**PRACTICAL:05**

**AIM: Create an App to Demonstrate Notifications with different features.**

**Source Code :**

**Main.dart:**

import 'dart:async';

import 'package:flutter/material.dart';

import 'package:flutter\_local\_notifications/flutter\_local\_notifications.dart';

void main() {

  runApp(MaterialApp(

    debugShowCheckedModeBanner: false,

    home: MyApp(),));

}

class MyApp extends StatefulWidget {

  @override

  \_MyAppState createState() => \_MyAppState();

}

class \_MyAppState extends State<MyApp> {

  FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin;

  @override

  void initState() {

    super.initState();

    flutterLocalNotificationsPlugin = new FlutterLocalNotificationsPlugin();

    var android = new AndroidInitializationSettings('@mipmap/ic\_launcher');

    var iOS = new IOSInitializationSettings();

    var initSettings = new InitializationSettings(android: android, iOS: iOS);

    flutterLocalNotificationsPlugin.initialize(initSettings,

        onSelectNotification: onSelectNotification);

  }

  Future onSelectNotification(String payload) {

    debugPrint("payload : $payload");

    showDialog(

      context: context,

      builder: (\_) => new AlertDialog(

        title: new Text('Notification'),

        content: new Text('THIS IS LOCAL NOTIFICATION'),

      ),

    );

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: new AppBar(

        title: new Text('Local Notification'),

      ),

      body: new Center(

        child: new RaisedButton(

          onPressed: showNotification,

          child: new Text(

            'Click for Notification',

            style: Theme.*of*(context).textTheme.headline,

          ),

        ),

      ),

    );

  }

  showNotification() async {

    var android = new AndroidNotificationDetails(

        'channel id', 'channel NAME', 'CHANNEL DESCRIPTION',

        priority: Priority.*high*,importance: Importance.*max*

);

    var iOS = new IOSNotificationDetails();

    var platform = new NotificationDetails(android: android, iOS: iOS);

    await flutterLocalNotificationsPlugin.show(0, '18IT100', 'PRACTICAL-5',platform);

  }

}

**Pubsec.yaml**

name: flutter\_app1

description: A new Flutter application.

*# The following line prevents the package from being accidentally published to*

*# pub.dev using `pub publish`. This is preferred for private packages.*

publish\_to: 'none' *# Remove this line if you wish to publish to pub.dev*

*# The following defines the version and build number for your application.*

*# A version number is three numbers separated by dots, like 1.2.43*

*# followed by an optional build number separated by a +.*

*# Both the version and the builder number may be overridden in flutter*

*# build by specifying --build-name and --build-number, respectively.*

*# In Android, build-name is used as versionName while build-number used as versionCode.*

*# Read more about Android versioning at https://developer.android.com/studio/publish/versioning*

*# In iOS, build-name is used as CFBundleShortVersionString while build-number used as CFBundleVersion.*

*# Read more about iOS versioning at*

*# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html*

version: 1.0.0+1

environment:

  sdk: ">=2.7.0 <3.0.0"

dependencies:

  flutter:

    sdk: flutter

*# The following adds the Cupertino Icons font to your application.*

*# Use with the CupertinoIcons class for iOS style icons.*

cupertino\_icons: ^1.0.0

  flutter\_local\_notifications: ^4.0.1+1

dev\_dependencies:

  flutter\_test:

    sdk: flutter

*# For information on the generic Dart part of this file, see the*

*# following page: https://dart.dev/tools/pub/pubspec*

*# The following section is specific to Flutter.*

flutter:

*# The following line ensures that the Material Icons font is*

*# included with your application, so that you can use the icons in*

*# the material Icons class.*

uses-material-design: true

*# To add assets to your application, add an assets section, like this:*

*# assets:*

*#   - assets/*

*#   - images/a\_dot\_ham.jpeg*

*# An image asset can refer to one or more resolution-specific "variants", see*

*# https://flutter.dev/assets-and-images/#resolution-aware.*

*# For details regarding adding assets from package dependencies, see*

*# https://flutter.dev/assets-and-images/#from-packages*

*# To add custom fonts to your application, add a fonts section here,*

*# in this "flutter" section. Each entry in this list should have a*

*# "family" key with the font family name, and a "fonts" key with a*

*# list giving the asset and other descriptors for the font. For*

*# example:*

*# fonts:*

*#   - family: Schyler*

*#     fonts:*

*#       - asset: fonts/Schyler-Regular.ttf*

*#       - asset: fonts/Schyler-Italic.ttf*

*#         style: italic*

*#   - family: Trajan Pro*

*#     fonts:*

*#       - asset: fonts/TrajanPro.ttf*

*#       - asset: fonts/TrajanPro\_Bold.ttf*

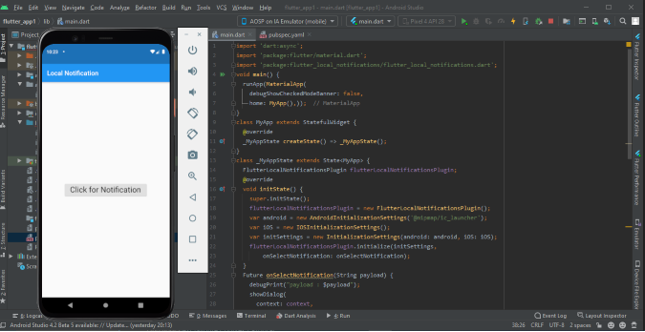
*#         weight: 700*

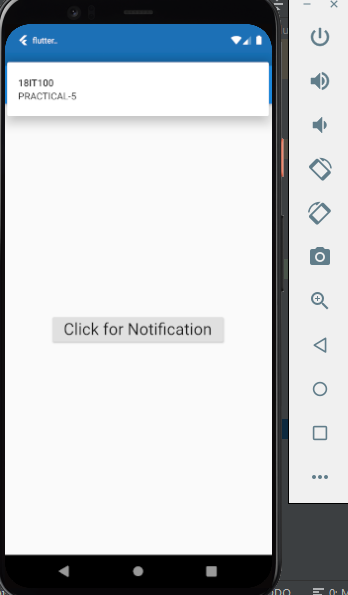
*#*

*# For details regarding fonts from package dependencies,*

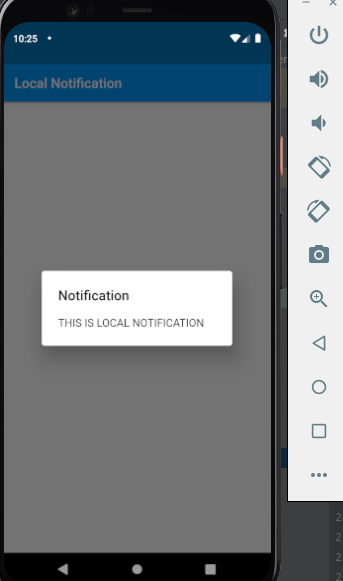
*# see https://flutter.dev/custom-fonts/#from-packages*

**OUTPUT:**





**After clicking on notification**



**Learning Outcome:**  We have learned about the how to print the notification using the local notification plugins of flutter.