

Alert Dialog Example

```
Button btnAlert = findViewById(R.id.btnAlert);
btnAlert.setOnClickListener(v -> {

    AlertDialog.Builder alert = new AlertDialog.Builder(NotiActivity.this);
    alert.setMessage("Login Success");
    alert.setTitle("This is title Page!");
    alert.setIcon(R.drawable.alert);
    alert.setCancelable(true);
    alert.setPositiveButton("OK", (dialog, which) -> {
        Toast.makeText(NotiActivity.this, "OK Clicked",
            Toast.LENGTH_LONG).show();
    });

    alert.setNegativeButton("CANCEL", (dialog, which) -> {
        Toast.makeText(NotiActivity.this, "Cancel Clicked",
            Toast.LENGTH_LONG).show();
    });
    alert.show();

});
```

(Note: Replace NotiActivity.this with your Activity class Name do not use getApplicationContext())

Simple Snackbar

Below is the syntax of a simple snackbar. The **make** function accepts three parameters. View, display message and duration of the message to be displayed.

Normally passing **CoordinatorLayout** as view param is the best option as it allows Snackbar to enable some features like swipe-to-dismiss and automatically moving of widgets like FloatingActionButton.

And the duration should be **LENGTH_SHORT**, **LENGTH_LONG** or **LENGTH_INDEFINITE**. When **LENGTH_INDEFINITE** is used, the snackbar will be displayed indefinite time and can be dismissed with swipe off.

```
Snackbar snackbar = Snackbar
    .make(coordinatorLayout, "Welcome to NAST College",
        Snackbar.LENGTH_LONG);

snackbar.show();
```

Note: coordinatorLayout is the id of the layout is set in the setContentView() layout.

```
LinearLayout coordinatorLayout=findViewById(R.id.coordinatorLayout);
```

Snackbar with Action Callback

You can also mention a callback interaction method using *setAction()* method. This allows us to take certain action when user interacts with the snackbar.

```
Snackbar snackbar = Snackbar
    .make(coordinatorLayout, "Message is deleted", Snackbar.LENGTH_LONG)
    .setAction("UNDO", view -> {
        Snackbar snackbar1 = Snackbar.make(coordinatorLayout, "Message is
        restored!", Snackbar.LENGTH_SHORT);
        snackbar1.show();
    });

snackbar.show();
```

Customizing the Snackbar View

Snackbar comes with default **white** color text and **#323232** background color. You can override these colors as mentioned below.

```
Snackbar snackbar = Snackbar
    .make(coordinatorLayout, "No internet connection!",
Snackbar.LENGTH_LONG)
    .setAction("RETRY", view -> {
        Toast.makeText(getApplicationContext(), "Retrying...",
Toast.LENGTH_SHORT).show();
    });

// Changing message text color
snackbar.setActionTextColor(Color.RED);

// Changing action button text color
View sbView = snackbar.getView();
TextView textView = (TextView)
sbView.findViewById(com.google.android.material.R.id.snackbar_text);
textView.setTextColor(Color.YELLOW);
snackbar.show();
```

Custom Toast

To create a custom toast notification, you need to create a layout with custom controls as shown below:

customtoast.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:id="@+id/customToastLayout"
        android:background="#21ab29"
        android:orientation="horizontal"
        android:padding="10dp">

        <ImageView
            android:id="@+id/customToastImage"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@android:drawable/ic_dialog_alert" />

        <TextView
            android:id="@+id/customToastMessage"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:paddingLeft="10dp"
            android:text="This is my custom toast message"
            android:textColor="#fff" />
    </LinearLayout>

```

After design this custom toast layout you need to work on Activity file as:

```

//Creating the LayoutInflater instance
LayoutInflater li = getLayoutInflater();

//Getting the View object as defined in the customtoast.xml file
View layout = li.inflate(R.layout.customtoast, (ViewGroup)
    findViewById(R.id.customToastLayout));

//Creating the Toast object
Toast toast = new Toast(getApplicationContext());
toast.setDuration(Toast.LENGTH_SHORT);
toast.setGravity(Gravity.CENTER_VERTICAL, 0, 0);
toast.setView(layout); //setting the view of custom toast layout
toast.show();

```

Notification Example

Add permission to AndroidManifest.xml file

```

<uses-permission android:name="android.permission.POST_NOTIFICATIONS" />

```

Create Notification Channel and add to manifest file

```

<application
    android:name=".App"
    .
    .
    .
>

public class App extends Application {
    public static final String CHANNEL_1_ID="channel1";
    public static final String CHANNEL_2_ID="channel2";

    @Override
    public void onCreate() {
        super.onCreate();
        createNotiChannel();
    }

    private void createNotiChannel() {
        if (android.os.Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
            NotificationChannel channel1 = new NotificationChannel(
                CHANNEL_1_ID, "Channel 1",
                NotificationManager.IMPORTANCE_HIGH);
            channel1.setDescription("This is channel 1");

            NotificationChannel channel2 = new NotificationChannel(
                CHANNEL_2_ID, "Channel 2",
                NotificationManager.IMPORTANCE_LOW);
            channel2.setDescription("This is channel 2");

            NotificationManager manager
            =getSystemService(NotificationManager.class);
            manager.createNotificationChannel(channel1);
            manager.createNotificationChannel(channel2);
        }
    }
}

```

Design layout as:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".NotiActivity">

```

```

<EditText
    android:id="@+id/etTitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Notification Title" />

<EditText
    android:id="@+id/etDetail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Notification Details" />

<Button
    android:id="@+id/btnCh1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Channel One" />

<Button
    android:id="@+id/btnCh2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Channel Two" />
</LinearLayout>

```

Create NotiActivity class as

```

public class NotiActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_noti);

        NotificationManagerCompat notificationManager =
        NotificationManagerCompat.from(this);
        EditText etTitle = findViewById(R.id.etTitle);
        EditText etDetail = findViewById(R.id.etDetail);
        Button btnOne = findViewById(R.id.btnCh1);
        Button btnTwo = findViewById(R.id.btnCh2);

        btnOne.setOnClickListener(view -> {
            Notification notification = new
            NotificationCompat.Builder(getApplicationContext(), App.CHANNEL_1_ID)
                .setContentTitle(etTitle.getText().toString())
                .setTicker("one")

```

```

        .setWhen(System.currentTimeMillis())
        .setSmallIcon(R.mipmap.ic_launcher)
        .setContentText(etDetail.getText().toString())
        .setPriority(NotificationCompat.PRIORITY_HIGH)
        .setCategory(NotificationCompat.CATEGORY_MESSAGE)
        .build();
    if (ActivityCompat.checkSelfPermission(getApplicationContext(),
android.Manifest.permission.POST_NOTIFICATIONS) !=
PackageManager.PERMISSION_GRANTED) {
        return;
    }
    notificationManager.notify(1, notification);
});

    btnTwo.setOnClickListener(view -> {
        Notification notification2 = new
NotificationCompat.Builder(getApplicationContext(),
App.CHANNEL_2_ID)
        .setContentTitle(etTitle.getText().toString())
        .setTicker("two")
        .setSmallIcon(R.drawable.ic_launcher_foreground)
        .setContentText(etDetail.getText().toString())
        .setPriority(NotificationCompat.PRIORITY_LOW)
        .setCategory(NotificationCompat.CATEGORY_MESSAGE)
        .build();
    if (ActivityCompat.checkSelfPermission(getApplicationContext(),
        android.Manifest.permission.POST_NOTIFICATIONS) !=
PackageManager.PERMISSION_GRANTED) {
        return;
    }
    notificationManager.notify(2, notification2);
});
}

}

```