

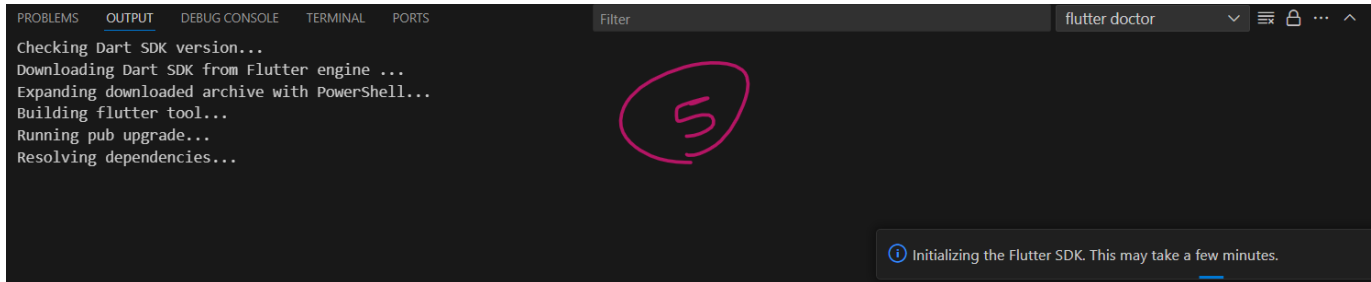
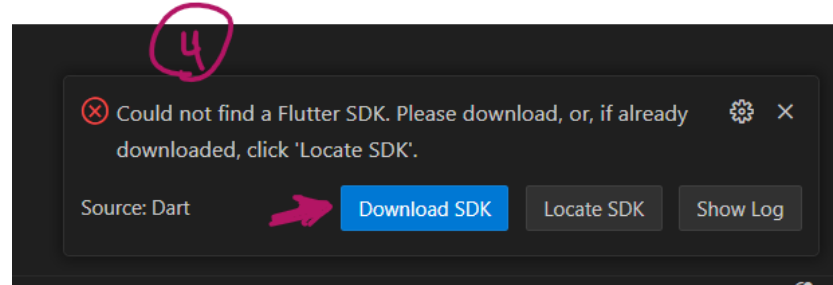
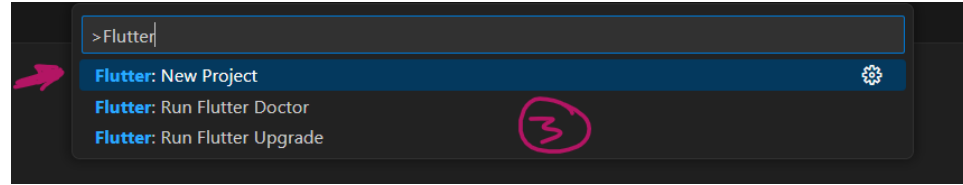
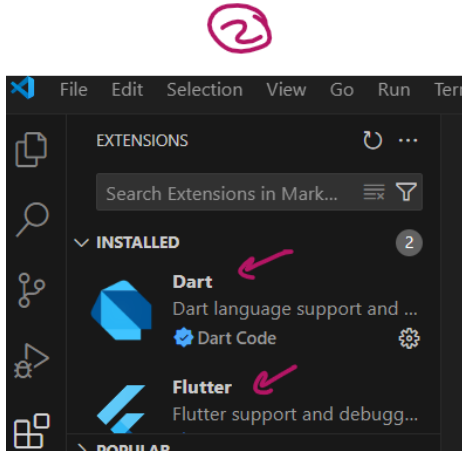
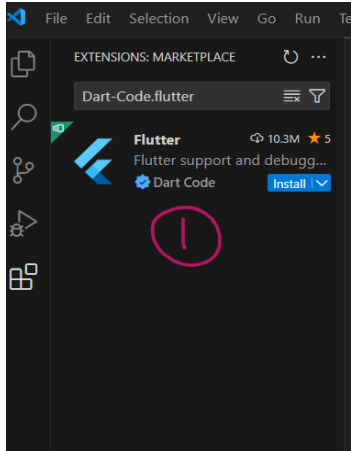
CET215: Mobile Application Development

Lecture 3: Flutter Intro





Install Flutter SDK + Create New Project





Check any missing requirements (flutter doctor)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.29.0, on Microsoft Windows [Version 10.0.26100.2894], locale en-GB)
[✓] Windows Version (11 Pro 64-bit, 24H2, 2009)
[X] Android toolchain - develop for Android devices
    X Unable to locate Android SDK.
      Install Android Studio from: https://developer.android.com/studio/index.html
      On first launch it will assist you in installing the Android SDK components.
      (or visit https://flutter.dev/to/windows-android-setup for detailed instructions).
      If the Android SDK has been installed to a custom location, please use
      `flutter config --android-sdk` to update to that location.
```

```
[✓] Chrome - develop for the web
[!] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.12.4)
    X Visual Studio is missing necessary components. Please re-run the Visual Studio installer for the "Desktop development with C++" workload,
      components:
        MSVC v142 - VS 2019 C++ x64/x86 build tools
          - If there are multiple build tool versions available, install the latest
        C++ CMake tools for Windows
        Windows 10 SDK
[!] Android Studio (not installed)
[✓] VS Code (version 1.97.2)
[✓] Connected device (3 available)
[✓] Network resources
```





Download Android Studio

The screenshot shows a web browser at the URL <https://developer.android.com/studio>. The page features a navigation bar with 'Developers', 'Essentials', 'Design & Plan', 'Develop' (highlighted), and 'Google Play'. Below this, the 'ANDROID STUDIO' section has tabs for 'Download', 'Android Studio editor', 'Gemini in Android Studio', and 'Android Gradle Plug'. The main content area has a dark background with the text 'Android Studio' in large white font, followed by 'The official IDE for Android app development now accelerates your productivity with Gemini in Android Studio, your AI-powered coding companion.' At the bottom, a white button contains the text 'Download Android Studio Ladybug Feature Drop' and a download icon.

← ↻ 🔒 <https://developer.android.com/studio>

lexfridman/mit-dee... kpdecker/jsdiff: A ja... Using EPPlus To Imp... sandeepdillerao/cor... Telegram: c


Developers Essentials ▾ Design & Plan ▾ **Develop** ▾ Google Play

ANDROID STUDIO

Download Android Studio editor Gemini in Android Studio Android Gradle Plug

Android Studio

The official IDE for Android app development now accelerates your productivity with Gemini in Android Studio, your AI-powered coding companion.

Download Android Studio Ladybug Feature Drop 





Check any missing requirements (flutter doctor)

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

Doctor summary (to see all details, run flutter doctor -v):

[✓] Flutter (Channel stable, 3.29.0, on Microsoft Windows [Version 10.0.26100.2894], locale en-GB)

[✓] Windows Version (11 Pro 64-bit, 24H2, 2009)

[!] Android toolchain - develop for Android devices (Android SDK version 35.0.1)

X cmdline-tools component is missing

Run ``path/to/sdkmanager --install "cmdline-tools;latest"``

See <https://developer.android.com/studio/command-line> for more details.

X Android license status unknown.

Run ``flutter doctor --android-licenses`` to accept the SDK licenses.

See <https://flutter.dev/to/windows-android-setup> for more details.

[✓] Chrome - develop for the web

[!] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.12.4)





missing cmd line

✅ Fix the Issue by Installing cmdline-tools

Option 1: Install via SDK Manager (Recommended)

1 Open Android SDK Manager:

- Open Android Studio.
- Go to Settings → Appearance & Behavior → System Settings → Android SDK.
- Click on the SDK Tools tab.

2 Enable Command-Line Tools:

- Check "Android SDK Command-line Tools (latest)".
- Click Apply → OK to install.

The screenshot shows the 'Settings' window in Android Studio, specifically the 'Languages & Frameworks' section under 'Android SDK'. The 'SDK Tools' tab is selected, showing a list of available SDK developer tools. The 'Android SDK Command-line Tools (latest)' are checked for installation. An 'SDK Quickfix Installation' dialog is also visible in the foreground, showing the 'SDK Component Installer' and the progress of the installation.

Settings

Appearance & Behavior

Keymap

Editor

Plugins

Build, Execution, Deployment

Languages & Frameworks

Android SDK

Kotlin

Tools

Advanced Settings

Languages & Frameworks > Android SDK

Revert changes

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: C:\Users\ahmed\AppData\Local\Android\Sdk

Edit Optimize disk space

SDK Platforms

SDK Tools

SDK Update Sites

Below are the available SDK developer tools. Once installed, the IDE will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Vers...	Status
<input checked="" type="checkbox"/> Android SDK Build-Tools 36-rc5		Update Available: 36.0.0 rc5
<input type="checkbox"/> NDK (Side by side)		Not Installed
<input checked="" type="checkbox"/> Android SDK Command-line Tools (latest)		Not Installed

SDK Quickfix Installation

SDK Component Installer

Completing Requested Actions



Accept License

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Windows 10 SDK
[✓] Android Studio (version 2024.2)
[✓] VS Code (version 1.97.2)
[✓] Connected device (3 available)
[✓] Network resources

! Doctor found issues in 2 categories.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> flutter doctor --android-licenses
Android sdkmanager not found. Update to the latest Android SDK and ensure that the cmdline-tools are installed to resolve this.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> flutter doctor --android-licenses
[=====] 73% Fetch remote repository...
```

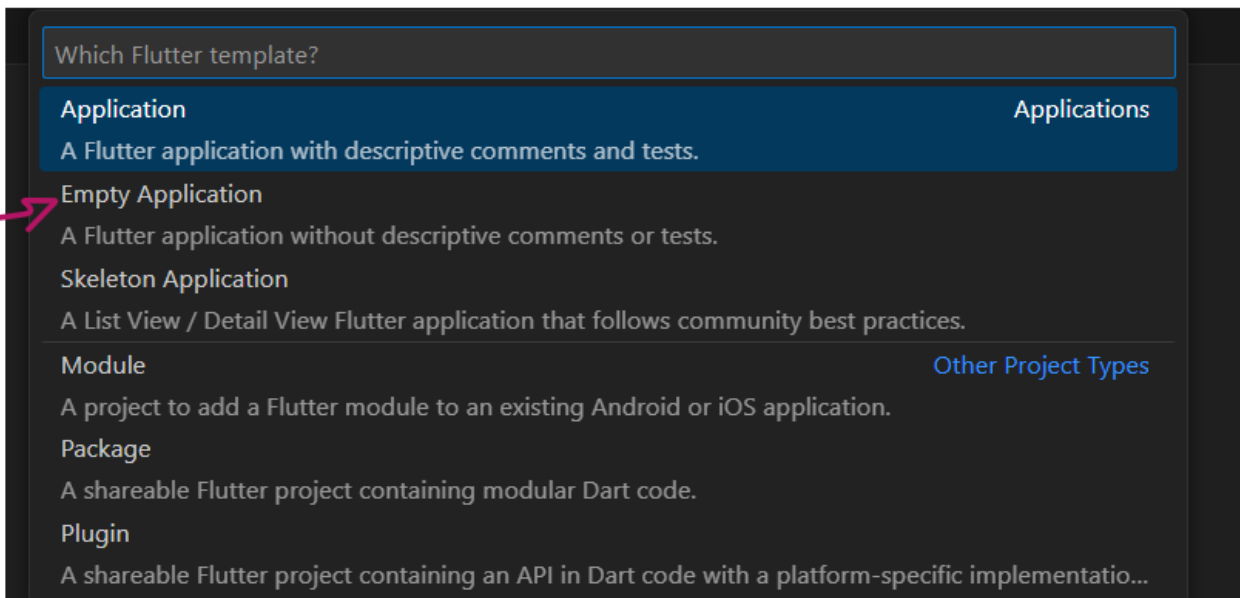


Check any missing requirements (flutter doctor)

```
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.29.0, on Microsoft Windows [Version 10.0.26100.2894], locale en-GB)
[✓] Windows Version (11 Pro 64-bit, 24H2, 2009)
[✓] Android toolchain - develop for Android devices (Android SDK version 35.0.1)
[✓] Chrome - develop for the web
[!] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.12.4)
    X Visual Studio is missing necessary components. Please re-run the Visual Studio installer for the "Desktop
      components:
        MSVC v142 - VS 2019 C++ x64/x86 build tools
        - If there are multiple build tool versions available, install the latest
        C++ CMake tools for Windows
        Windows 10 SDK
[✓] Android Studio (version 2024.2)
[✓] VS Code (version 1.97.2)
[✓] Connected device (3 available)
```



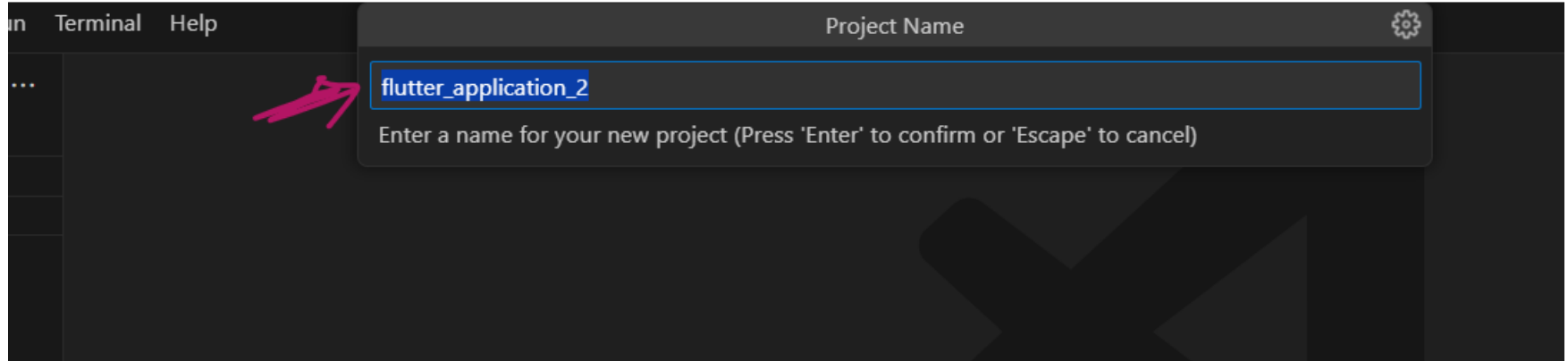

Create New Project





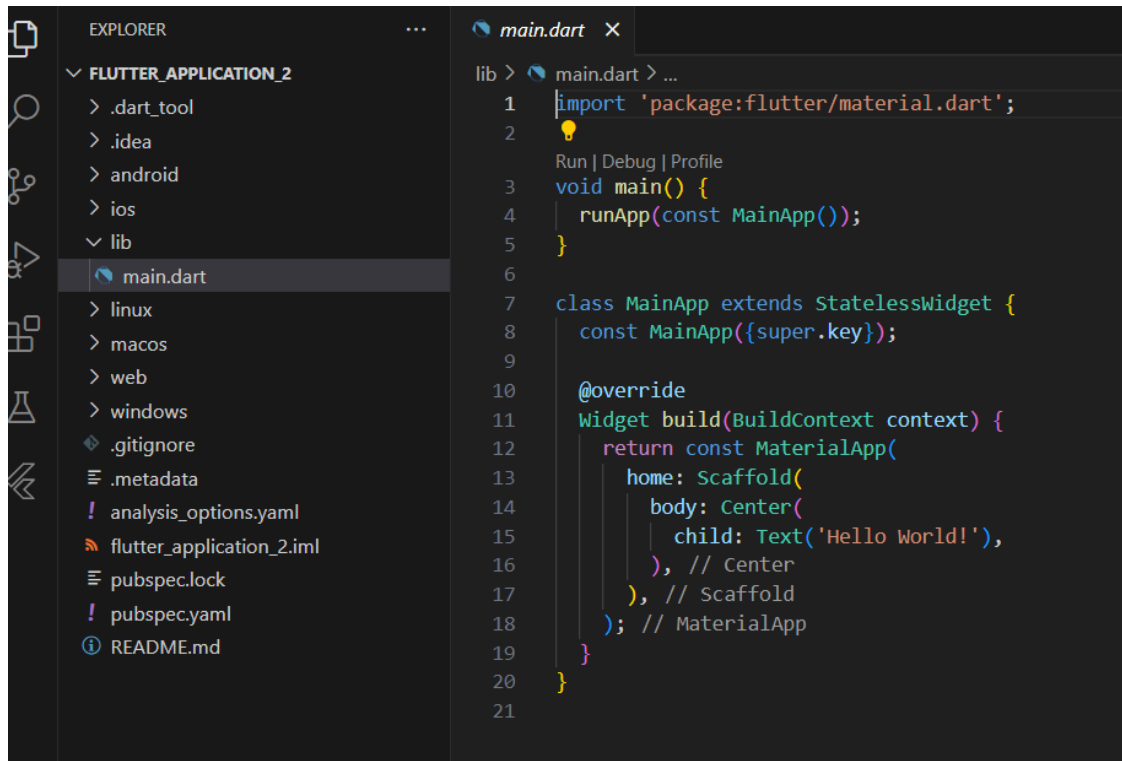
Create New Project

2





Create New Project (Use F5 to run)



The screenshot shows an IDE interface with two main panels. The left panel, titled 'EXPLORER', displays the project structure for 'FLUTTER_APPLICATION_2'. The right panel shows the code for 'main.dart'.

EXPLORER Panel:

- FLUTTER_APPLICATION_2
 - > .dart_tool
 - > .idea
 - > android
 - > ios
 - ▼ lib
 - main.dart (selected)
 - > linux
 - > macos
 - > web
 - > windows
 - ◆ .gitignore
 - ≡ .metadata
 - ! analysis_options.yaml
 - 🔥 flutter_application_2.iml
 - ≡ pubspec.lock
 - ! pubspec.yaml
 - 📖 README.md

main.dart Panel:

```
lib > main.dart > ...
1  import 'package:flutter/material.dart';
2
3  void main() {
4    runApp(const MainApp());
5  }
6
7  class MainApp extends StatelessWidget {
8    const MainApp({super.key});
9
10   @override
11   Widget build(BuildContext context) {
12     return const MaterialApp(
13       home: Scaffold(
14         body: Center(
15           child: Text('Hello World!'),
16         ), // Center
17       ), // Scaffold
18     ); // MaterialApp
19   }
20 }
21
```





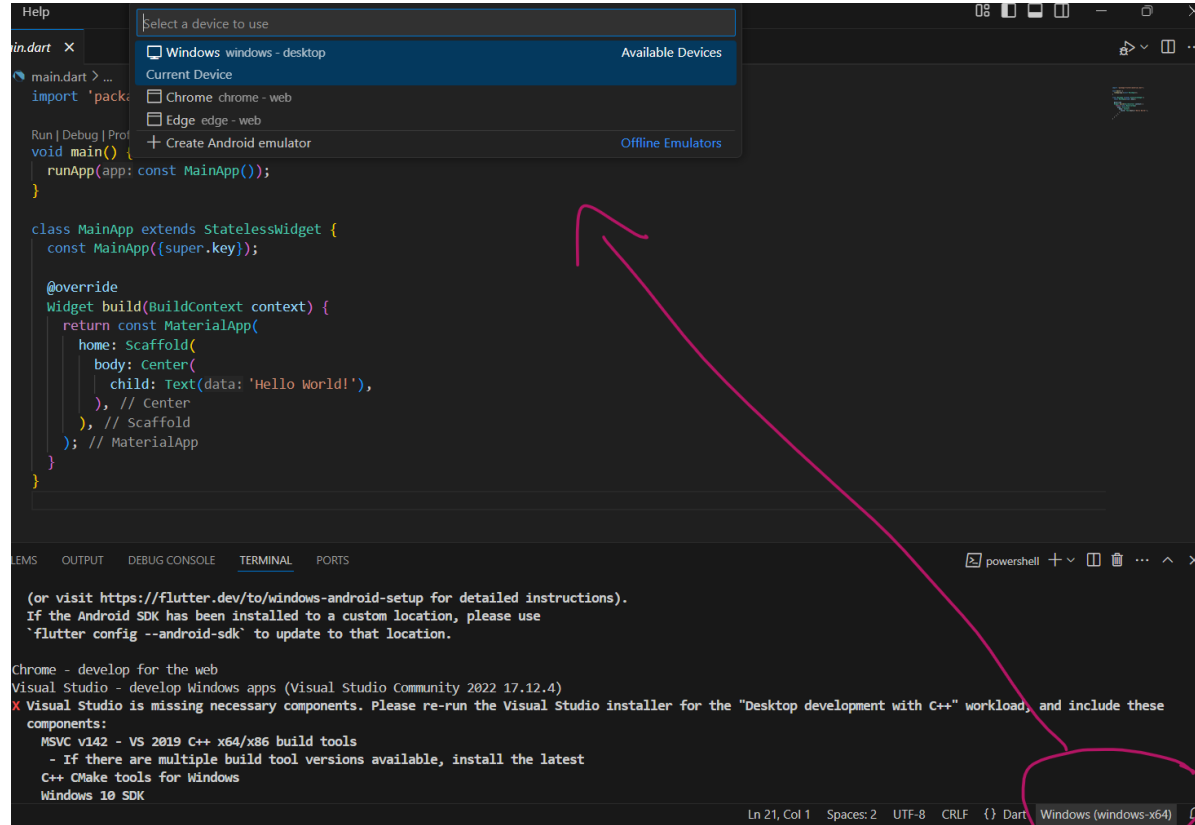
Open Project from window to VS code

Name	Date modified	Type	Size
.dart_tool	26/02/2025 17:11	File folder	
.idea	26/02/2025 17:11	File folder	
android	26/02/2025 17:11	File folder	


```
C:\Windows\System32\cmd.exe X + v  
Microsoft Windows [Version 10.0.26100.2894]  
c) Microsoft Corporation. All rights reserved.  
:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>code .|
```

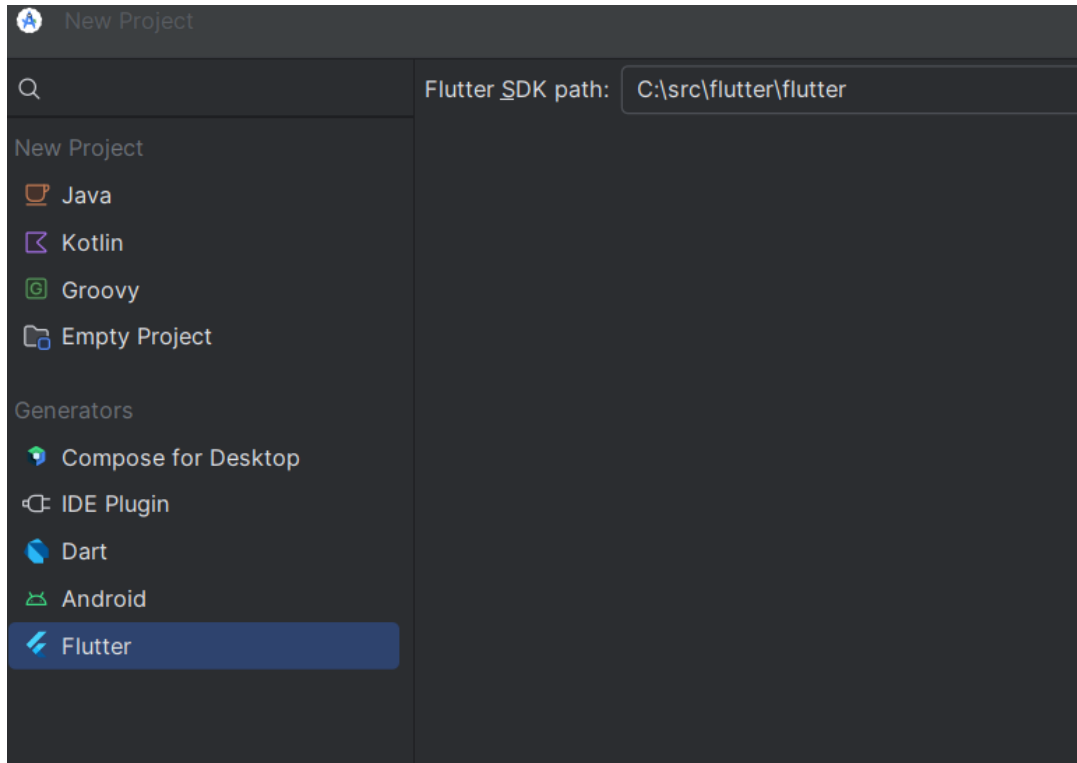


Choose Platform to run the APP





Use Android Studio to add (new devices)





Use Android Studio to add (new devices)

Project name: testFlutter

Project location: ~\StudioProjects\testFlutter

Description: A new Flutter project.

Project type: Application

Organization: com.example

Android language: ☐ Java ☒ Kotlin

Platforms: ☒ Android

When create

Invalid module name: 'testFlutter' - must be a valid Dart package name (lower_case_with_underscores).

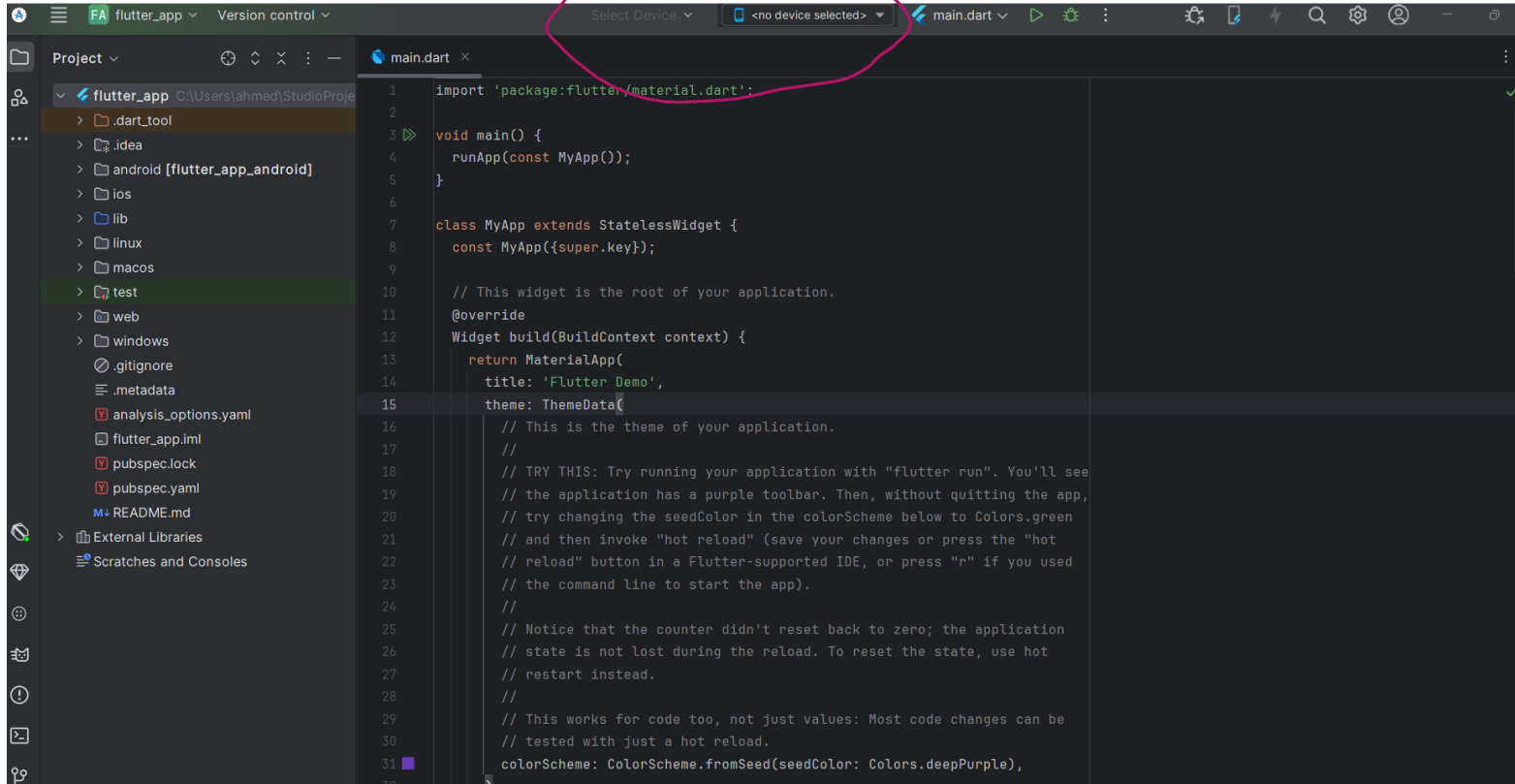
OK

Create project offline





Use Android Studio to add (new devices)





Tools → Device Manager → Add new device

The screenshot shows the Android Studio interface. The 'Virtual Device Configuration' dialog is open, displaying a 'Select Hardware' window. The 'Choose a device definition' section shows a table of device definitions. The 'Pixel Fold' device is selected. The 'Device Manager' tab is also visible, showing the 'Medium Phone API 35' device.

Select Hardware

Choose a device definition

Category	Name	Play Store	Size	Resolution	Density
Phone	Pixel 9 Pro Fold		8.0"	2076x2152	390dpi
Tablet	Pixel 9 Pro		6.3"	1280x2856	xxhdpi
Wear OS	Pixel 9		6.24"	1080x2424	420dpi
Desktop	Pixel 8a		6.1"	1080x2400	420dpi
TV	Pixel 8 Pro		6.7"	1344x2992	xxhdpi
Automotive	Pixel 8		6.17"	1080x2400	420dpi
Legacy	Pixel Fold		7.58"	1840x2208	420dpi

Pixel Fold

Size: xlarge
Ratio: notlong
Density: 420dpi
Folded: 1080x2092

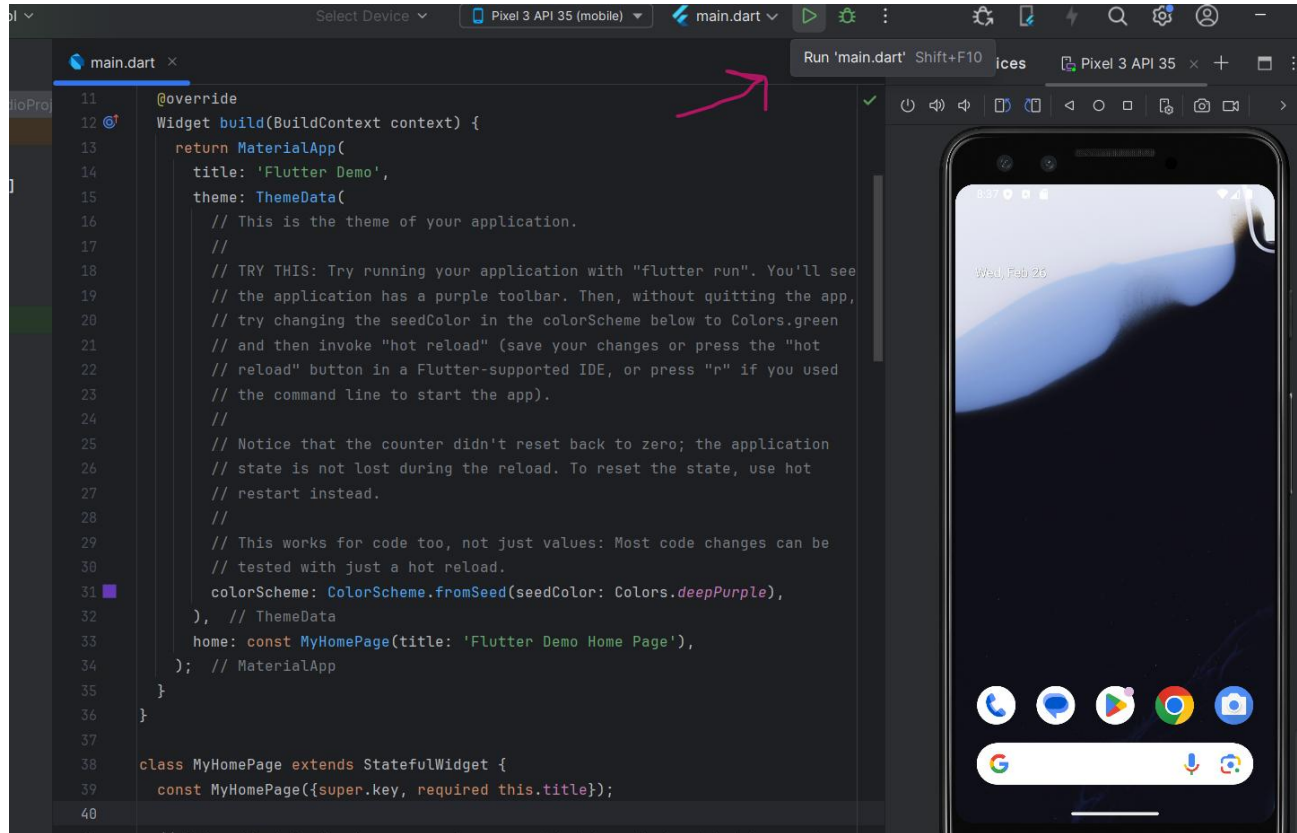
Device Manager

Medium Phone API 35
Android 15.0 ("VanillaIceCream")

Next



Choose the device and run





Material APP: Write your first code

1. Create main function

- Render your app widget.
- Mean: main function take my app

To display it

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

Run | Debug | Profile

```
void main () {
```

```
  runApp(app: app)
```

```
}
```

runApp(widget app)

Inflate the given widget and attach it to the view.

The [runApp] method renders the provided `app` widget into the [PlatformDispatcher.implicitView] by wrapping it in a [View] widget, which will bootstrap the render tree for the app. Apps that want to control which [FlutterView] they render into can use [runWidget] instead.

The widget is given constraints during layout that force it to fill the entire view. If you wish to align your widget to one side of the view (e.g., the top), consider using the [Align] widget. If you wish to center your widget, you can also use the [Center] widget.



Text Widget

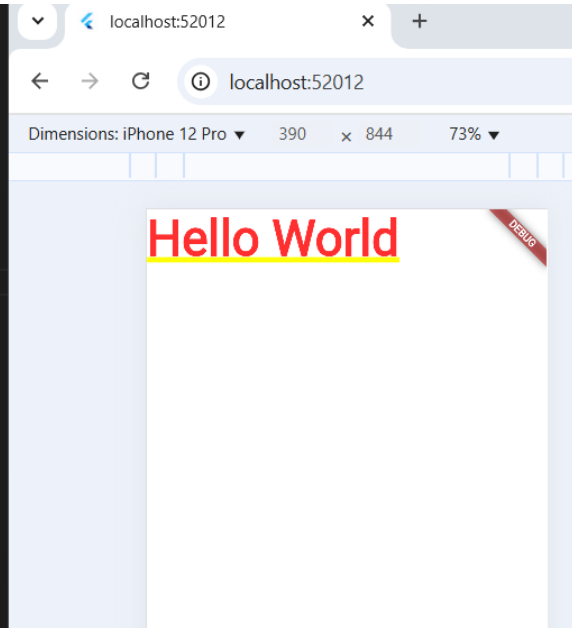
- Inside runApp, we will write the App widget.
- Widget is a Class have a specific UI design
- Main widget for the app is called **MaterialApp**
- **Have all guidelines for other components in UI**

◆ Purpose of `MaterialApp()`

- It **wraps** the entire application with Material Design support.
- Provides **navigation, theming, fonts, localization**, and more.
- Works as the **entry point** for the UI.

```
main.dart > main
import 'package:flutter/material.dart';

Run | Debug | Profile
void main () {
  runApp(
    app: MaterialApp(
      home: Text(data: "Hello World"),
    ) // MaterialApp
  );
}
```



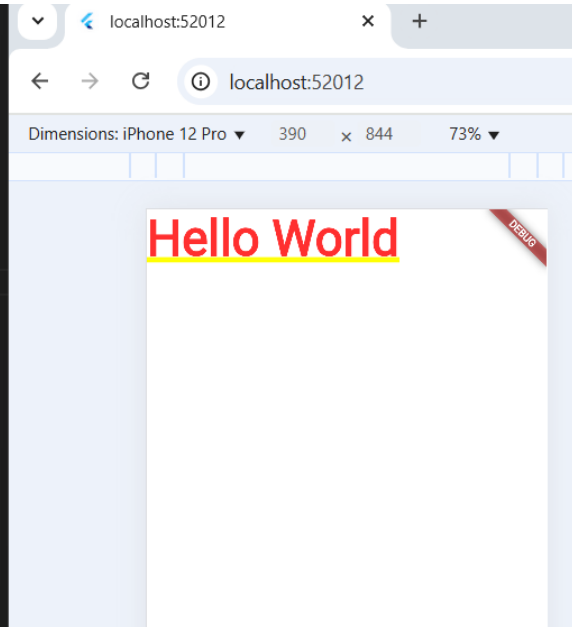


Text Widget

- **MaterialApp**: have **home** property.
- **home**: take another widget.
- For example, it take **Text widget**.
- By default, **Text widget** appear in **top left**
- How to center it ?

```
main.dart > main
import 'package:flutter/material.dart';

void main () {
  runApp(
    app: MaterialApp(
      home: Text(data: "Hello World"),
    ) // MaterialApp
  );
}
```





Center Widget

- How to center it ?
- Using **Center Widget**
- **Center Widget** → center its child in the previous parent
- But, why text is red with underline yellow??

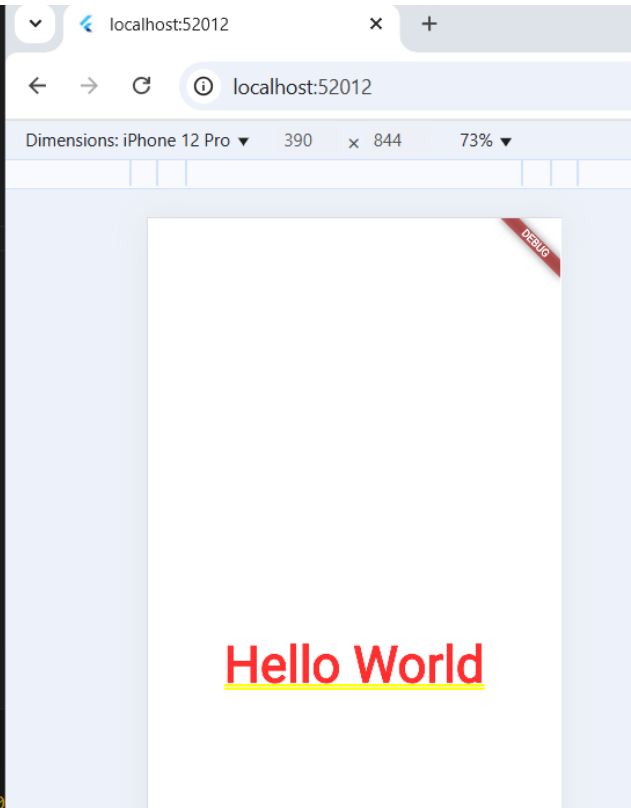
```
main.dart > main
import 'package:flutter/material.dart';

Run | Debug | Profile
void main () {
  runApp(
    app: MaterialApp(
      home: Center(
        child: Text(data: "Hello World"),
      ), // Center
    ) // MaterialApp
  );
}
```

LEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Launching lib\main.dart on Chrome in debug mode...

This app is linked to the debug service: ws://127.0.0.1:5220





Scaffold Widget

- But why text is red with underline yellow??
- Because we need another widget that responsible for implementing material design rules. → called **Scaffold widget**
- What is Scaffold widget??

```
main.dart > main
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(
5     app: MaterialApp(
6       home: Scaffold(
7         body: Center(
8           child: Text(data: "Hello world 2")
9         ) // Center
10      ) // Scaffold
11    ); // MaterialApp
12  }
13
```

Run | Debug | Profile

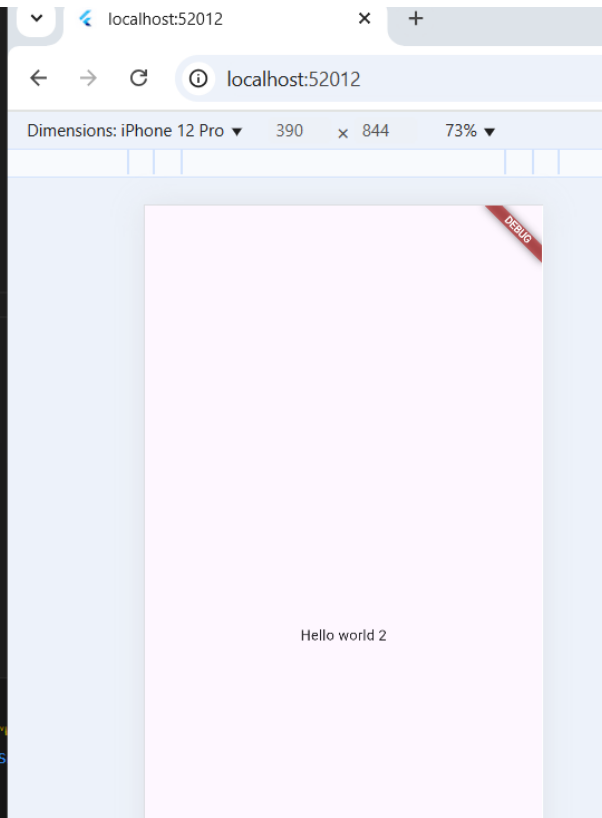
DEBUG CONSOLE

Debug service listening on ws://127.0.0.1:52087/L0CSP

Connecting to VM Service at ws://127.0.0.1:52087/L0CSP

Connected to the VM Service.

Restarted application in 537ms.





How to get other widgets (Widget catalog)

https://docs.flutter.dev/ui/widgets

kpdecker/jsdiff: A ja... Using EPPlus To Imp... sandeepdillerao/cor... Telegram: Contact... CouponXxL - Deals... How to crop and up... HOW TO EXTRACT...

Docs

Homepage Community Packages API reference

Base widgets

Base widgets support a range of common rendering options like input, layout, and text.

Accessibility Make your app accessible.	Animation and motion Bring animations to your app.	Assets, images, and icons Manage assets, display images, and show icons.
Async Widgets supporting async patterns in your Flutter apps.	Basics Widgets to know before building your first Flutter app.	Input Take user input in addition to input widgets in Material components and Cupertino.
Interaction models Respond to touch events and route users to different views.	Layout Arrange other widgets columns, rows, grids, and many other layouts.	Painting and effects These widgets apply visual effects to the children without changing their layout, size, or position.
Scrolling	Styling	Text





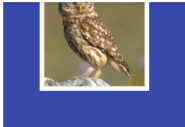
How to get other widgets (Widget catalog)

https://docs.flutter.dev/ui/widgets/basics

kpdecker/jsdiff: A ja... Using EPPlus To Imp... sandeepdillerao/cor... Telegram: Contact... CouponXtL - Deals,... How to crop and up... HOW TO EXTRACT...


Docs

Homepage Community Packages API reference C




Image

A widget that displays an image.




Placeholder

A widget that draws a box that represents where other widgets will one day be added.




Row

Layout a list of child widgets in the horizontal direction.



Scaffold


Implements the basic Material Design visual layout structure. This class provides APIs for showing drawers, snack bars, and bottom sheets.



Text

A run of text with a single style.

g
tions





Scaffold Widget: divide screen to 3 parts (AppBar, body, floating button)

The screenshot shows an IDE with a Dart file named `main.dart`. The code defines a `main` function that runs a `MaterialApp` with a `Scaffold` widget. The `Scaffold` has a `floatingActionButton` that calls `onPressed: () {}`, an `AppBar` with the title `Sample Code`, and a `body` that is a `Center` widget containing a `Text` widget with the text `"Hello world"`.

```
lib > main.dart > main
1 import 'package:flutter/material.dart';
2
3 Run | Debug | Profile
4 void main() {
5   runApp(
6     app: MaterialApp(
7       home: Scaffold(
8         floatingActionButton: FloatingActionButton(onPressed: () {}),
9         appBar: AppBar(title: const Text(data: 'Sample Code')),
10        body: Center(
11          child: Text(data: "Hello world")
12        ) // Center
13      ) // Scaffold
14    ); // MaterialApp
15  }
```

The mobile emulator preview shows the app running on a device. The status bar at the top shows the time 10:18. The app has a light blue background. The `AppBar` has the title `Sample Code`. The `body` is a `Center` widget containing the text `Hello world`. A purple `FloatingActionButton` is at the bottom right.

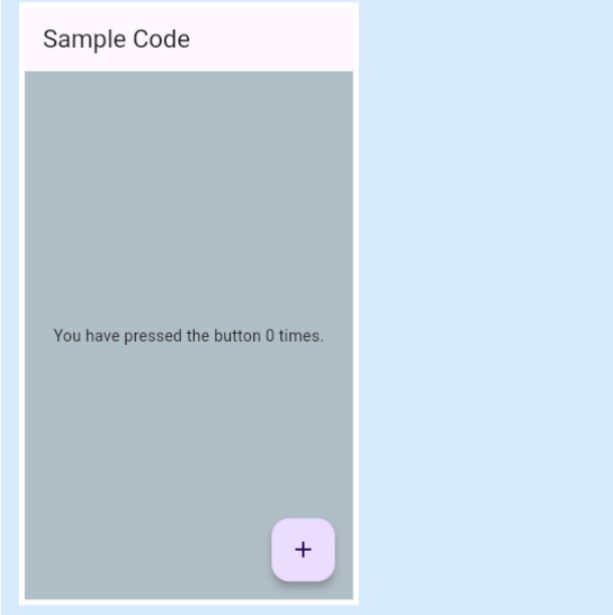
Below the code editor, the `TERMINAL` tab shows the following output:

```
! Doctor found issues in 2 categories.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
* History restored

PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> fl
>>
could not find a command named "restart".

Run 'flutter -h' (or 'flutter <command> -h') for available flutter commands and opt
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
```

This example shows a `Scaffold` with a `blueGrey` `backgroundColor`, `body` center the text within the `Scaffold`. The `FloatingActionButton` is connect





Custom Widget: Organize your code

- Instead of write all widgets inside main function, we make custom widgets
- Custom Widgets can:
 - Stateless or Stateful
- Just extend StatelessWidget, and create your own widget
- Shortcut used: stless

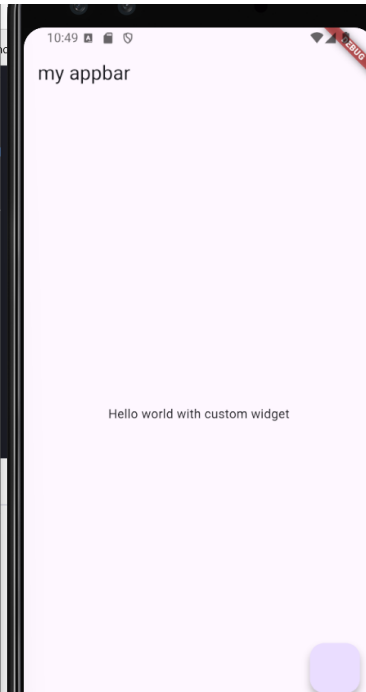
```
lib > main.dart > HelloWorldAPP > build
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(app: HelloWorldAPP());
5 }
6
7 class HelloWorldAPP extends StatelessWidget {
8   HelloWorldAPP();
9
10  @override
11  Widget build(BuildContext context) {
12    return MaterialApp(
13      home: Scaffold(
14        floatingActionButton: FloatingActionButton(onPressed: (){}),
15        appBar: AppBar(title: const Text(data: 'my appbar')),
16        body: Center(
17          child: Text(data: "Hello world with custom widget")
18        ) // Center
19      ) // Scaffold
20    ); // MaterialApp
21  }
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + -v - - - ^ x

```
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
* History restored

PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2> flutter restart
>>
could not find a command named "restart".

Run 'flutter -h' (or 'flutter <command> -h') for available flutter commands and options.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
```





Let's try to make some style on the Text

- use **Container Widget** to wrap text inside it, and add some style on it
- **Why text appear in Container Widget at Top Left?**
 - This is the default
 - How to Solve it?

```
lib > main.dart > HelloWorldAPP > build
3 void main() {
4
5 }
6
7 class HelloWorldAPP extends StatelessWidget {
8   HelloWorldAPP();
9
10  @override
11  Widget build(BuildContext context) {
12    return MaterialApp(
13      home: Scaffold(
14        floatingActionButton: FloatingActionButton(onPressed: (){}),
15        appBar: AppBar(title: const Text('appbar here')),
16        body: Center(
17          child: Container(
18            color: colors.amber,
19            width: 200,
20            height: 200,
21            child: Text("Hello world") // Container
22          ) // Center
23        ) // Scaffold
24      ); // MaterialApp
25    }
```

! Doctor found issues in 2 categories.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
* History restored


appbar here

Hello world



- use **Container Widget** to wrap text inside it, and add some style on it
- **Why text appear in Container Widget at Top Left?**
 - This is the default
 - How to Solve it?
 - Wrap again in center

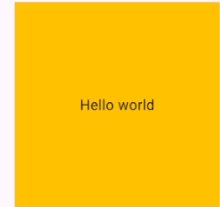
```
class HelloWorldApp extends StatelessWidget {
  HelloWorldApp();

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        floatingActionButton: FloatingActionButton(onPressed: (){}),
        appBar: AppBar(title: const Text('appbar here')),
        body: Center(
          child: Container(
            color:  colors.amber,
            width: 200,
            height: 200,
            child: Center(child: Text("Hello world"))) // Container
          ) // Center
        ) // Scaffold
      ); // MaterialApp
    }
  }
}
```

```
D:\SUI\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
History restored

start

ld not find a command named "restart".
```





Column Widgets: with multiple children

- Y-axis called main axis
- X-axis is called cross axis
- Can you Gess how to use Row Widget?

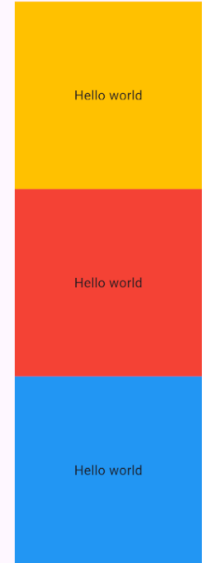
```
lib > main.dart > HelloWorldAPP > build
7 class HelloWorldAPP extends StatelessWidget {
10 @override
11 Widget build(BuildContext context) {
12   return MaterialApp(
13     home: Scaffold(
14       floatingActionButton: FloatingActionButton(onPressed: (){}),
15       appBar: AppBar(title: const Text('appbar here')),
16       body: Column(
17         children: [
18           Center(
19             child: Container(
20               color: Colors.amber,
21               width: 200,
22               height: 200,
23               child: Center(child: Text("Hello world"))) // Container
24           ), // Center
25           Center(
26             child: Container(
27               color: Colors.red,
28               width: 200,
29               height: 200,
30               child: Center(child: Text("Hello world"))) // Container
31           ), // Center
32         ],
33       ),
34     ),
35   );
36 }
```

! Doctor found issues in 2 categories.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>
* History restored

restart
>>
could not find a command named "restart".

Run 'flutter -h' (or 'flutter <command> -h') for available flutter commands and options.
PS D:\SUT\Courses\Flutter\Lectures\Lectures Practice\Lec3\flutter_application_2>

appbar here





Practice

Flutter Topics

✓ Creating flutter project
▶ 7min

✓ Material App
▶ 14min

✓ Scaffolding our app
▶ 15min

✓ Custom widget
▶ 9min

✓ Hot reload
▶ 6min

✓ Container widget
▶ 10min

✓ Column widget
▶ 13min

✓ Row widget
▶ 8min

