

MAD PWA Lab

EXPERIMENT NO.3

Name : Pranathi Narsupalli

Div : D15B

Roll no. : 45

AIM : To design Flutter UI by including icons, images and fonts .

1. Create a form with a Global key
2. Add a TextFormField with validation logic
3. Create a button to validate and submit the form

THEORY :

1. Icons:

- Flutter provides a set of built-in icons, and you can use them easily in your application.
- Icons are represented by the 'Icon' widget, and you can use them in various Flutter components such as 'AppBar', 'BottomNavigationBar', and more.
- Icons are categorized based on their type (outlined, rounded, sharp, etc.), and you can find them in the 'Icons' class in the 'material' package.

Example of using an icon in Flutter:

```
Icon(  
  Icons.favorite,  
  color: Colors.red,  
)
```

2. Images:

- Flutter supports various image formats, including PNG, JPEG, GIF, and WebP.
- Images can be loaded using the 'Image' widget or the more advanced 'Image.network' or 'Image.asset' constructors.
- The 'Image.network' constructor loads an image from the network, while 'Image.asset' loads an image from the app's assets.

Example of using an image in Flutter:

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

Example of using an image from assets:

```
Image.asset('assets/images/my_image.png')
```

3. Fonts:

- Flutter allows you to use custom fonts in your application.
- Fonts can be loaded using the 'pubspec.yaml' file, where you define the font file and its family.
- The 'TextStyle' widget is used to apply custom fonts to text in Flutter.

Example of loading a custom font in 'pubspec.yaml':

```
flutter:  
  fonts:  
    - family: MyCustomFont  
      fonts:  
        - asset: assets/fonts/my_custom_font.ttf
```

Example of applying a custom font to text in Flutter:

```
Text(  
  'Hello, Flutter!',  
  style: TextStyle(  
    fontFamily: 'MyCustomFont',  
    fontSize: 20.0,  
  ),  
)
```

INPUT :

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Icons, Fonts, Images Example',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
        fontFamily: 'CustomFont', // Use your custom font family here  
      ),  
    );  
  }  
}
```

```

        home: MyHomePage(),
    );
}
}

class MyHomePage extends StatelessWidget {
  final List<String> imagePaths = [
    'assets/images/bookIcon.png', // Replace with your image assets
    'assets/images/book3.png',
    'assets/images/book.png',
    'assets/images/book0.png',
    'assets/images/book2.png',
    'assets/images/book1.png',
    'assets/images/book6.png',
    'assets/images/book5.png',
  ];

  final List<String> titles = [
    'Title 1', 'Title 2', 'Title 3', 'Title 4',
    'Title 5', 'Title 6', 'Title 7', 'Title 8',
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Row(
          children: [
            Icon(Icons.favorite,
              color: Colors.red,
              size: 48.0,),
            SizedBox(width: 8.0),
            Text('Explore Books'),
          ],
        ),
      ),
      body: Padding(
        padding: EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [

```

```

// Header
Padding(
  padding: EdgeInsets.only(bottom: 16.0),
  child: Row(
    children: [
      Icon(Icons.book, size: 40.0, color: Colors.green[300]),
      SizedBox(width: 16.0),
      Text(
        'Grid of Books',
        style: TextStyle(
          fontSize: 24.0,
        ),
      ),
    ],
  ),
),
// Grid of 8 images
Expanded(
  child: GridView.builder(
    gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
      crossAxisCount: 2,
      crossAxisSpacing: 6.0,
      mainAxisSpacing: 6.0,
    ),
    itemCount: imagePaths.length,
    itemBuilder: (context, index) {
      return Column(
        children: [
          ClipRRect(
            borderRadius: BorderRadius.circular(6.0),
            child: Image.asset(
              imagePaths[index],
              fit: BoxFit.cover,
              height: 100.0, // Adjust the height as needed
            ),
          ),
          SizedBox(height: 8.0),
          Text(
            titles[index],
            style: TextStyle(

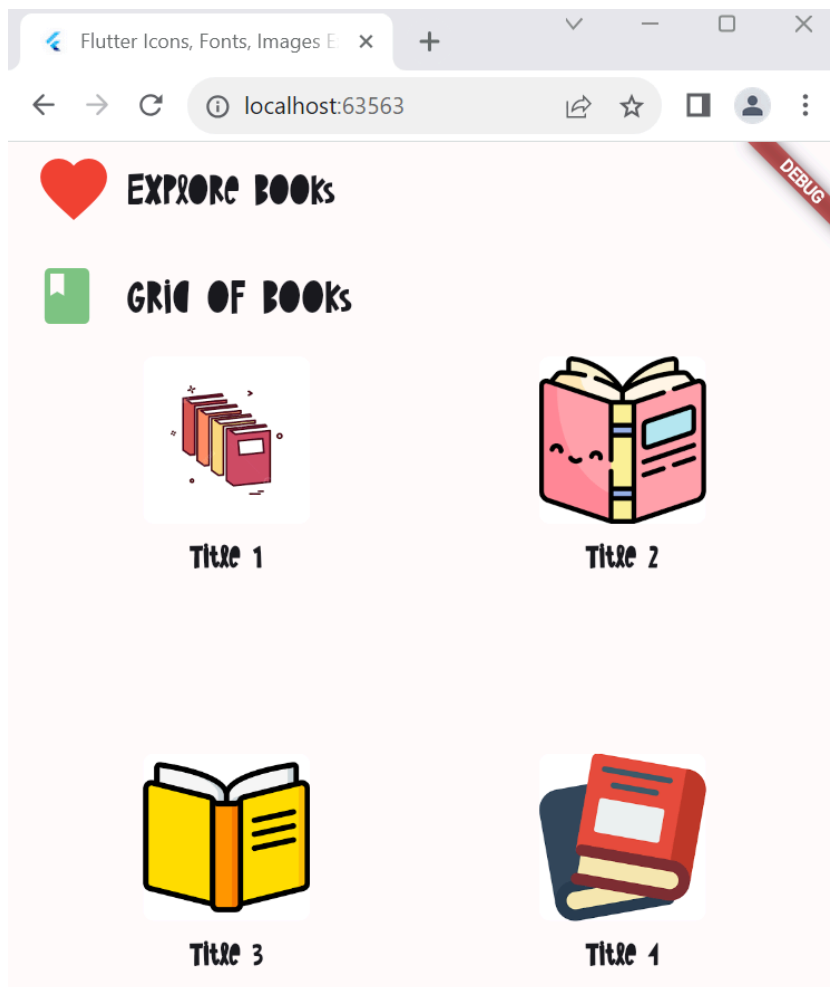
```

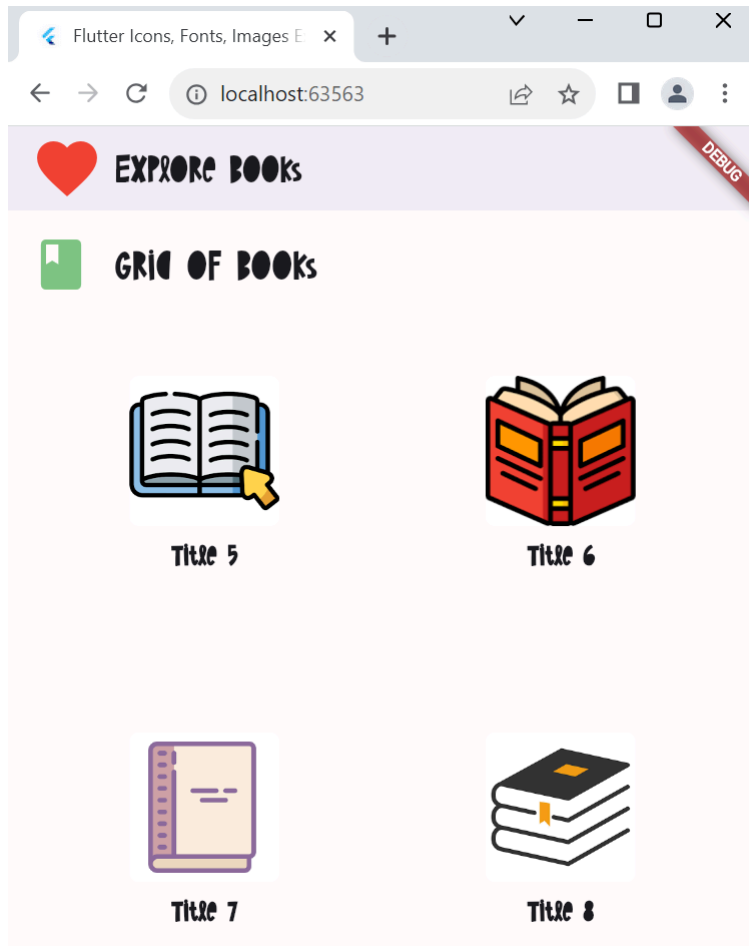
```

fontSize: 16.0,
),
),
],
);
},
),
),
],
),
),
);
}
}

```

OUTPUT :





CONCLUSION : Hence we have understood and implemented icons, images, and custom fonts in Flutter. We mastered the integration of built-in icons for diverse UI elements, loaded images from network and assets, and applied custom fonts to elevate typography. By exploring these essential visual elements, we gained proficiency in crafting versatile and visually engaging Flutter applications. This foundational knowledge sets the stage for further exploration and creative application in future Flutter projects, enhancing our capabilities in UI design and development.