Experiment 3

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

Theory:

        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.

Steps to Add an Image:

Step 1. Create a new folder
- It should be in the root of your flutter project. You can name it whatever you want, but assets are preferred.
- If you want to add other assets to your app, like fonts, it is preferred to make another subfolder named images.

Step 2. Now you can copy your image to images sub-folder. The path should look like assets/images/yourImage. Before adding images also check the above-mentioned supported image formats.

Step 3. Register the assets folder in pubspec.yaml file and update it.
To add images, write the following code:

```
flutter:
  assets:
    - assets/images/yourFirstImage.jpg
    - assets/image/yourSecondImage.jpg
```

If you want to include all the images of the assets folder then add this:

```
flutter:
  assets:
    - assets/images/
```

Step 4. Insert the image code in the file, where you want to add the image.

Step 5. Now you can save all the files and run the app, you will find the output as shown below.

        Icon class in Flutter is used to show specific icons in our app. Instead of creating an image for our icon, we can simply use the Icon class for inserting an icon in our app. For using this class you must ensure that you have set uses-material-design: true in the pubsec.yml file of your object.

Properties:

- color: It is used to set the color of the icon
- size: It is used to resize the Icon
- semanticLable: It comes in play while using the app in accessibility mode (ie, voice-over)
- textDirection: It is used for rendering Icon

Note: The semanticLabel are not visible in the UI.

Code in main.dart:
```dart
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:flutter_youtube_ui/screens/nav_screen.dart';

void main() {
 runApp(ProviderScope(child: MyApp()));
}

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  SystemChrome.setPreferredOrientations([DeviceOrientation.portraitUp]);
  return MaterialApp(
   title: 'Flutter YouTube UI',
   debugShowCheckedModeBanner: false,
   theme: ThemeData(
    brightness: Brightness.dark,
    bottomNavigationBarTheme:
      const BottomNavigationBarThemeData(selectedItemColor: Colors.white),
   ),
   home: NavScreen(),
  );
 }
}
```

Code in video_screen.dart:
```dart
import 'package:flutter/material.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:flutter_youtube_ui/data.dart';
import 'package:flutter_youtube_ui/screens/nav_screen.dart';
import 'package:flutter_youtube_ui/widgets/widgets.dart';
import 'package:miniplayer/miniplayer.dart';

class VideoScreen extends StatefulWidget {
 @override
```

```dart
  _VideoScreenState createState() => _VideoScreenState();
}

class _VideoScreenState extends State<VideoScreen> {
  ScrollController? _scrollController;

  @override
  void initState() {
    super.initState();
    _scrollController = ScrollController();
  }

  @override
  void dispose() {
    _scrollController?.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return GestureDetector(
      onTap: () => context
          .read(miniPlayerControllerProvider)
          .state
          .animateToHeight(state: PanelState.MAX),
      child: Scaffold(
        body: Container(
          color: Theme.of(context).scaffoldBackgroundColor,
          child: CustomScrollView(
            controller: _scrollController,
            shrinkWrap: true,
            slivers: [
              SliverToBoxAdapter(
                child: Consumer(
                  builder: (context, watch, _) {
                    final selectedVideo = watch(selectedVideoProvider).state;
                    return SafeArea(
                      child: Column(
                        children: [
                          Stack(
                            children: [
                              Image.network(
                                selectedVideo!.thumbnailUrl,
                                height: 220.0,
```

```dart
                  width: double.infinity,
                  fit: BoxFit.cover,
                ),
                IconButton(
                  iconSize: 30.0,
                  icon: const Icon(Icons.keyboard_arrow_down),
                  onPressed: () => context
                      .read(miniPlayerControllerProvider)
                      .state
                      .animateToHeight(state: PanelState.MIN),
                ),
              ],
            ),
            const LinearProgressIndicator(
              value: 0.4,
              valueColor: AlwaysStoppedAnimation<Color>(
                Colors.red,
              ),
            ),
            VideoInfo(video: selectedVideo),
          ],
        ),
      );
    },
  ),
),
SliverList(
  delegate: SliverChildBuilderDelegate(
    (context, index) {
      final video = suggestedVideos[index];
      return VideoCard(
        video: video,
        hasPadding: true,
        onTap: () => _scrollController!.animateTo(
          0,
          duration: const Duration(milliseconds: 200),
          curve: Curves.easeIn,
        ),
      );
    },
    childCount: suggestedVideos.length,
  ),
),
],
```

```
          ),
        ),
      ),
    );
  }
}
```

Output: