**Week 2:**  **Create an application using Flutter to print hello world.**

/\*  Flutter hello world app \*/

import 'package:flutter/material.dart';

void main()

{

runApp(const MyApp());

}

class MyApp extends StatelessWidget

{

const MyApp({Key? key}) : super(key: key);

@override

Widget build(BuildContext context)

{

// Material App

return MaterialApp(

// Scaffold Widget

home: Scaffold(

appBar: AppBar(

// AppBar takes a Text Widget in it's title parameter

title: const Text('Home Page'),

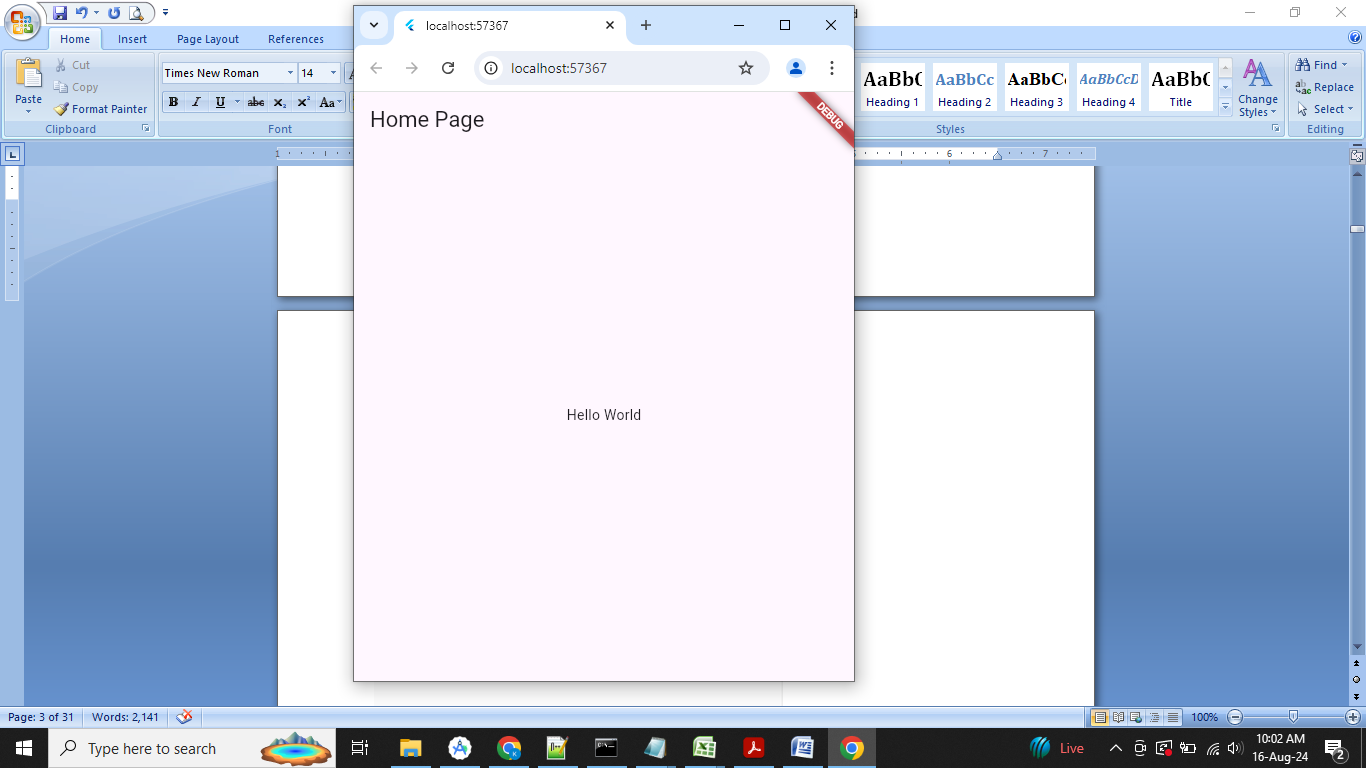
),

body: const Center(child: Text('Hello World')),

));

}

}



**Week 6: Create an application for platform basic widgets (Text, Image, and Icon).**

/\* Flutter Text Widget app \*/

import 'package:flutter/material.dart';

void main()

{

runApp(const MyApp());

}

class MyApp extends StatelessWidget

{

const MyApp({Key? key}) : super(key: key);

@override

Widget build(BuildContext context)

{

// Material App

return MaterialApp(

// Scaffold Widget

home: Scaffold(

backgroundColor: Colors.lightGreen,

appBar: AppBar(

backgroundColor: Colors.green,

// AppBar takes a Text Widget in it's title parameter

title: const Text('Home Page'),

),

body: const Center(// Display a centered text widget

child: Text(

"Welcome to VJIT!!",

// Apply text styling

style: TextStyle(

fontSize: 24, // Set font size

fontWeight: FontWeight.bold, // Set font weight

color: Colors.purple,

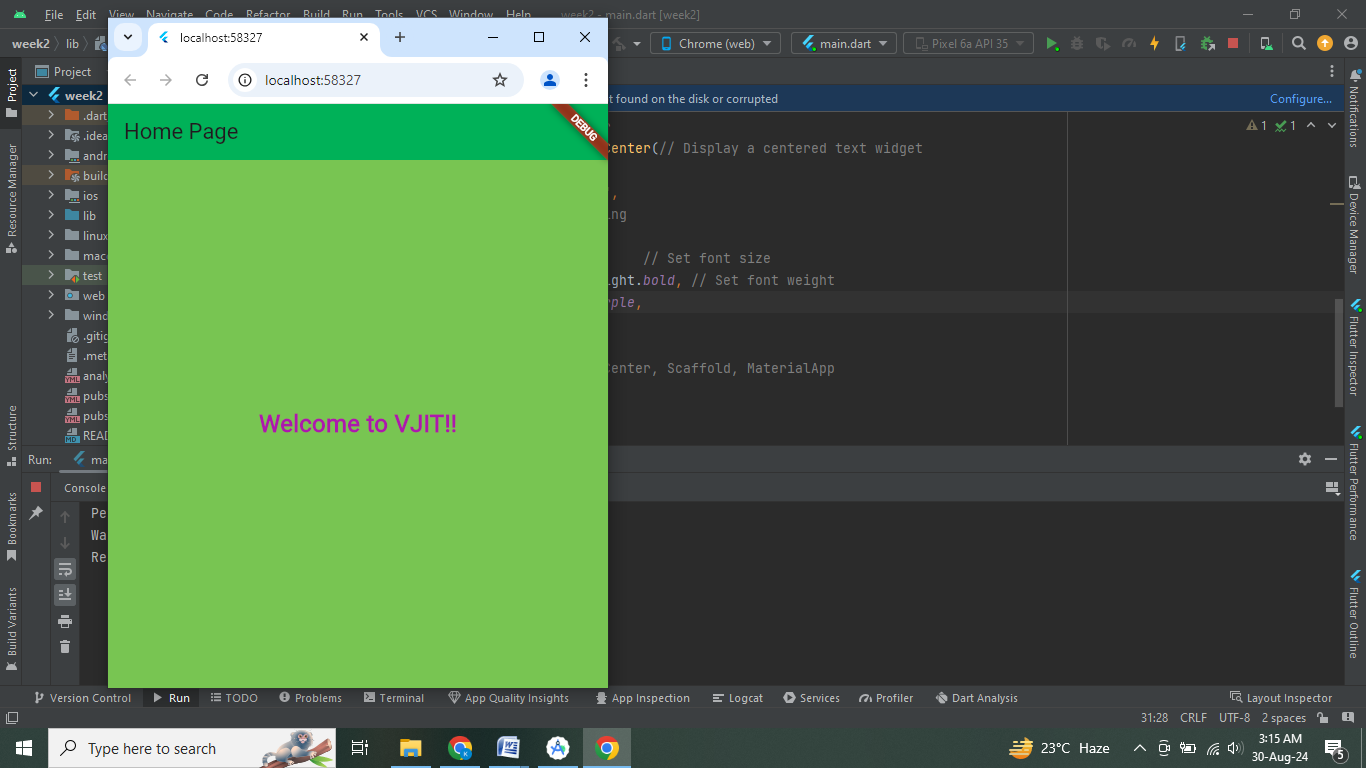
),

),

), ));

}

}



How to display the image in Flutter

To display an image in Flutter, do the following steps:

**Step 1:** First, we need to create a new **folder** inside the root of the Flutter project and named it assets. We can also give it any other name if you want.

**Step 2:** Next, inside this folder, add one image manually.

**Step 3:** Update the **pubspec.yaml** file. Suppose the image name is VJIT\_logo\_2023.png'**,**  then pubspec.yaml file is:

assets:

    - assets/VJIT\_logo\_2023.png'

import 'package:flutter/material.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

home: Scaffold(

appBar: AppBar(

title: Text('Flutter Image Demo'),

),

body: Center(

child: Column(

children: <Widget>[

Image.asset('assets/VJIT\_logo\_2023.png'),

Text(

'Welcome to VJIT::Hyd',

style: TextStyle(fontSize: 20.0),

)

],

),

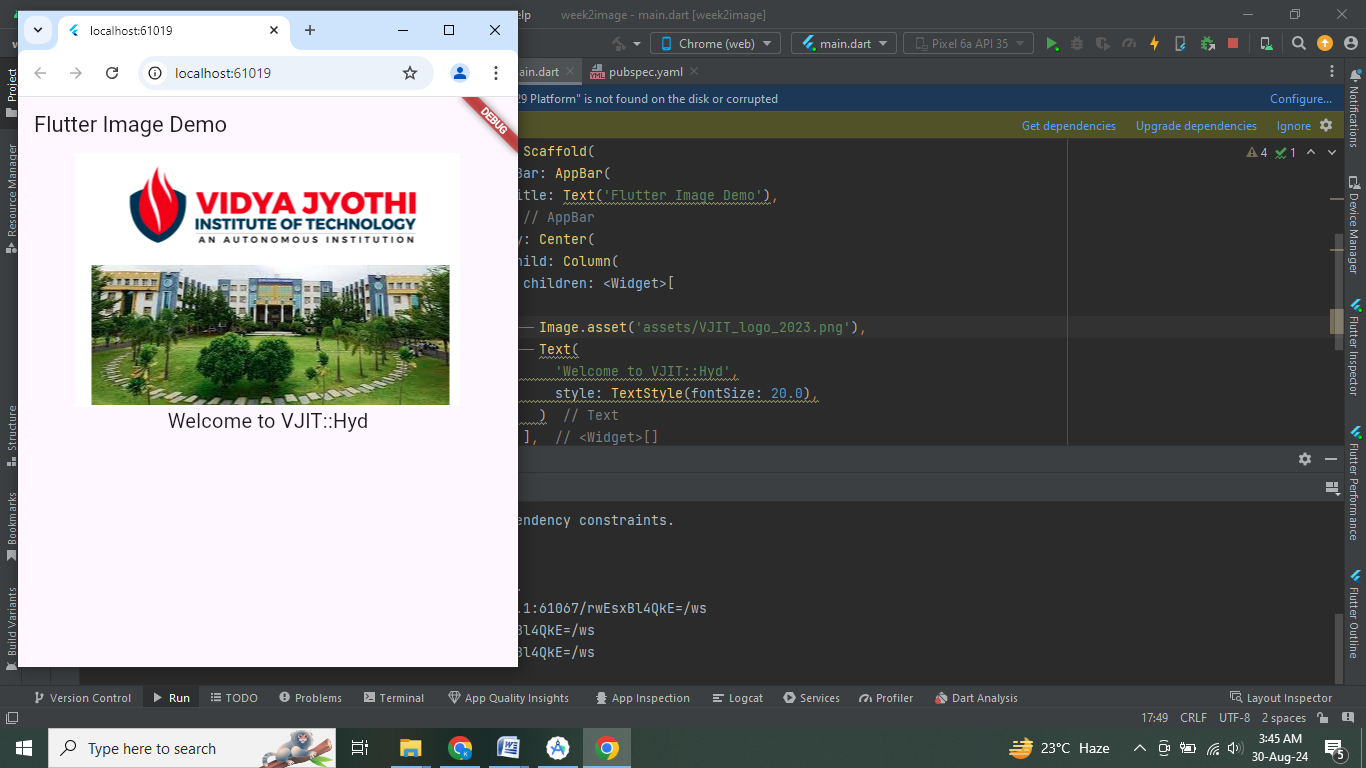
),

),

);

}

}



# Flutter Icons

An icon is a **graphic image** representing an application or any specific entity containing meaning for the user. It can be selectable and non-selectable. **For example**, the company's logo is non-selectable. Sometimes it also contains a **hyperlink** to go to another page. It also acts as a sign in place of a detailed explanation of the actual entity.

[Flutter](https://www.javatpoint.com/flutter) provides an **Icon Widget** to create icons in our applications. We can create icons in Flutter, either using inbuilt icons or with the custom icons. Flutter provides the list of all icons in the **Icons class**. In this article, we are going to learn how to use Flutter icons in the application.

### Icon Widget Properties

Flutter icons widget has different properties for customizing the icons. These properties are explained below:

|  |  |
| --- | --- |
| **Property** | **Descriptions** |
| icon | It is used to specify the icon name to display in the application. Generally, Flutter uses material design icons that are symbols for common actions and items. |
| color | It is used to specify the color of the icon. |
| size | It is used to specify the size of the icon in pixels. Usually, icons have equal height and width. |
| textDirection | It is used to specify to which direction the icon will be rendered. |

import 'package:flutter/material.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

return MaterialApp(

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: MyIconPage(),

);

}

}

class MyIconPage extends StatefulWidget {

@override

\_MyIconPageState createState() => \_MyIconPageState();

}

class \_MyIconPageState extends State<MyIconPage> {

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Flutter Icon App'),

),

body: Row(

mainAxisAlignment: MainAxisAlignment.spaceAround,

children: <Widget>[

Icon(

Icons.camera\_enhance,

size: 70,

color:Colors.green

),

Icon(

Icons.camera\_front,

size: 70,

color:Colors.orange

),

Icon(

Icons.camera\_rear,

size: 70,

color:Colors.black

),

]),

);

}

}

