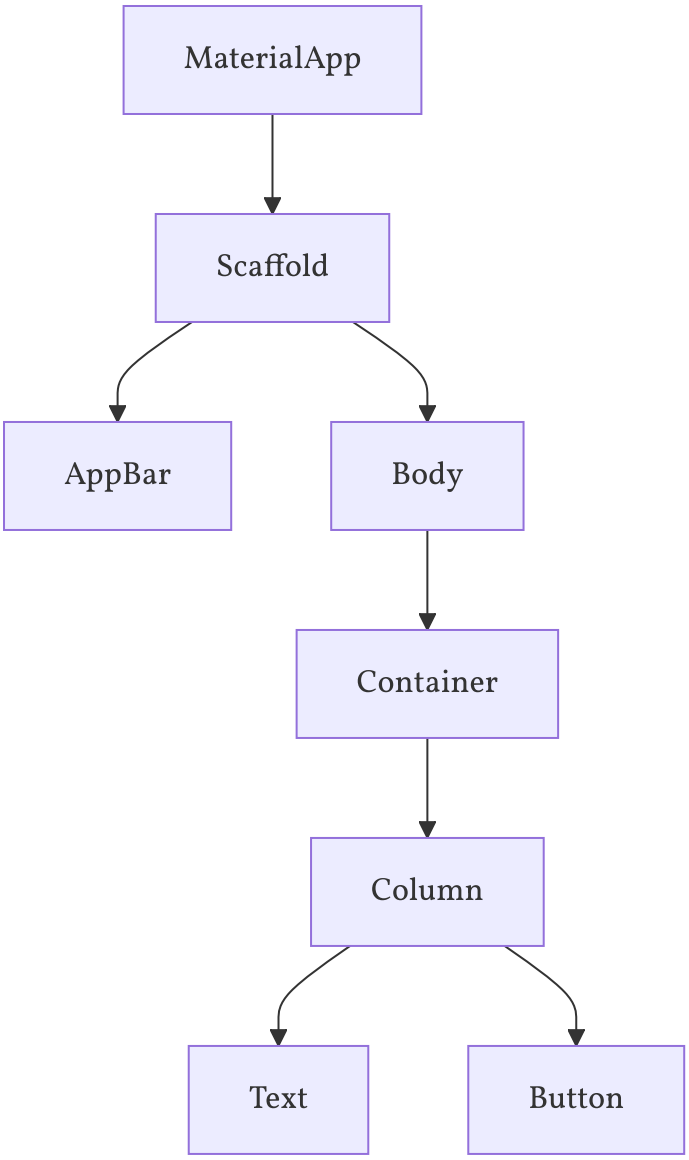
- Flutter Documentation - Official documentation
- Flutter in 100 Seconds - Quick overview
- DartPad with Flutter - Online Flutter playground
- Flutter Widget Catalog - Widget reference

Understanding Flutter

Flutter is Google's UI frame for building natively compiled applications from a single codebase. It uses the Dart programming language and a widget-based approach to UI development.

Core Concepts

1. **Everything is a Widget**:
    - UI elements are widgets
    - Layout elements are widgets
    - Even the app itself is a widget
2. **Widget Tree**:



1. **Types of Widgets**:
    - **Stateless**: Immutable, can't change their properties
    - **Stateful**: Can maintain state and rebuild when needed
    - **Inherited**: Share data down the widget tree

Installation

- Use This Guide to Install Flutter SDK (Dart SDK included) on your System.
- On Vscode, Install the Flutter Extension

Building Your First Flutter App

Step 1: Project Setup

2. Create a new Flutter project:

```
flutter create my_first_app
cd my_first_app
```

3. Project structure explained:

```
my_first_app/
├── lib/
│   ├── main.dart          # Entry point
│   └── screens/           # Your app screens
├── pubspec.yaml           # Dependencies
└── test/                  # Test files
```

Step 2: Understanding main.dart

Let's break down a basic Flutter app:

```dart
// 1. Import necessary packages
import 'package:flutter/material.dart';

// 2. Main function - Entry point
void main() {
  runApp(const MyApp());
}

// 3. Root widget
class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'My First App',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: const MyHomePage(),
```

```
    );
  }
}

// 4. Home page widget
class MyHomePage extends StatelessWidget {
  const MyHomePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('My First App'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: const [
            Text('Welcome to Flutter!'),
          ],
        ),
      ),
    );
  }
}
```

Step 3: Essential Widgets Explained

Layout Widgets

```
// Container - Like a div in web development
Container(
  padding: EdgeInsets.all(16.0),
  margin: EdgeInsets.symmetric(vertical: 8.0),
  decoration: BoxDecoration(
    color: Colors.white,
    borderRadius: BorderRadius.circular(8.0),
    boxShadow: [
      BoxShadow(
        color: Colors.grey.withOpacity(0.5),
        spreadRadius: 2,
        blurRadius: 5,
      ),
    ],
  ),
```

```
  child: Text('Hello!'),
)

// Row - Horizontal layout
Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    Icon(Icons.star),
    Text('4.5'),
    Text('Reviews'),
  ],
)

// Column - Vertical layout
Column(
  crossAxisAlignment: CrossAxisAlignment.start,
  children: [
    Text('Title'),
    Text('Subtitle'),
    Text('Description'),
  ],
)
```

Common UI Widgets

```
// Buttons
ElevatedButton(
  onPressed: () {
    print('Button pressed!');
  },
  child: Text('Click Me'),
)

// Text Input
TextField(
  decoration: InputDecoration(
    labelText: 'Enter your name',
    border: OutlineInputBorder(),
    prefixIcon: Icon(Icons.person),
  ),
)

// Images
Image.network(
  'https://placehold.co/600x400/png',
```

```
  width: 200,
  height: 200,
  fit: BoxFit.cover,
)
```

Step 4: Building a Complete Screen

Let's build a profile card screen:

```
class ProfileScreen extends StatelessWidget {
  const ProfileScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Profile'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          children: [
            // Profile Image
            CircleAvatar(
              radius: 50, // How Rounded are the Corners
              backgroundImage: NetworkImage(
                 'https://placehold.co/600x400/png'
              ),
            ),
            // Spacing (There are multiple ways, this is the simplest)
            const SizedBox(height: 16),

            // Name
            const Text(
              'Abbas Ibn Firnas',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
              ),
            ),

            const SizedBox(height: 8),

            // Bio
            const Text(
```

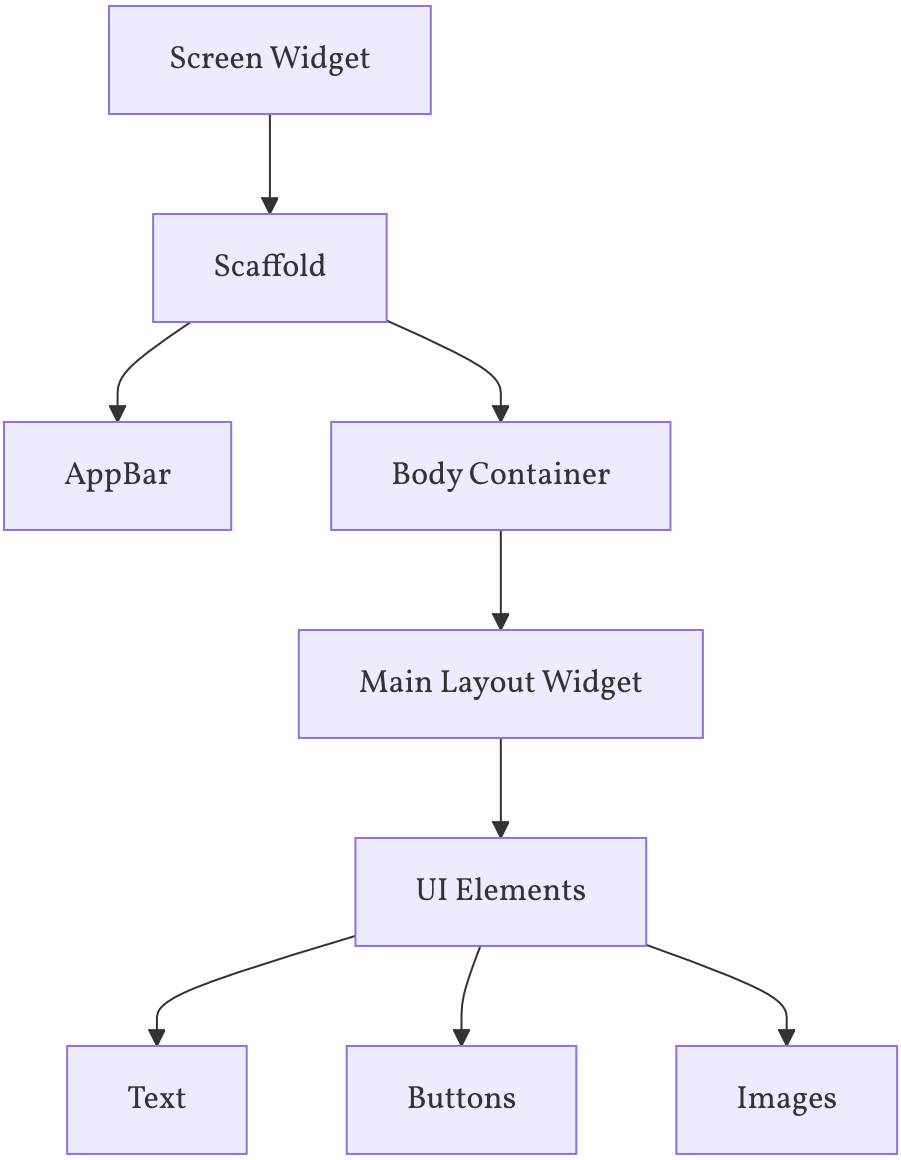
```
            'Flutter Developer',
            style: TextStyle(
              fontSize: 16,
              color: Colors.grey,
            ),
          ),

          const SizedBox(height: 16),

          // Stats Row
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: [
              _buildStat('Posts', '255'),
              _buildStat('Followers', '12k'),
              _buildStat('Following', '420'),
            ],
          ),

          const SizedBox(height: 16),

          // Edit Profile Button
          ElevatedButton(
            onPressed: () {},
            child: const Text('Edit Profile'),
          ),
        ],
      ),
    ),
  );
}

Widget _buildStat(String label, String value) {
  return Column(
    children: [
      Text(
        value,
        style: const TextStyle(
          fontSize: 20,
          fontWeight: FontWeight.bold,
        ),
      ),
      Text(
        label,
        style: const TextStyle(color: Colors.grey),
      ),
```

```
      ],
    );
  }
}
```

Tasks

- [ ] Create a new Flutter project
- [ ] Modify the `main.dart` file to understand the basic structure
- [ ] Build a simple profile screen using the widgets we learned
- [ ] Experiment with different layouts (Row, Column)
- [ ]
    - Optional: Try to build any basic UI View of your liking (e.g. a login Screen)

Common Widget Patterns



> **Note**
>
> Always wrap content in a Scaffold widget for material design structure

> **Tip**
>
> Use const constructors when possible for better performance

Debugging Tips

- Use `flutter run` to run your app, or preferably `Ctrl F5` if Flutter Extension is installed
- Hot reload (r) updates UI changes instantly
- Hot restart (R) resets app state

- Use Flutter DevTools for detailed debugging