# Lab 11

Instructor: Sidra Khatoon E-mail: skhatoon[@uit.edu](mailto:ad@uit.edu)

# Objective

# The objective of lab is exploring inkwell widgets and its usages.

**Student Information**

|  |  |
| --- | --- |
| **Student Name** |  |
| **Student ID** |  |
| **Date** |  |

**Assessment**

|  |  |
| --- | --- |
| **Marks Obtained** |  |
| **Remarks** |  |
| **Signature** |  |

# Objective

# The objective of lab is exploring custom text style , widget style and theme.

# Instructions

You have to perform the following tasks yourselves. Raise your hand if you face any difficulty in understanding and solving these tasks. **Plagiarism** is an abhorrent practice and you should not engage in it.

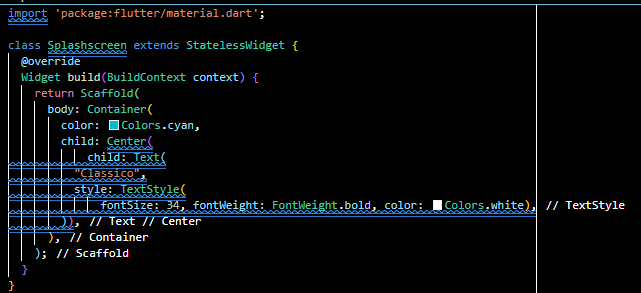
# How to Submit?

Submit lab work using Teams.

**Flutter Splash Screen**

A splash screen is a launch screen, start screen, or boot screen, which is a graphical control element containing the image, logo, and current version of the software. It is the first screen of the app that displays whenever the application is loading. It can also be the app's welcome screen that provides a simple initial experience when a mobile game or program is launching. The splash screen is just a display screen that allows users to look something while the hardware is loading to present software to the user.

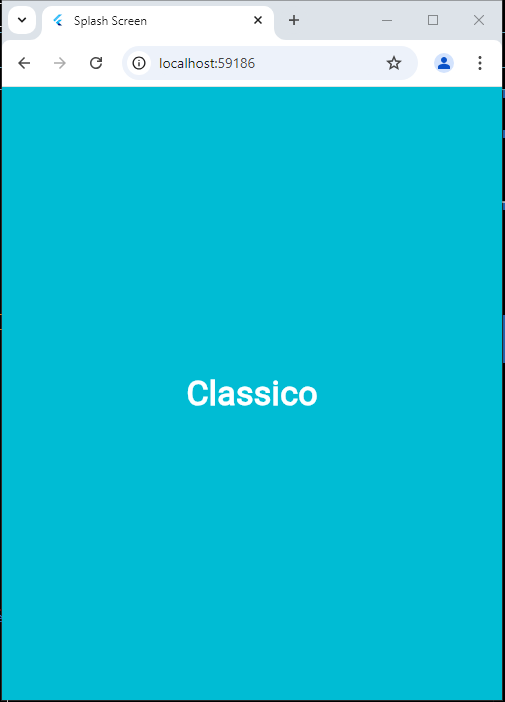
The common elements of a splash screen contain a company name and logo or a title. The most common example of a splash screen is the Flutter logo on starting the Flutter application or Microsoft logo while starting the Microsoft operating system. In this tutorial, we are going to see how a splash screen is created in Flutter application.

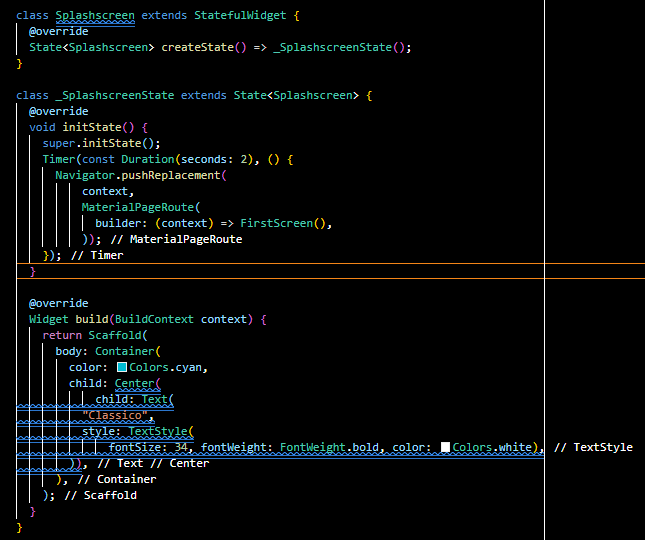
****

**Splash Screen Characteristics**

The following are the essential characteristics of the splash screen:

* It is mainly used for branding or identity recognition of the application and puts the branding impression to users.
* It can also be used to show some loading progress indicator while the hardware is loading to present software to the user.
* When the loading of the splash screen completes, the user gets another functional screen that would be home screen or dashboard, then is forgotten. Once the loading completes, we cannot press the back button to return the splash screen.

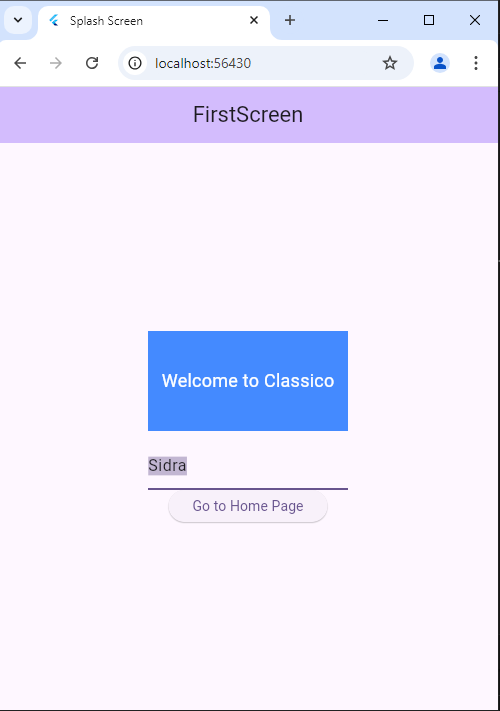
****

****

**Send Data from One Screen to Another Screen**

As a developer, most of the time, we need to pass a data from one screen to another. For this we need

to add a attribute in Second Screen. Before that let’s change the First Screen to [StatefulWidget](https://api.flutter.dev/flutter/widgets/StatefulWidget-class.html) Second Screen as well so that we can use some reactive widget. Let’s add one text field and send the value of the field to second Screen when button pressed.



A screenshot of a computer

Description automatically generated

**Assessment**

Create a simple Flutter app that demonstrates passing data between two screens. The app should have a form on the first screen where users can enter their name and age. Upon submitting, it should navigate to the second screen and display a personalized message with the entered details.

**Requirements**

1. **Screen 1** - InputScreen
   * A TextField for the user’s name.
   * A TextField for the user’s age.
   * A RaisedButton or ElevatedButton to submit and navigate to the next screen.
2. **Screen 2** - DisplayScreen
   * Displays a greeting message like "Hello [Name], you are [Age] years old!"

**Instructions**

1. **Create a Flutter Project**:
   * Set up a new Flutter project (or use an existing one) in your IDE.
2. **Design the Input Screen**:
   * Add two TextField widgets for the user’s name and age.
   * Add a button to navigate to the next screen. Ensure the button is only enabled when both fields are filled.
3. **Pass Data to the Second Screen**:
   * Use the Navigator.push() method to navigate from InputScreen to DisplayScreen.
   * Pass the entered name and age as arguments to DisplayScreen.
4. **Display Data on the Second Screen**:
   * Retrieve the passed arguments in DisplayScreen.
   * Use them to display a personalized message.