Atharva Mulam Batch B
D15A Roll No: 37

Exp No: 3

Aim: To include icons, images, fonts in Flutter app

Theory:

1) **NetworkImage:**
   Purpose: The NetworkImage widget is specifically designed to load and display images from URLs on the internet.
   Usage: You provide the URL of the image you want to display as a string parameter to the NetworkImage constructor.
   Loading: It handles the process of fetching the image from the network asynchronously, which means it won't block the UI thread while waiting for the image to download.
   Caching: Flutter's image caching mechanism helps improve performance by caching images, reducing unnecessary network requests.

2) **Icon:**
   Purpose: The Icon widget is used to display vector icons in Flutter apps. It's commonly used to represent actions, buttons, or other UI elements.
   IconData: Icons are represented by IconData objects, each of which uniquely identifies an icon. Flutter provides a set of built-in icons through the Icons class.
   Customization: Icons can be customized using the color, size, and semanticLabel properties. Additionally, you can customize the appearance of icons using the IconTheme widget.

3) **Image**
   Purpose: The Image widget is a versatile tool for displaying images in Flutter apps, supporting various image sources such as assets, files, memory, and network URLs.
   Source Types: Depending on the source of the image, you can use different constructors such as AssetImage, FileImage, MemoryImage, and NetworkImage.

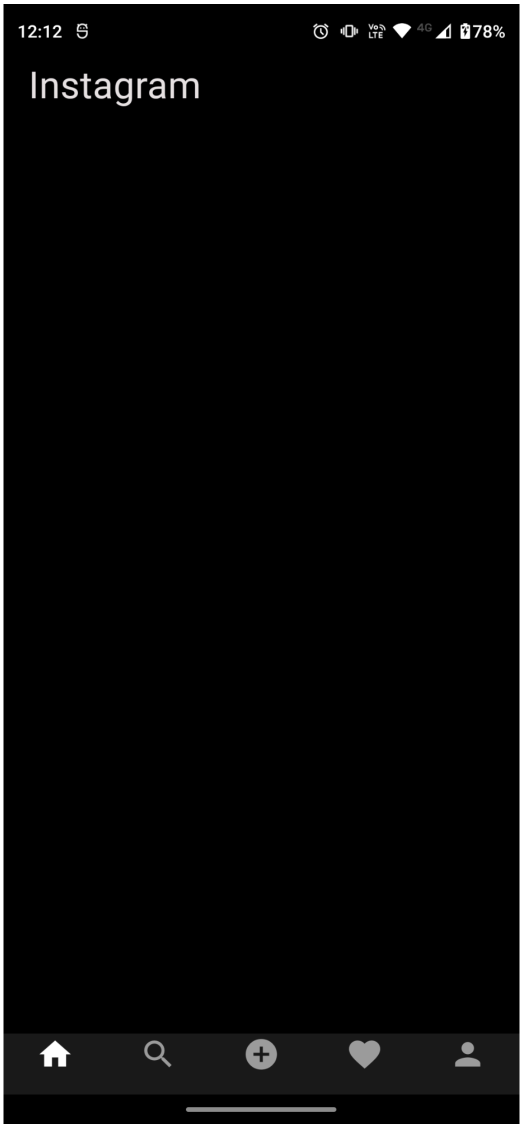
Performance: Flutter optimizes image loading and rendering for better performance. It also provides features like image caching and image format decoding to improve efficiency.

Code:

```
import 'package:flutter/cupertino.dart’;
import 'package:flutter/material.dart’;
import 'package:igclone/utils/colors.dart’;
import 'package:igclone/utils/global_variables.dart’;

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

  @override
  State<MobileScreenLayout> createState() => _MobileScreenLayoutState();
}

class _MobileScreenLayoutState extends State<MobileScreenLayout> {
  int _page = 0;
  late PageController pageController;

  @override
  void initState() {
    super.initState();
    pageController = PageController();
  }

  @override
  void dispose() {
    pageController.dispose();
    super.dispose();
  }

  void onPageChanged(int page) {
    setState(() {
      _page = page;
    });
```

```
}

void navigationTapped(int page) {
  pageController.jumpToPage(page);
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: PageView(
      children: homeScreenItems,
      physics: NeverScrollableScrollPhysics(),
      controller: pageController,
      onPageChanged: onPageChanged,

    ),

    bottomNavigationBar: CupertinoTabBar(
      items: [
        BottomNavigationBarItem(
          icon: Icon(
            Icons.home,
            color: _page == 0 ? primaryColor : secondaryColor,
          ),
          label: '', // Removed empty string label
          backgroundColor: primaryColor,
        ),
        BottomNavigationBarItem(
          icon: Icon(
            Icons.search,
            color: _page == 1 ? primaryColor : secondaryColor,
          ),
          label: '', // Removed empty string label
          backgroundColor: primaryColor,
        ),
        BottomNavigationBarItem(
          icon: Icon(
              Icons.add_circle,
              color: _page == 2 ? primaryColor : secondaryColor,
            ),
            label: ″, // Removed empty string label
            backgroundColor: primaryColor,
          ),
          BottomNavigationBarItem(
            icon: Icon(
              Icons.favorite,
              color: _page == 3 ? primaryColor : secondaryColor,
            ),
            label: ″, // Removed empty string label
            backgroundColor: primaryColor,
          ),
          BottomNavigationBarItem(
            icon: Icon(
              Icons.person,
              color: _page == 4 ? primaryColor : secondaryColor,
            ),
            label: ″, // Removed empty string label
            backgroundColor: primaryColor,
          ),
        ],
        onTap: navigationTapped,
      ),
    );
  }
}
```

Output:



12:12
78%
Instagram