

Experiment No. 2

AIM: To design Flutter UI by including common widgets.

THEORY:

1. Introduction to Flutter UI

Flutter is a UI toolkit by Google that enables the development of cross-platform applications. It follows a widget-based architecture, where everything in the UI is a widget. There are two main types of widgets:

- Stateless Widgets – Immutable widgets that do not change after creation.
- Stateful Widgets – Widgets that can change dynamically based on user interaction or data updates.

2. Commonly Used Widgets in Flutter

2.1. Layout Widgets

These widgets structure the UI by defining how elements are positioned:

- Container – A box-like widget used for layout customization.
- Row & Column – Used to arrange widgets horizontally and vertically.
- Stack – Places widgets on top of each other for layered designs.
- Expanded & Flexible – Adjusts child widget sizes dynamically within Row or Column.

2.2. Interactive Widgets

Widgets that respond to user input:

- ElevatedButton, TextButton, OutlinedButton – Different button styles for actions.
- GestureDetector – Detects taps, swipes, and other touch interactions.

2.3. Input Widgets

Used for collecting user data:

- TextField – For entering text input.
- Checkbox, Switch, Radio Button – For selections and toggles.

2.4. Display & Styling Widgets

Enhance UI appearance:

- Text – Displays text with various styles.
- Image – Loads and displays images from different sources.
- Card – A material-style container with elevation.
- ListView & GridView – Displays scrollable lists and grids of content.

2.5. Navigation & Routing Widgets

Widgets for navigating between screens:

- Navigator & Routes – Manages screen transitions and navigation.

By combining these widgets, Flutter enables the creation of responsive and visually appealing UIs.

CODE & OUTPUT:**Home Screen:**

```

import 'package:card_swiper/card_swiper.dart';
import 'package:flutter/material.dart';
import 'package:flutter_icons/flutter_icons.dart';
import 'package:grocery_app/inner_screens/feeds_screen.dart';
import 'package:grocery_app/inner_screens/on_sale_screen.dart';
import 'package:grocery_app/provider/dark_theme_provider.dart';
import 'package:grocery_app/services/utls.dart';
import 'package:grocery_app/widgets/text_widget.dart';
import 'package:provider/provider.dart';

import '../consts/contss.dart';
import '../models/products_model.dart';
import '../providers/products_provider.dart';
import '../services/global_methods.dart';
import '../widgets/feed_items.dart';
import '../widgets/on_sale_widget.dart';

class HomeScreen extends StatefulWidget {
  const HomeScreen({Key? key}) : super(key: key);

  @override
  State<HomeScreen> createState() => _HomeScreenState();
}

class _HomeScreenState extends State<HomeScreen> {
  @override
  Widget build(BuildContext context) {
    final Utlis utls = Utlis(context);
    final themeState = utls.getTheme;
    final Color color = Utlis(context).color;
    Size size = utls.getScreenSize;
    final productProviders = Provider.of<ProductsProvider>(context);
    List<ProductModel> allProducts = productProviders.getProducts;
    List<ProductModel> productsOnSale = productProviders.getOnSaleProducts;
    return Scaffold(
      body: SingleChildScrollView(
        child: Column(
          children: [
            SizedBox(
              height: size.height * 0.33,
              child: Swiper(
                itemBuilder: (BuildContext context, int index) {
                  return Image.asset(
                    Constss.offerImages[index],
                    fit: BoxFit.fill,
                  );
                },
                autoplay: true,
                itemCount: Constss.offerImages.length,
                pagination: const SwiperPagination(
                  alignment: Alignment.bottomCenter,
                  builder: DotSwiperPaginationBuilder(
                    color: Colors.white, activeColor: Colors.red)),
                // control: const SwiperControl(color: Colors.black),

```

```

    ),
  ),
  const SizedBox(
    height: 6,
  ),
  TextButton(
    onPressed: () {
      GlobalMethods.navigateTo(
        ctx: context, routeName: OnSaleScreen.routeName);
    },
    child: TextWidget(
      text: 'View all',
      maxLines: 1,
      color: Colors.blue,
      textSize: 20,
    ),
  ),
  const SizedBox(
    height: 6,
  ),
  Row(
    children: [
      RotatedBox(
        quarterTurns: -1,
        child: Row(
          children: [
            TextWidget(
              text: 'On sale'.toUpperCase(),
              color: Colors.red,
              textSize: 22,
              isTitle: true,
            ),
            const SizedBox(
              width: 5,
            ),
            const Icon(
              IconLight.discount,
              color: Colors.red,
            ),
          ],
        ),
      ),
    ],
  ),
  const SizedBox(
    width: 8,
  ),
  Flexible(
    child: SizedBox(
      height: size.height * 0.24,
      child: ListView.builder(
        itemCount: productsOnSale.length < 10
          ? productsOnSale.length
          : 10,
        scrollDirection: Axis.horizontal,
        itemBuilder: (ctx, index) {
          return ChangeNotifierProvider.value(
            value: productsOnSale[index],
            child: const OnSaleWidget());
        },
      ),
    ),
  ),
],

```

```

    ),
    const SizedBox(
      height: 10,
    ),
    Padding(
      padding: const EdgeInsets.all(8.0),
      child: Row(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          TextWidget(
            text: 'Our products',
            color: color,
            textSize: 22,
            isTitle: true,
          ),
          // const Spacer(),
          TextButton(
            onPressed: () {
              GlobalMethods.navigateTo(
                ctx: context, routeName: FeedsScreen.routeName);
            },
            child: TextWidget(
              text: 'Browse all',
              maxLines: 1,
              color: Colors.blue,
              textSize: 20,
            ),
          ),
        ],
      ),
    ),
    GridView.count(
      shrinkWrap: true,
      physics: const NeverScrollableScrollPhysics(),
      crossAxisCount: 2,
      padding: EdgeInsets.zero,
      // crossAxisSpacing: 10,
      childAspectRatio: size.width / (size.height * 0.59),
      children: List.generate(
        allProducts.length < 4
          ? allProducts.length // length 3
          : 4, (index) {
        return ChangeNotifierProvider.value(
          value: allProducts[index],
          child: const FeedsWidget(),
        );
      }
    ),
  ),
),
),

```

Feeds:

```
import 'package:flutter/material.dart';
import 'package:flutter_icons/flutter_icons.dart';
import 'package:grocery_app/consts/contss.dart';
import 'package:grocery_app/models/products_model.dart';
import 'package:grocery_app/providers/products_provider.dart';
import 'package:provider/provider.dart';

import '../services/utils.dart';
import '../widgets/back_widget.dart';
import '../widgets/feed_items.dart';
import '../widgets/text_widget.dart';

class FeedsScreen extends StatefulWidget {
  static const routeName = "/FeedsScreenState";
  const FeedsScreen({Key? key}) : super(key: key);

  @override
  State<FeedsScreen> createState() => _FeedsScreenState();
}

class _FeedsScreenState extends State<FeedsScreen> {
  final TextEditingController? _searchTextController = TextEditingController();
  final FocusNode _searchTextFocusNode = FocusNode();
  @override
  void dispose() {
    _searchTextController!.dispose();
    _searchTextFocusNode.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
```

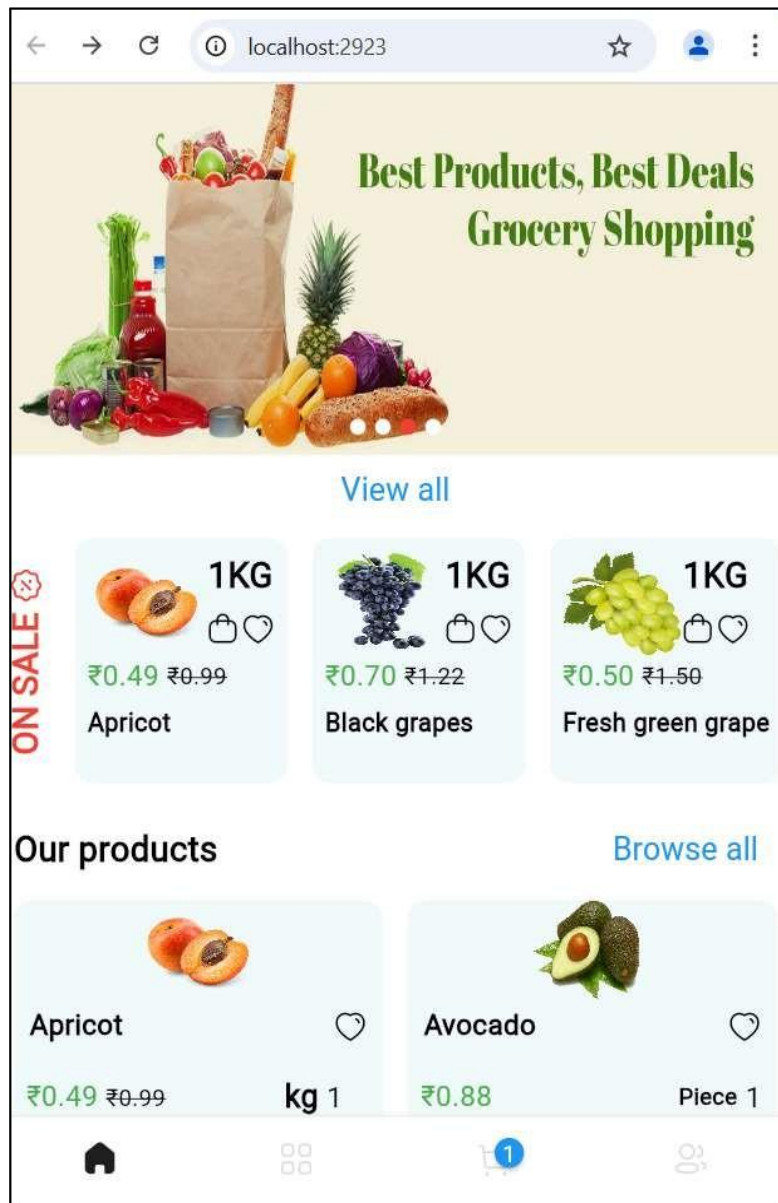
```

final Color color = Utils(context).color;
Size size = Utils(context).getScreenSize;
final productProviders = Provider.of<ProductsProvider>(context);
List<ProductModel> allProducts = productProviders.getProducts;
return Scaffold(
  appBar: AppBar(
    leading: const BackWidget(),
    elevation: 0,
    backgroundColor: Theme.of(context).scaffoldBackgroundColor,
    centerTitle: true,
    title: TextWidget(
      text: 'All Products',
      color: color,
      textSize: 20.0,
      isTitle: true,
    ),
  ),
  body: SingleChildScrollView(
    child: Column(children: [
      Padding(
        padding: const EdgeInsets.all(8.0),
        child: SizedBox(
          height: kBottomNavigationBarHeight,
          child: TextField(
            focusNode: _searchTextFocusNode,
            controller: _searchTextController,
            onChanged: (valuee) {
              setState(() {});
            },
            decoration: InputDecoration(
              focusedBorder: OutlineInputBorder(
                borderRadius: BorderRadius.circular(12),
                borderSide:
                  const BorderSide(color: Colors.greenAccent, width: 1),
              ),
              enabledBorder: OutlineInputBorder(
                borderRadius: BorderRadius.circular(12),
                borderSide:
                  const BorderSide(color: Colors.greenAccent, width: 1),
              ),
              hintText: "What's in your mind",
              prefixIcon: const Icon(Icons.search),
              suffix: IconButton(
                onPressed: () {
                  _searchTextController!.clear();
                  _searchTextFocusNode.unfocus();
                },
                icon: Icon(
                  Icons.close,

```

```
        color: _searchTextFocusNode.hasFocus ? Colors.red :  
color,  
  
      ),  
  
    ),  
  
  ),  
  
),  
  
),  
GridView.count(  
  shrinkWrap: true,  
  physics: const NeverScrollableScrollPhysics(),  
  crossAxisCount: 2,  
  padding: EdgeInsets.zero,  
  // crossAxisSpacing: 10,  
  childAspectRatio: size.width / (size.height * 0.59),  
  children: List.generate(allProducts.length, (index) {  
    return ChangeNotifierProvider.value(  
      value: allProducts[index],  
      child: const FeedsWidget(),  
    );  
  }  
),  
),  
]),  
),  
);  
}  
}
```


Launch Screen:



Browse all:

