

Param keswani

D15A 27

Batch B

Experiment no 3

Aim:To include images and fonts in flutter app.

Theory:

Theory: Certainly! Here's a simplified process for adding images in Flutter:

1. Import Libraries:

Ensure that you have the necessary libraries imported in your Dart file. For images, you'll typically use `dart:ui` and other relevant Flutter packages.

2. Adding Local Images:

- Place your local images in the `assets` folder.
- Declare the images in the `pubspec.yaml` file.

3. Adding Network Images:

- Use the `Image.network` widget for displaying images from the internet.

4. Image Widget:

- Create an `Image` widget and provide it with an `ImageProvider`.
- Use `AssetImage` for local images and `NetworkImage` for network images.

5. ImageProvider:

- Understand that `AssetImage` and `NetworkImage` are subclasses of the `ImageProvider` class.

- You can create custom `ImageProvider` if needed.

6. CachedNetworkImage (Optional):

- If you want to cache network images, consider using the `cached\_network\_image` package.

7. Image Loading and Error Handling:

- Customize the loading and error behavior using `loadingBuilder` and `errorBuilder` properties of the `Image` widget or other relevant widgets.

Remember, the actual implementation details might vary based on your specific use case and the packages you choose to use. The key is to understand the concepts of working with local and network images and the various widgets and packages available in Flutter for handling images.

Code:

