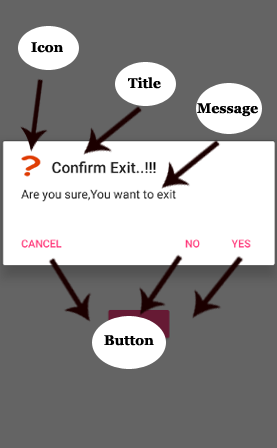
**AlertDialog.Builder Components Used In Alert Dialog**

AlertDialog.Builder is used to create an interface for Alert Dialog in Android for setting like alert title, message, image, button, button onclick functionality etc.

AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(this);

Below are the components of Alert Dialog:

**1. setTitle(CharSequence title) –** This component is used to set the title of the alert dialog. It is optional component.

// Setting Alert Dialog Title

alertDialogBuilder.setTitle("Confirm Exit..!!!");

**2. setIcon(Drawable icon) –** This component add icon before the title. You will need to save image in drawable icon.

// Icon Of Alert Dialog

alertDialogBuilder.setIcon(R.drawable.question);

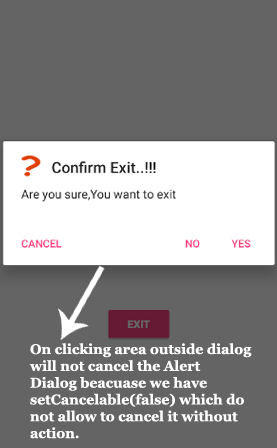
**3. setMessage(CharSequence message)** – This component displays the required message in the alert dialog.

// Setting Alert Dialog Message

alertDialogBuilder.setMessage("Are you sure,You want to exit");

**4. setCancelable(boolean cancelable) –** This component has boolean value i.e true/false. If set to false it allows to cancel the dialog box by clicking on area outside the dialog else it allows.

alertDialogBuilder.setCancelable(false);

  
**5. setPositiveButton(CharSequence text, DialogInterface.OnClickListener listener)**– This component add positive button and further with this user confirm he wants the alert dialog question to happen.

alertDialogBuilder.setPositiveButton("yes", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface arg0, int arg1) {

finish();

}

});

**6. setNegativeButton(CharSequence text, DialogInterface.OnClickListener listener) –**This component add negative button and further with this user confirm he doesn’t want the alert dialog question to happen.

alertDialogBuilder.setNegativeButton("No", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Toast.makeText(MainActivity.this,"You clicked over No",Toast.LENGTH\_SHORT).show();

}

});

**7. setNeutralButton(CharSequence text, DialogInterface.OnClickListener listener) –** This component simply add a new button and on this button developer can set any other onclick functionality like cancel button on alert dialog.

alertDialogBuilder.setNeutralButton("Cancel", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Toast.makeText(getApplicationContext(),"You clicked on Cancel",Toast.LENGTH\_SHORT).show();

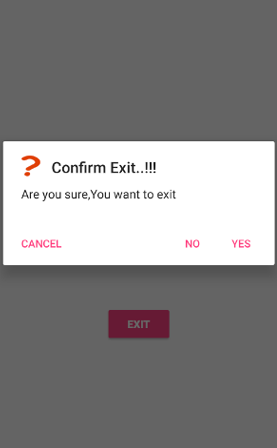
}

});

**Alert Dialog Example In Android Studio**

Below is the example of Alert Dialog in which the functionality of Alert Dialog is defined over button click. In this example we have used a simple button and on that button click the alert dialog window will appear.

Below you can download code, see final output and step by step explanation of Alert Dialog example in Android Studio.



**Step 1:** Create a new project and name it **AlertDialogExample**.

**Step 2:** Open res -> layout -> activity\_main.xml (or) main.xml and add following code:

Here we add the button UI on click of which alert dialog will appear. We also used textview for asking user to click on button.

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/activity\_main"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.alertdialogexample.MainActivity">

<Button

android:text="@string/exit"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/button"

android:onClick="exit"

android:textStyle="normal|bold"

style="@style/Widget.AppCompat.Button.Colored"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:layout\_marginBottom="131dp"/>

<TextView

android:text="@string/click\_over\_button\_to\_exit"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="68dp"

android:id="@+id/textView2"

android:layout\_above="@+id/button"

android:layout\_centerHorizontal="true"

android:textSize="18sp"

android:textStyle="normal|bold"

android:gravity="center" />

</RelativeLayout>

**Step 3 :** Now open app -> java -> package -> MainActivity.java and add the below code.

In this step firstly AlertDialog.Builder is used to create a interface like setting alert title, message, image, button, button onclick functionality etc.

**Important Note:** Add a image to the drawable folder before setting the alert dialog icon.

package com.example.alertdialogexample;

import android.content.DialogInterface;

import android.support.v7.app.AlertDialog;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

public void exit(View view){

AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(this);

// Setting Alert Dialog Title

alertDialogBuilder.setTitle("Confirm Exit..!!!");

// Icon Of Alert Dialog

alertDialogBuilder.setIcon(R.drawable.question);

// Setting Alert Dialog Message

alertDialogBuilder.setMessage("Are you sure,You want to exit");

alertDialogBuilder.setCancelable(false);

alertDialogBuilder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface arg0, int arg1) {

finish();

}

});

alertDialogBuilder.setNegativeButton("No", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Toast.makeText(MainActivity.this,"You clicked over No",Toast.LENGTH\_SHORT).show();

}

});

alertDialogBuilder.setNeutralButton("Cancel", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

Toast.makeText(getApplicationContext(),"You clicked on Cancel",Toast.LENGTH\_SHORT).show();

}

});

AlertDialog alertDialog = alertDialogBuilder.create();

alertDialog.show();

}

}

**Output:**

Now run the App and click on the button. The alert dialog will appear asking user to confirm if he wants to exit App. If user click on ‘No’, he remains in the App and if he click on ‘Yes’ the user will get exit from the App.