

---

# Flutter Local Notifications: Sending Alerts to Users

**Subtitle:** Learn how to schedule and display local notifications in Flutter apps to engage users.

**Website Name:** haas.dev

**Website Link:** <https://dev-roast-app.vercel.app>

---

## Introduction

Notifications keep users **engaged and informed**. Flutter allows you to **send local alerts**, schedule reminders, and trigger notifications without relying on a server.

---

## Step 1: Add Dependencies

```
dependencies:  
  flutter_local_notifications: ^15.1.0
```

Run:

```
flutter pub get
```

Import in Dart:

```
import 'package:flutter_local_notifications/flutter_local_notifications.dart';
```

---

## Step 2: Initialize Notifications

```
final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =  
  FlutterLocalNotificationsPlugin();  
  
void initNotifications() async {  
  const AndroidInitializationSettings initializationSettingsAndroid =  
    AndroidInitializationSettings('@mipmap/ic_launcher');  
  
  const InitializationSettings initializationSettings =  
    InitializationSettings(android: initializationSettingsAndroid);  
  
  await flutterLocalNotificationsPlugin.initialize(initializationSettings);
```

```
}
```

**Exercise:** Initialize notifications in your app's main function.

---

### Step 3: Show a Simple Notification

```
Future<void> showNotification() async {
  const AndroidNotificationDetails androidDetails =
    AndroidNotificationDetails(
      'channelId',
      'channelName',
      importance: Importance.max,
      priority: Priority.high,
    );
  const NotificationDetails generalNotificationDetails =
    NotificationDetails(android: androidDetails);
  await flutterLocalNotificationsPlugin.show(
    0,
    'Hello!',
    'This is a local notification',
    generalNotificationDetails,
  );
}
```

**Exercise:** Display a notification when a button is pressed.

---

### Step 4: Schedule Notifications

```
await flutterLocalNotificationsPlugin.zonedSchedule(
  1,
  'Reminder',
  'Check your tasks',
  tz.TZDateTime.now(tz.local).add(Duration(seconds: 10)),
  const NotificationDetails(
    android: AndroidNotificationDetails('channelId', 'channelName')),
  androidAllowWhileIdle: true,
  uiLocalNotificationDateInterpretation:
    UILocalNotificationDateInterpretation.absoluteTime,
);
```

**Exercise:** Schedule a notification 10 seconds after a button press.

---

## Step 5: Mini Project

Build a **Flutter Reminder App**:

- Schedule one-time or repeating notifications
  - Allow notifications for tasks, events, or messages
  - Bonus: Customize notification sound, icon, and priority
- 

## Key Takeaways

- Local notifications keep users **engaged and informed**
  - Initialization, display, and scheduling are core concepts
  - Notifications are essential for **task management, reminders, and alerts**
  - Mastering notifications improves **app usability and retention**
- 

Skipping notifications reduces **user engagement and timely interactions**, making your app less effective for reminders or alerts.

Visit **haas.dev** for more Flutter guides, tutorials, and complete project resources.

Website Name: haas.dev

Website Link: <https://dev-roast-app.vercel.app>

---