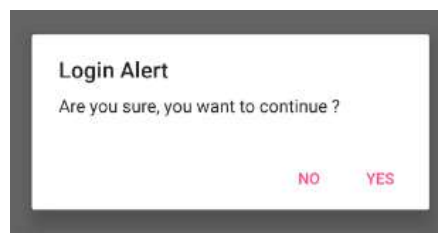# Android UI Control
[AlertDialog]

Android – UI Control

AlertDialog

- **Dialog** is a small window that prompt messages to the user to make a decision or enter additional details.

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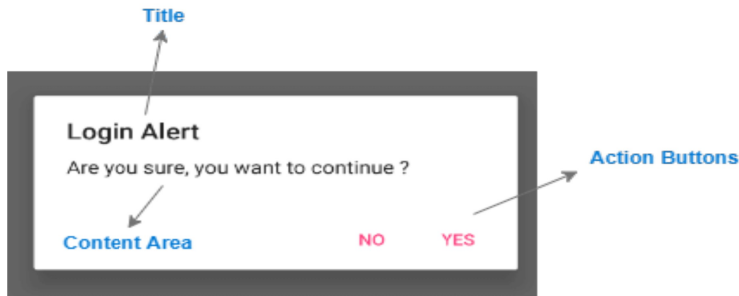

- In android, we have a different type of **Dialogs** available, those are

Dialog	Description
AlertDialog	This dialog is used to display prompt to the user with title, upto three buttons, list of selectable items or a custom layout.
DatePickerDialog	This dialog is a predefined UI control and it allows a user to select Date.
TimePickerDialog	It's a predefined UI control and it allows a user to select Time.

AlertDialog

- AlertDialog is used to prompt a dialog to the user with messages and **button** to perform an action to proceed further.

- The AlertDialog in an android application will contain three regions like as shown below.

- **Alert Dialogs**, we can show a **title**, up to three **buttons**, a list of selectable items or a custom layout based on our requirements.

AlertDialog

Region	Description
Title	It's optional and it can be used to show the detailed messages based on our requirements.
Content Area	It is used to display a message, list or other custom layouts based on our requirements.
Action Buttons	It is used to display action buttons to interact with users. We can use upto 3 different action buttons in alert dialogs, such as positive, negative and neutral.

- In android we can build AlertDialog in our **activity** file using different dialog methods.

Android AlertDialog Methods

Method	Description
setTitle()	It is used to set the title of alertdialog and its an optional component.
setIcon()	It is used to set the icon before the title
setMessage()	It is used to set the message required message to display in alertdialog.
setCancelable()	It is used to allow users to cancel alertdialog by clicking on outside of dialog area by setting true/false.
setPositiveButton()	It is used to set the positive button for alertdialog and we can implement click event of a positive button.
setNegativeButton()	It is used to set the negative button for alertdialog and we can implement click event of a negative button.
setNeutralButton()	It is used to set the neutral button for alertdialog and we can implement click event of a neutral button.

5

Android AlertDialog Example

- In this example we define one **Button** control in **RelativeLayout** to show the AlertDialog and get the action that was performed by a user on **Button** click in the android application.

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

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

Android AlertDialog Example

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent">
    <Button
     android:id="@+id/getBtn"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:layout_marginLeft="150dp"
     android:layout_marginTop="200dp"
     android:text="Show Alert" />
</RelativeLayout>
```

Android AlertDialog Example

```
MainActivity.java
package com.tutlane.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.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.getBtn);
```

Android AlertDialog Example

```
btn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
            builder.setTitle("Login Alert")
                .setMessage("Are you sure, you want to continue ?")
                .setCancelable(false)
                .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        Toast.makeText(MainActivity.this,"Selected Option:
YES",Toast.LENGTH_SHORT).show();
                    }
                })
                .setNegativeButton("No", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        Toast.makeText(MainActivity.this,"Selected Option:
No",Toast.LENGTH_SHORT).show();
                    }
                });
```

Android AlertDialog Example

```
//Creating dialog box
        AlertDialog dialog  = builder.create();
        dialog.show();
      }
   });
  }
}
```

Output of Android AlertDialog Example