Day 25: Local Notifications

Today, we will explore how to create Local Notifications in Xamarin.Android.

Notifications in Android are use two types of layouts –

1. Base Layout
2. Expanded Layout

# Base Layout Notification

Base Layout Notification is the simplest form of Android Notifications. The Base Layout Notifications consist of the following –

* A Notification Icon. This is a simple Android Drawable Resource Image that indicates either the originating app icon or type of the notification shown.
* A title for the notification.
* Additional text for the notification.
* Timestamp at which the notification is generated.

Here is an example of Base Layout Notification, which shows where all the parts of the Layout fit in –



Image Courtesy: <http://developer.android.com/design/patterns/notifications.html>

Now let’s see how to create this Base Layout Notification in code –

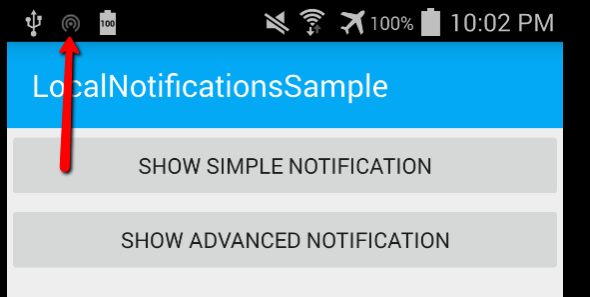
To create any Notification in Android, be it be base or expanded, we need the Notification.Builder class. We then set Notification Icon, Content Title, and Content Text for the Notification. The timestamp part of the notification is automatically added by Android to the current system time but there are ways to override timestamp.

Once we have a builder object, we need to call **Build()** factory method on the builder to create a Notification. And finally, to show the Notification, we need the NotificationManager class and we call the **Notify** method on it passing in the notification and the id of the notification.

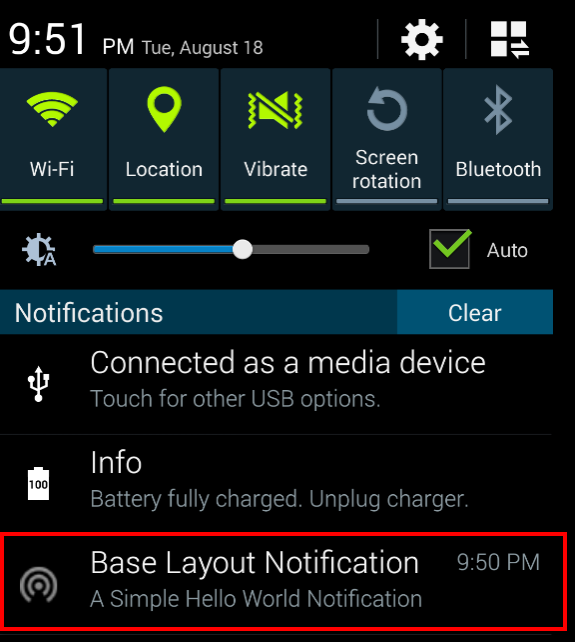
|  |
| --- |
| private void SimpleNotificationButton\_Click(object sender, EventArgs e)  {  Notification.Builder builder = new Notification.Builder(this)  .SetContentTitle("Base Layout Notification")  .SetContentText("A Simple Hello World Notification")  .SetSmallIcon(Resource.Drawable.notifcation\_icon);  Notification notification = builder.Build();  NotificationManager notificationManager = GetSystemService(NotificationService) as NotificationManager;  const int notificationId = 0;  notificationManager.Notify(notificationId, notification);  } |

Gist file link: <https://gist.github.com/vkoppaka/c9e3c4155b99c8b8946e>

The notification icon is also shown on the top widget notification bar in Android –



And if we swipe down to reveal all notifications in Android –



# Expanded Layout Notification

Expanded Layout Notification allow us to show much more detail of the notification. We can show summary of a notification message which can be further expanded to reveal larger image and more text.

Here is an example of Expanded Layout Notification, which shows where all the parts of the Layout fit in –

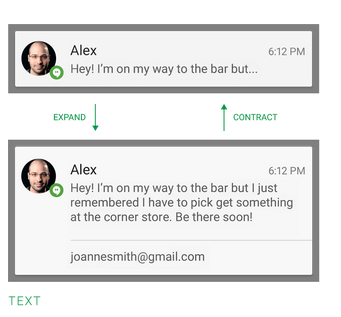


Image Courtesy: <http://developer.android.com/design/patterns/notifications.html>

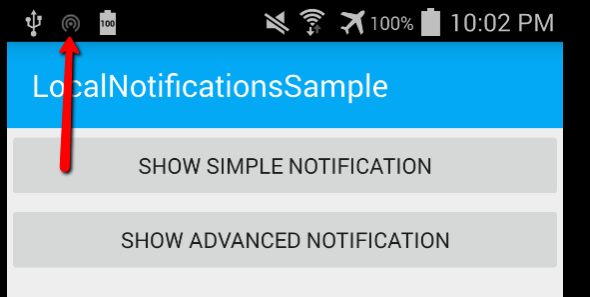
Now let’s see how to create this Expanded Layout Notification in code –

As we did above, we will need the Notification.Builder class to set the Title, text and icons. We then need to work in setting BigTextStyle for the notification using the Notification.BigTextStyle class. The Notification.BigTextStyle class has a handy BigText() method which is used to set Big Text. We can optionally also set a summary text. Once we have the Style created, we just use the Builder class and call **SetStyle** method on it. And the rest of notification creation is similar to Base Layout Notification.

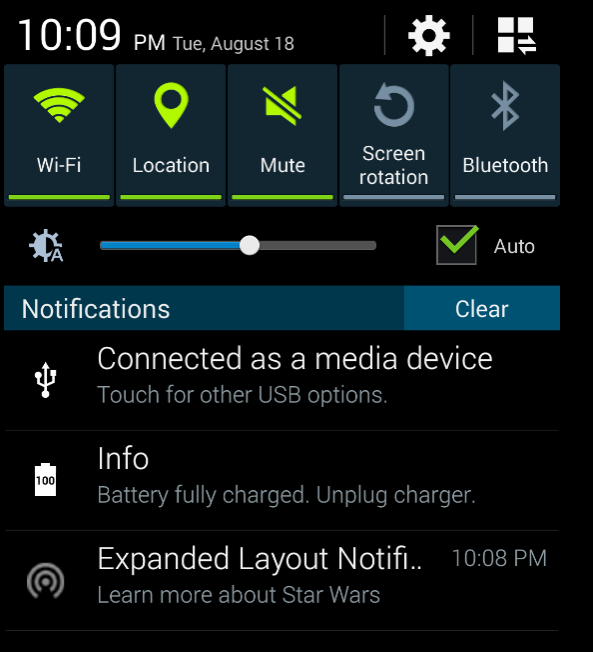
|  |
| --- |
| private void AdvancedNotificationButton\_Click(object sender, EventArgs e)  {  Notification.Builder builder = new Notification.Builder(this)  .SetContentTitle("Expanded Layout Notification")  .SetContentText("Learn more about Star Wars")  .SetSmallIcon(Resource.Drawable.notifcation\_icon);  Notification.BigTextStyle textStyle = new Notification.BigTextStyle();  string longTextMessage = "Lucas ipsum dolor sit amet darth alderaan droid kessel organa jango leia amidala leia aayla. Darth lars sidious grievous. Mara mara wampa skywalker dantooine mon. Watto sith calamari lobot organa qui-gonn alderaan. Boba watto yoda sidious skywalker skywalker ahsoka skywalker.";  textStyle.BigText(longTextMessage);  textStyle.SetSummaryText("The Star Wars Ipsum");  builder.SetStyle(textStyle);  Notification notification = builder.Build();  NotificationManager notificationManager = GetSystemService(NotificationService) as NotificationManager;  const int notificationId = 1;  notificationManager.Notify(notificationId, notification);  } |

Gist file link: <https://gist.github.com/vkoppaka/ecc3ef502dcd22427ab6>

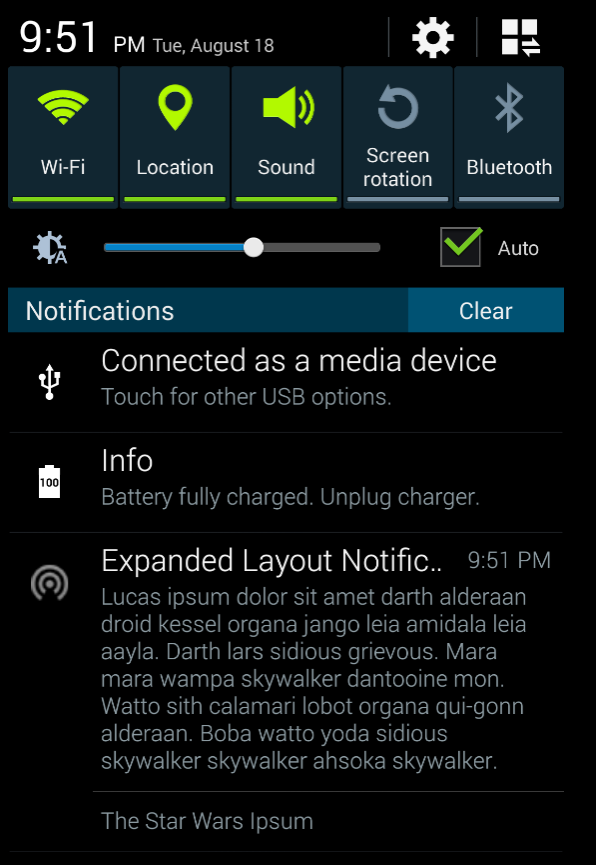
The notification icon is also shown on the top widget notification bar in Android –



And if we swipe down to reveal all notifications in Android –



We see the Expanded Notification with Content Text. And if we use the two finger swipe down gesture on the notification, we can then see the full notification message we set –



That’s it for today, tomorrow, we will explore how to trigger these notifications using a remote service.