

## **- Including Icons in Flutter**

Flutter provides built-in icons via the Icons class and allows the use of custom icons through the pubspec.yaml file.

### **1.1 Using Built-in Icons**

Flutter provides an extensive set of material icons that can be used as follows:

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

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Icons',
      theme: ThemeData(
        primaryColor: Colors.red,
      ),
      home: Scaffold(
        appBar: AppBar(title: Text("Flutter Icons")),
        body: Center(
          child: Icon(
            Icons.favorite,
            color: Colors.red,
            size: 50.0,
          ),
        ),
      ),
    );
  }
}
```

### **1.2 Using Custom Icons**

To use custom icons, first, download or generate an icon font using [FlutterIcon](#) and add it to the pubspec.yaml file:

```
flutter:
  fonts:
    -
      family: CustomIcons
      fonts:
        - asset: assets/fonts/custom_icons.ttf
```

Use it in your app:

```
Icon(IconData(0xe900, fontFamily: 'CustomIcons'))
```

## - Including Images in Flutter

Flutter supports various ways to include images, such as from the assets folder, network URLs, and memory.

### 2.1 Adding Image Assets

1. Place images inside the assets/images/ directory.
2. Declare them in pubspec.yaml:

```
flutter:
```

```
assets:
```

```
- assets/images/sample.png
```

3. Use them in your app:

```
Image.asset('assets/images/sample.png', width: 200, height: 200)
```

### 2.2 Using Network Images

Load images directly from the internet:

```
Image.network('https://example.com/sample.jpg')
```

## - Including Custom Fonts in Flutter

Custom fonts can enhance the UI by providing unique typography.

### 3.1 Adding Custom Fonts

1. Place font files in assets/fonts/.
2. Declare them in pubspec.yaml:

```
flutter:
```

```
fonts:
```

```
- family: CustomFont
```

```
  fonts:
```

```
- asset: assets/fonts/CustomFont-Regular.ttf
```

## 3.2 Using Custom Fonts in the App

Apply the font to text widgets:

```
Text(  
  'Hello, Flutter!',  
  style: TextStyle(fontFamily: 'CustomFont', fontSize: 24),  
)
```

# Output

