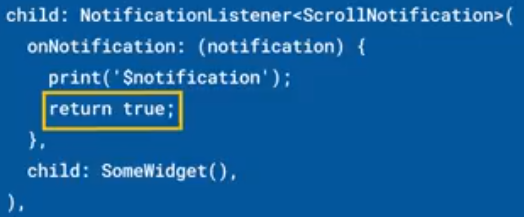
**NotificationListener**

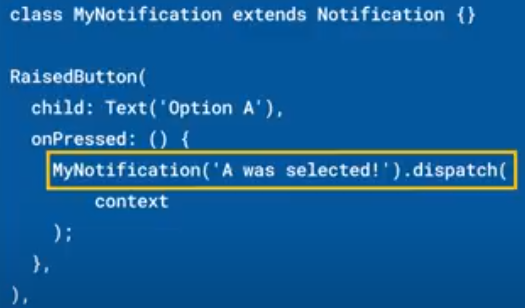
Algunos widgets envían notificaciones a sus padres.

Por ejemplo el ListView envía ScrollNotifications

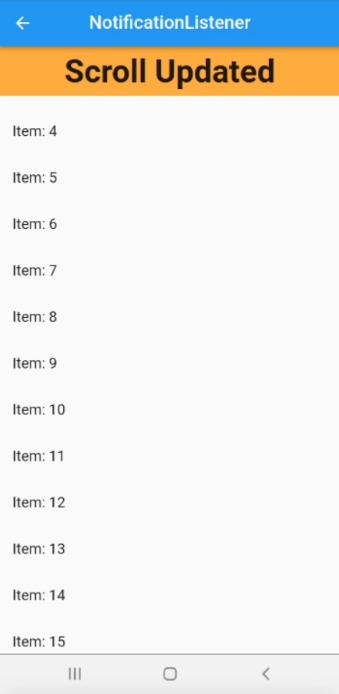
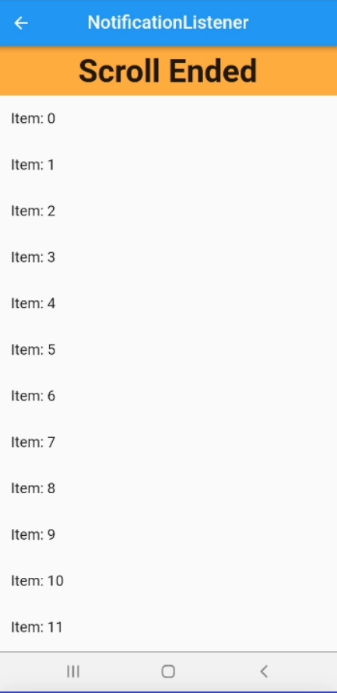
NotificationListener se usa para escuchar esas notificaciones



Se puede extender las clase Notificatrion para enviar las propias notificaciones



|  |  |
| --- | --- |
| **Código** | **Comentarios** |
| import 'package:flutter/material.dart';  class NotificationListenerScreen extends StatefulWidget {    const NotificationListenerScreen({Key? key}) : super(key: key);    @override    State<NotificationListenerScreen> createState() =>        \_NotificationListenerScreenState();  }  class \_NotificationListenerScreenState      extends State<NotificationListenerScreen> {    String message = 'New';    @override    Widget build(BuildContext context) {      return Scaffold(        appBar: AppBar(          title: const Text('NotificationListener'),          centerTitle: true,        ),        body: Column(          children: [            Container(              height: 60,              color: Colors.orangeAccent,              child: Center(                child: Text(                  message,                  style: TextStyle(fontSize: 35, fontWeight: FontWeight.bold),                ),              ),            ),            Expanded(                child: NotificationListener<ScrollNotification>(              onNotification: (scrollNotification) {                // Logic of scrollNotification                if (scrollNotification is ScrollStartNotification) {                  setState(() {                    message = 'Scroll Started';                  });                } else if (scrollNotification is ScrollUpdateNotification) {                  setState(() {                    message = 'Scroll Updated';                  });                } else if (scrollNotification is ScrollEndNotification) {                  setState(() {                    message = 'Scroll Ended';                  });                }                return true;              },              // child of type required. Which means its compulsory to pass in NotificationListener              child: ListView.builder(                  itemCount: 100,                  itemBuilder: (context, index) {                    return ListTile(                      title: Text('Item: $index'),                    );                  }),            ))          ],        ),      );    }  } |  |

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -