

EXPERIMENT NO: - 03

Name:- Kunal Punjabi

Class:- D15A

Roll:No: - 43

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: - product_details.dart

```

import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import
'package:url_launcher/url_launcher.dart
';
import          'cart_provider.dart';

class ProductDetailsPage extends
StatefulWidget {
  final      String      imagePath;
  final      String      productName;
  final      double      price;
  final      List<String> colors;

  const      ProductDetailsPage({
    Key?      key,
    required  this.imagePath,
    required  this.productName,
    required  this.price,
    required  this.colors,
  })      :      super(key:      key);

  @override
  _ProductDetailsPageState
createState()      ==>
  _ProductDetailsPageState();
}

class _ProductDetailsPageState extends
State<ProductDetailsPage> {
  late      String      selectedColor;
  int      selectedSize      =      28;
  final List<int> availableSizes = [28,
30,      32,      34,      36,      38];

  @override
  void      initState()      {
    super.initState();

```

```

    selectedColor = widget.colors.first;
  }

  void      _addToCart()      {
    final      cartProvider      =
Provider.of<CartProvider>(context,
listen:      false);

    cartProvider.addToCart({
      'imagePath':      widget.imagePath,
      'productName':
widget.productName,
      'price':      widget.price,
      'color':      selectedColor,
      'size':      selectedSize,
      'quantity':      1,
    });

    ScaffoldMessenger.of(context).showSn
ackBar(
      SnackBar(
        content:
Text("${widget.productName} added to
cart!"),
        duration: const Duration(seconds:
2),
      ),
    );
  }

  Future<void> _orderNow() async {
    String      message      =
      "Hello, I want to order:\n-
*${widget.productName}* \n- Price:
₹${widget.price} \n- Color:
$selectedColor \n- Size: $selectedSize";
    Uri      url      =

```

```
Uri.parse("https://wa.me/+919876543210?text=${Uri.encodeComponent(message)}");
```

```
    if (await canLaunchUrl(url)) {
      await launchUrl(url, mode:
LaunchMode.externalApplication);
    } else {
      if (mounted) {
```

```
ScaffoldMessenger.of(context).showSnackBar(
  const SnackBar(content:
Text("Could not open WhatsApp")),
);
}
}
}
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title:
Text(widget.productName)),
    body: SingleChildScrollView(
      child: Padding(
        padding: const
EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment:
CrossAxisAlignment.start,
          children: [
            // Product Image
            ClipRRect(
              borderRadius:
BorderRadius.circular(10),
              child: Image.asset(
                widget.imagePath,
```

```
width: double.infinity,
height: 250,
fit: BoxFit.cover,
),
),
const SizedBox(height: 16),
// Product Name
Text(
  widget.productName,
  style: const
TextStyle(fontSize: 22, fontWeight:
FontWeight.bold),
),
const SizedBox(height: 8),
// Product Price
Text(
```

```
"₹${widget.price.toStringAsFixed(2)}"
,
style: const
TextStyle(fontSize: 18, color:
Colors.green, fontWeight:
FontWeight.bold),
),
const SizedBox(height: 16),
// Select Color
const Text("Select Color:",
style: TextStyle(fontSize: 16,
fontWeight: FontWeight.bold)),
Wrap(
  spacing: 8,
  children:
widget.colors.map((color) {
    return ChoiceChip(
      label: Text(color),
```

```

        selected: selectedColor ==
color,
        selectedColor:
Colors.blueAccent,
        onSelect: (selected) =>
setState(() => selectedColor = color),
        labelStyle: TextStyle(color:
selectedColor == color ? Colors.white :
Colors.black),
    );
    }).toList(),
),
const SizedBox(height: 16),

//      Select      Size
const Text("Select Size:", style:
TextStyle(fontSize: 16, fontWeight:
FontWeight.bold)),
Wrap(
    spacing: 8,
    children:
availableSizes.map((size) {
    return ChoiceChip(
        label: Text(size.toString()),
        selected: selectedSize ==
size,
        selectedColor:
Colors.blueAccent,
        onSelect: (selected) =>
setState(() => selectedSize = size),
        labelStyle: TextStyle(color:
selectedSize == size ? Colors.white :
Colors.black),
    );
    }).toList(),
),
const SizedBox(height: 24),

```

```

//                      Buttons
Row(
    mainAxisAlignment:
MainAxisAlignment.spaceEvenly,
    children: [
        ElevatedButton.icon(
            onPressed: _addToCart,
            icon: const
Icon(Icons.shopping_cart),
            label: const Text("Add to
Cart"),
        ),
        ElevatedButton.icon(
            onPressed: _orderNow,
            icon: const
Icon(Icons.shopping_bag),
            label: const Text("Order
Now"),
            style:
ElevatedButton.styleFrom(background
Color: Colors.green),
        ),
    ],
),
),
),
),
);
}
}

```

Screenshots :

