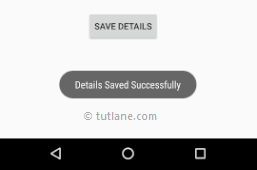
**2. Android Toast with Examples**

In android, **Toast** is a small popup notification which is used to display an information about the operation which we performed in our app. The Toast will show the message for a small period of time and it will disappear automatically after a timeout.

Generally, the size of Toast will be adjusted based on the space required for the message and it will be displayed on the top of main content of an [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle) for a short period of time.

For example, some of the apps will show a message like “**Press again to exit**” in toast, when we pressed a back button in the home page or showing a message like “**saved successfully**” toast, when we click on button to save the details.

Following is the pictorial representation of using **Toast** in android applications.



**Create a Toast in Android**

In android, we can create a Toast by instantiating an **android.widget.Toast** object using **makeText()** method. The **makeText()** method will take three parameters: application context, text message and the duration for the toast. We can display the **Toast notification** by using **show()** method.

Following is the syntax of creating a **Toast** in android applications.

Toast.makeText(context, "message", duration).show();

If we observe above syntax, we defined a Toast **notification** using **makeText()** method with three parameters, those are

| **Parameter** | **Description** |
| --- | --- |
| context | It’s our application context. |
| message | It’s our custom message which we want to show in Toast notification. |
| duration | It is used to define the duration for notification to display on screen. |

We have a two ways to define the Toast **duration**, either in **LENGTH\_SHORT** or **LENGTH\_LONG** to display the toast notification for short or longer period of time.

Following is the example of defining a Toast in android applications.

Toast.makeText(MainActivity.this, "Details Saved Successfully.", Toast.LENGTH\_SHORT).show();

Now we will see how to implement a Toast notification in android applications with examples.

**Android Toast Notification Example**

Create a new android application using android studio and give names as **ToastExample**.

Now open an **activity\_main.xml** file from **\res\layout** path and write the code like as shown below

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout\_width="match\_parent"  
    android:layout\_height="match\_parent"  
    android:orientation="vertical" >  
  
    <Button  
        android:id="@+id/btnShow"  
        android:layout\_width="wrap\_content"  
        android:layout\_height="wrap\_content"  
        android:text="Show Toast"  
        android:layout\_marginTop="200dp" android:layout\_marginLeft="140dp"/>  
</LinearLayout>

If we observe above code we created a one [Button](https://www.tutlane.com/tutorial/android/android-button-with-examples) control in XML Layout file to show the toast **notification** when we click on [Button](https://www.tutlane.com/tutorial/android/android-button-with-examples).

Once we are done with creation of layout with required controls, we need to load the XML layout resource from our [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle) **onCreate()** callback method, for that open main [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle) file **MainActivity.java** from **\java\com.tutlane.toastexample** path and write the code like as shown below.

**MainActivity.java**

package com.tutlane.toastexample;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity\_main);  
        Button btn = (Button)findViewById(R.id.btnShow);  
        btn.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                Toast.makeText(MainActivity.this, "You Clicked on Button..", Toast.LENGTH\_SHORT).show();  
            }  
        });  
    }  
}

If we observe above code we are created a toast notification using **makeText()** method and showing a toast notification on [Button](https://www.tutlane.com/tutorial/android/android-button-with-examples) click.

Generally, during the launch of our [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle), **onCreate()** callback method will be called by android framework to get the required layout for an [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle).

**Output of Android Toast Notification Example**

When we run above example using android virtual device (AVD) we will get a result like as shown below.



If we observe above result we created a toast notification and shown it on Button click based on our requirements.

**Change the Position of Android Toast Notification**

By default, the android Toast notification will always appear near the bottom of the screen, centred horizontally like as shown in above image.

In case if we want to change the position of Toast notification, we can do it by using **setGravity(int, int, int)** method. The **setGravity()** method will accepts three parameters: a **Gravity** constant, an **x-position** offset, and a **y-position** offset.

Following is the example of changing the position of android Toast notification to top-right based on offset positions by using **setGravity()** method.

Toast toast =  Toast.makeText(MainActivity.this, "You Clicked on Button..", Toast.LENGTH\_SHORT);  
toast.setGravity(Gravity.TOP|Gravity.RIGHT, 100, 250);  
toast.show();

If we want to move the position of toast to right, increase the value of the **second** parameter. To move it down, increase the value of the **last** parameter.

**Android Toast Positioning Example**

Following is the example of changing the position of android toast notification to top-right side using **setGravity()**method.

We need to modify our main [activity](https://www.tutlane.com/tutorial/android/android-activity-lifecycle) file **MainActivity.java** code like as shown below.

**MainActivity.java**

package com.tutlane.toastexample;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.Gravity;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity\_main);  
        Button btn = (Button)findViewById(R.id.btnShow);  
        btn.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                //display toast message in top right side  
              Toast toast =  Toast.makeText(MainActivity.this, "You Clicked on Button..", Toast.LENGTH\_SHORT);  
                toast.setGravity(Gravity.TOP|Gravity.RIGHT, 100, 250);  
                toast.show();  
            }  
        });  
    }  
}

If we observe above code we are changing the position of android toast notification using **setGravity()** property.

**Output of Android Toast Positioning Example**

When we run above example using android virtual device (AVD) we will get a result like as shown below.



If we observe above result we changed the position of toast notification to top right side on [Button](https://www.tutlane.com/tutorial/android/android-button-with-examples) click based on our requirements.

In case if we want to change the style of toast notification, then we can do it by creating a custom XML layout file.