

**Aim:**

To learn how to include icons, images, and custom fonts in a Flutter application to enhance its visual design.

**Theory:**

Flutter allows the integration of icons, images, and custom fonts to improve UI. Icons represent actions, images display rich content, and custom fonts personalize typography. The Icon widget, Image widget, and the pubspec .yaml file help in adding these elements to the app.

**Code:****menupage.dart**

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

class GoToMenuPage extends StatefulWidget {
  final Map<String, int> cart; // Accept cart as a parameter
  final Function(Map<String, int>) onCartUpdated; // Callback to update parent cart state
  const GoToMenuPage({
    Key? key,
    required this.cart,
    required this.onCartUpdated, // Add this parameter
  }) : super(key: key);
  @override
  State<GoToMenuPage> createState() => _GoToMenuPageState();
}

class _GoToMenuPageState extends State<GoToMenuPage> {
  String _searchTerm = "";
  final TextEditingController _searchController = TextEditingController();
  String _selectedCategory = 'All';
  // Local cart state
  Map<String, int> _cart = {};
  // Local data
```

```
final List<Map<String, dynamic>> menuItems = [
{
    'Name': 'Pizza',
    'Description': 'Delicious cheese pizza WheatBased',
    'Image': 'https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTyMkRu4MFF5OT6HzojWJOq9YTAsvjNYqAYmw&
s',
    'Ingr': 'WheatDough,Cheese,Capsicum,Tomato',
    'Type': 'WheatBased',
    'Calories': 300,
    'Price': 250,
    'CarbonFootprint': '300 gms',
    'showInAll': true, // This item will appear in the "All" category
},
{
    'Name': 'Juice',
    'Description': 'Pure Orange Juice',
    'Image': 'https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRu5r5P2zouN2MCLj2foGmRwE_xG25s9Lwxkw&
s',
    'Ingr': 'Orange',
    'Type': 'WeightLoss',
    'Calories': 100,
    'Price': 100,
    'CarbonFootprint': '200 gms',
    'showInAll': true, // This item will appear in the "All" category
},
{
    'Name': 'Fruit Salad',
    'Description': 'Healthy Fruit Salad',
```

```
'Image': 'https://encrypted-  
tbn0.gstatic.com/images?q=tbn:ANd9GcRXKIQ3slSvcMrIYRs-iUXxIfCVuTwa9QgrQ&s',  
  'Ingr': 'Strawbeery,Blueberry,Raspberry,Kiwi,pomogrenate',  
  'Type': 'WeightLoss',  
  'Calories': 350,  
  'Price': 300,  
  'CarbonFootprint': '200gms',  
  'showInAll': true, // This item will appear in the "All" category  
},  
{  
  'Name': 'Avacado Toast',  
  'Description': 'Avacado Toast with Cottage Cheese Nutrious,Filling and Delicious',  
  'Image': 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQ-Q-  
mQTnFMqYdTeYyyHqyo6dZJOL95ISmFPY_MJENSs5RPhDoMYZCZgGgKWk5Wa9ZL240&us  
qp=CAU',  
  'Ingr': 'Avacado,Dough,Cottage Cheese',  
  'Type': 'WeightGain',  
  'Calories': 350,  
  'Price': 300,  
  'CarbonFootprint': '500gms',  
  'showInAll': true, // This item will appear in both "All" and "WeightGain"  
},  
{  
  'Name': 'Popcorn',  
  'Description': 'Yummy Salted Popcorn',  
  'Image': 'https://encrypted-  
tbn0.gstatic.com/images?q=tbn:ANd9GcQWYksOSYUtzxmteDlS5AAbkM5fJpT7qGRIZg&s',  
  'Ingr': 'Corn,Salt',  
  'Type': 'LightSnack',  
  'Calories': 93,
```

```

        'Price': 150,
        'CarbonFootprint': '200 gms',
        'showInAll': true, // This item will appear in the "All" category
    },
    {
        'Name': 'SugarFree Brownie',
        'Description': 'Low Calorie Brownie',
        'Image': 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT63vcW-
        Cw75HeR0YZcRoHj0yF40uEb21szqyL2VJ2UJajbd1ziebQq0d3jU5AxEp-dBDA&usqp=CAU',
        'Ingr': 'Cocoa,Dough,Milk,DatesBakingPowder',
        'Type': 'SugarFree',
        'Calories': 70,
        'Price': 150,
        'CarbonFootprint': '200 gms',
        'showInAll': true, // This item will appear in the "All" category
    },
];

```

@override

```

void initState() {
    super.initState();
    // Initialize the local cart state with the passed cart
    _cart = Map.from(widget.cart);
}

// Callback to handle quantity changes
void _handleQuantityChanged(String itemId, int quantity) {
    setState(() {
        _cart[itemId] = quantity; // Update the local cart state
    });
}

```

```

print('Cart updated - Item ID: $itemId, New Quantity: $quantity');
print('Updated Cart: $_cart');
// Notify the parent widget of the updated cart state
widget.onCartUpdated(_cart);
}

@override
Widget build(BuildContext context) {
  final double screenWidth = MediaQuery.of(context).size.width;
  final double mainAxisSpacing = screenWidth * 0.02;
  final double crossAxisSpacing = screenWidth * 0.02;
  // Filter menu items based on the selected category and search term
  List<Map<String, dynamic>> filteredMenuItems = menuItems.where((item) {
    bool matchesCategory = _selectedCategory == 'All'
      ? item['showInAll'] // Only show items with showInAll: true in the "All" category
      : item['Type'] == _selectedCategory; // Show items of the selected category
    bool matchesSearch = _searchTerm.isEmpty ||
      item['Name'].toLowerCase().contains(_searchTerm.toLowerCase()) ||
      item['Description'].toLowerCase().contains(_searchTerm.toLowerCase());
    return matchesCategory && matchesSearch;
  }).toList();
  return Padding(
    padding: const EdgeInsets.fromLTRB(0, 10, 0, 10),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Center(
          child: Text(
            'NUTRIEATS',

```

```
style: TextStyle(  
  color: Color.fromARGB(255, 24, 79, 87),  
  fontSize: 45,  
  fontWeight: FontWeight.w500,  
  fontFamily: 'Genos',  
)  
)  
)  
Padding(  
  padding: const EdgeInsets.symmetric(horizontal: 25, vertical: 10),  
  child: TextField(  
    controller: _searchController,  
    cursorColor: Color.fromARGB(255, 24, 79, 87),  
    decoration: InputDecoration(  
      contentPadding: EdgeInsets.symmetric(vertical: 15.0, horizontal: 10.0),  
      border: OutlineInputBorder(  
        borderRadius: BorderRadius.circular(30),  
        borderSide: BorderSide(  
          width: 0.8,  
          color: Color.fromARGB(255, 24, 79, 87),  
        ),  
      ),  
      enabledBorder: OutlineInputBorder(  
        borderRadius: BorderRadius.circular(30),  
        borderSide: BorderSide(  
          width: 0.8,  
          color: Color.fromARGB(255, 49, 49, 49),  
        ),  
      ),  
    ),  
  ),  
)
```

```

),
focusedBorder: OutlineInputBorder(
  borderRadius: BorderRadius.circular(30),
  borderSide: BorderSide(
    width: 1.5,
    color: Color.fromARGB(255, 71, 71, 71),
  ),
),
hintText: 'Search',
hintStyle: TextStyle(fontSize: 18),
prefixIcon: Icon(
  Icons.search,
  size: 30,
),
suffixIcon: IconButton(
  onPressed: () {
    _searchController.clear();
    if (mounted) {
      setState(() {
        _searchTerm = "";
      });
    }
  },
  icon: Icon(Icons.clear),
),
),
onChanged: (value) {
  if (mounted) {

```

```

        setState(() {
          _searchTerm = value.trim();
        });
      }
    },
  ),
),
Padding(
  padding: const EdgeInsets.only(top: 5, left: 25),
  child: Text(
    'Categories',
    style: TextStyle(
      fontFamily: 'Lalezar',
      color: Color.fromARGB(255, 24, 79, 87),
      fontSize: 30,
    ),
  ),
),
),
Padding(
  padding: const EdgeInsets.fromLTRB(0, 5, 0, 20),
  child: SingleChildScrollView(
    scrollDirection: Axis.horizontal,
    child: Container(
      height: 55,
      child: Row(
        children: [
          _buildCategoryButton('All'),
          _buildCategoryButton('WeightLoss'),

```



```

        _buildCategoryButton('WeightGain'),
        _buildCategoryButton('WheatBased'),
        _buildCategoryButton('SugarFree'),
        _buildCategoryButton('LightSnack'),
    ],
),
),
),
),
Expanded(
  child: Padding(
    padding: const EdgeInsets.symmetric(horizontal: 25, vertical: 5),
    child: GridView.builder(
      gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
        crossAxisCount: 2,
        childAspectRatio: 0.75,
        mainAxisSpacing: mainAxisSpacing,
        crossAxisSpacing: crossAxisSpacing,
      ),
      itemCount: filteredMenuItems.length,
      itemBuilder: (context, index) {
        final data = filteredMenuItems[index];

        return MenuItemWidget(
          name: data['Name'],
          des: data['Description'],
          img: data['Image'],
          ingr: data['Ingr'],

```

```

        type: data['Type'],
        cal: data['Calories'],
        price: data['Price'],
        carbonfootprint: data['CarbonFootprint'],
        itemId: index.toString(),
        quantity: _cart[index.toString()] ?? 0, // Get quantity from cart or default to 0
        onQuantityChanged: _handleQuantityChanged, // Pass the callback
      );
    },
  ),
),
),
],
),
);
}

```

```

Widget _buildCategoryButton(String category) {
  return Padding(
    padding: const EdgeInsets.all(3.0),
    child: SizedBox(
      height: 50,
      child: ElevatedButton(
        onPressed: () {
          if (mounted) {
            setState(() {
              _selectedCategory = category;
            });
          }
        },
      ),
    ),
  );
}

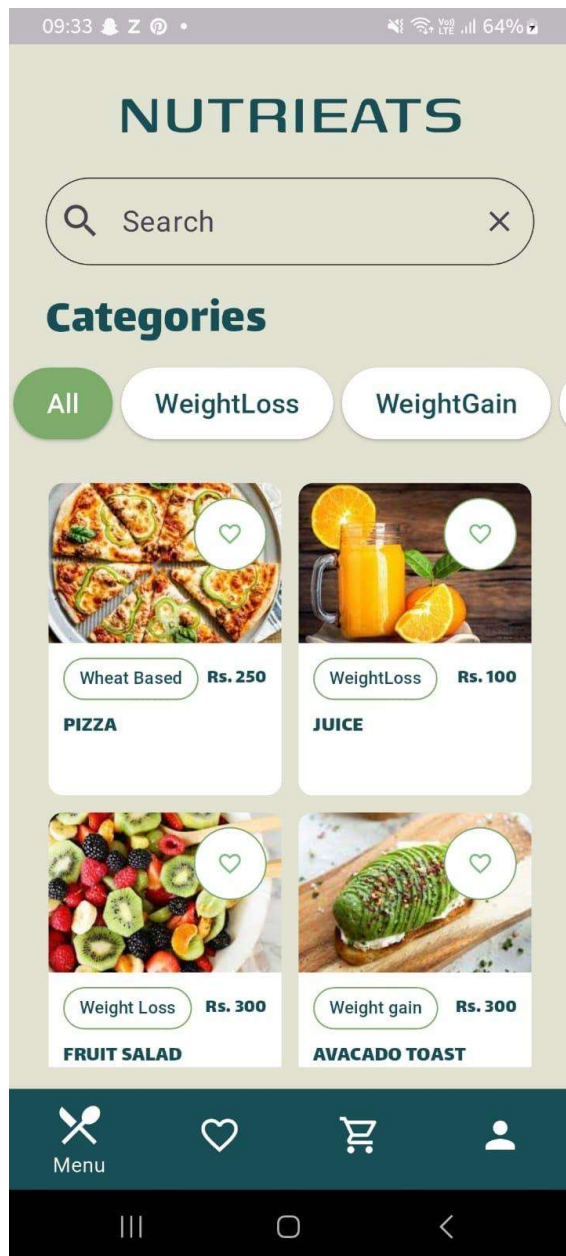
```

```

    }
  },
  style: ElevatedButton.styleFrom(
    backgroundColor: _selectedCategory == category
      ? Color.fromARGB(255, 125, 172, 106)
      : Colors.white,
    foregroundColor: _selectedCategory == category
      ? Colors.white
      : Color.fromARGB(255, 24, 79, 87),
  ),
  child: Text(
    category,
    style: TextStyle(
      fontSize: 17,
    ),
  ),
),
),
),
);
}
}

```

**Output:**



### Conclusion:

This experiment demonstrated how to add icons, images, and custom fonts in a Flutter app, improving its appearance and user experience. It highlighted the flexibility of Flutter in creating a customized and engaging interface.