

## **EXPERIMENT NO: - 03**

**Name:-** Arnav Sawant

**Class:-** D15A

**Roll:No: -** 52

**AIM: -** To include icons, images, fonts in Flutter app.

---

### **Theory: -**

Flutter is a versatile open-source UI framework , which allows developers to build natively compiled applications for mobile, web, and desktop platforms from a single codebase. One of the key strengths of Flutter is its flexibility in creating highly customizable UIs. This practical focuses on incorporating essential visual elements—icons, images, and custom fonts—into a Flutter application. These elements enhance the visual appeal and usability of the app, providing an engaging experience for users.

A flutter app when built has both assets (resources) and code. Assets are available and deployed during runtime. The asset is a file that can include static data, configuration files, icons, and images. The Flutter app supports many image formats, such as JPEG, WebP, PNG, GIF, animated WebP/GIF, BMP, and WBMP.

Visual elements play a significant role in app development.

- **Enhanced User Experience:** Images and icons make your app visually appealing and user-friendly.
- **Information Conveyance:** They convey information quickly and intuitively. A well-chosen icon can replace lengthy text.
- **Branding:** Custom icons and images reinforce your app's branding, making it memorable.

### ➤ **Adding Icons in Flutter**

Flutter provides built-in material design icons through the Icons class. Custom icons can also be added using third-party packages such as flutter\_launcher\_icons and font\_awesome\_flutter.

```
Icon(  
  Icons.home,  
  size: 40,  
);
```

## ➤ Adding Images in Flutter

Flutter supports images from three sources:

### 1. **Assets** (Stored locally in the project)

- Place the image inside the assets/images folder in the project.
- Declare the image in pubspec.yaml

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

- Display the image in the app

```
Image.asset('assets/images/sample.png');
```

### 2. **Network** (Fetched from the internet)

Displaying images from the internet or network is very simple. Flutter provides a built-in method `Image.network` to work with images from a URL. The `Image.network` method also allows you to use some optional properties, such as height, width, color, fit, and many more.

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

### 3. **Memory or File** (Stored on the device)

## ➤ Adding Custom Fonts in Flutter

By default, Flutter uses the Roboto font, but custom fonts can be added for a unique UI.

- Download the font and place it in the assets/fonts/ folder.
  - Declare the font in pubspec.yaml
  - Use the font in the app
- ```
Text(  
  'Custom Font Example',  
  style: TextStyle(fontFamily: 'CustomFont', fontSize: 24),  
);
```

Code: -

### **profile.dart**

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

class ProfileScreen extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Colors.white,
        elevation: 0,
        leading: IconButton(
          icon: Icon(Icons.arrow_back, color:
Colors.black),
          onPressed: () =>
Navigator.pop(context),
        ),
      actions: [
        TextButton(
          onPressed: () {},
          child: Text("Feedback", style:
TextStyle(color: Colors.black)),
        ),
        IconButton(
          icon: Icon(Icons.settings, color:
Colors.black),
          onPressed: () {},
        ),
      ],
    ),
    body: Column(
      crossAxisAlignment:
CrossAxisAlignment.center,
      children: [
        SizedBox(height: 10),

        // Profile Picture
        Center(
          child: Column(
            children: [
              CircleAvatar(
                radius: 40,
                backgroundImage:
AssetImage('assets/logo.jpg'), // Replace with
your image
              ),
              SizedBox(height: 10),
              Text("Arnav", style:
TextStyle(fontSize: 18, fontWeight:
FontWeight.bold)),
              SizedBox(height: 5),

              // Edit Profile Button
              ElevatedButton.icon(
                onPressed: () {
                  Navigator.push(
                    context,
                    MaterialPageRoute(builder:
(context) => EditProfileScreen()),
                  );
                },
                style: ElevatedButton.styleFrom(
                  backgroundColor:
Colors.grey[200],
                  foregroundColor: Colors.black,
                  padding:
EdgeInsets.symmetric(horizontal: 12, vertical:
5),
                  shape: RoundedRectangleBorder(
                    borderRadius:
BorderRadius.circular(20),
                  ),
                ),
                icon: Icon(Icons.edit, size: 16),
                label: Text("Edit Profile"),
              ),
            ],
          ),
        ),
        SizedBox(height: 20),

        // My Bookmarks Section
        Padding(
          padding:
```

```

EdgeInsets.symmetric(horizontal: 20),
  child: Column(
    crossAxisAlignment:
CrossAxisAlignment.start,
    children: [
      Text("My Bookmarks",
        style: TextStyle(fontSize: 16,
fontWeight: FontWeight.bold)),
      Container(
        width: 40,
        height: 2,
        color: Colors.blue,
        margin: EdgeInsets.only(top: 3,
bottom: 10),
      ),

      SizedBox(height: 20),

      // Empty Bookmarks Message
      Center(
        child: Column(
          children: [
            Text("It's Empty Here",
              style: TextStyle(fontSize: 16,
fontWeight: FontWeight.bold)),

```

```

SizedBox(height: 5),
Text(
  "Click on the bookmark icon to
bookmark a short",
  textAlign: TextAlign.center,
  style: TextStyle(color:
Colors.grey, fontFamily: 'GrechenFuemen',
fontSize: 25),
),
),
),
Image.network('https://www.shutter
stock.com/image-vector/vector-flat-
illustration-no-bookmarks-260nw-
2298489655.jpg', width: 250,height: 250,fit:
BoxFit.fitWidth,),
],
),
),
),
);
}
}

```

## settings.dart

```

import 'package:flutter/material.dart';

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: SettingsPage(),
    );
  }
}

```

```

class SettingsPage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Settings'),
        centerTitle: true,
      ),
      body: ListView(
        padding: EdgeInsets.all(16),
        children: [
          // Language
          _buildSettingsItem(
            icon: Icons.language,

```

```

    title: 'Language',
    subtitle: 'English',
    onTap: () {
      // Handle language selection
    },
  ),
  Divider(),

  // Notifications
  _buildSettingsItem(
    icon: Icons.notifications,
    title: 'Notifications',
    onTap: () {
      // Handle notifications settings
    },
  ),
  Divider(),

  // Your Preferences
  _buildSettingsItem(
    icon: Icons.settings,
    title: 'Your Preferences',
    onTap: () {
      // Handle preferences
    },
  ),
  Divider(),

  // HD Images
  _buildSettingsItem(
    icon: Icons.hd,
    title: 'HD Images',
    onTap: () {
      // Handle HD images settings
    },
  ),
  Divider(),

  // Night Mode
  _buildSwitchItem(
    icon: Icons.nightlight_round,
    title: 'Night Mode',
    subtitle: 'For better readability at
night',
    value: false, // Default value
    onChanged: (value) {
      // Handle night mode toggle

```

```

  },
),
Divider(),

// Autoplay
_buildSwitchItem(
  icon: Icons.play_circle_filled,
  title: 'Autoplay',
  subtitle: 'Off',
  value: false, // Default value
  onChanged: (value) {
    // Handle autoplay toggle
  },
),
Divider(),

// Text Size
_buildSettingsItem(
  icon: Icons.text_fields,
  title: 'Text Size',
  subtitle: 'Default',
  onTap: () {
    // Handle text size settings
  },
),
Divider(),

// Share this app
_buildSettingsItem(
  icon: Icons.share,
  title: 'Share this app',
  onTap: () {
    // Handle share functionality
  },
),
],
),
);
}

// Helper method to build a settings item with
an icon and text
Widget _buildSettingsItem({
  required IconData icon,
  required String title,
  String? subtitle,
  VoidCallback? onTap,

```

```
  }) {  
    return ListTile(  
      leading: Icon(icon, color: Colors.blue),  
      title: Text(title, style: TextStyle(fontSize:  
16)),  
      subtitle: subtitle != null ? Text(subtitle) :  
null,  
      onTap: onTap,  
    );  
  }
```

*// Helper method to build a settings item with  
a switch*

```
Widget _buildSwitchItem({  
  required IconData icon,  
  required String title,  
  String? subtitle,  
  required bool value,  
  required ValueChanged<bool> onChanged,  
}) {  
  return ListTile(  
    leading: Icon(icon, color: Colors.blue),  
    title: Text(title, style: TextStyle(fontSize:  
16)),  
    subtitle: subtitle != null ? Text(subtitle) :  
null,  
    trailing: Switch(  
      value: value,  
      onChanged: onChanged,  
    ),  
  );  
}
```

