



By: AbuBakar Ubaid

# MOBILE APPLICATION DEVELOPMENT

ANDROID

# EVENT HANDLING



By: AbuBakar Ubaid

ANDROID

# Android - Event Handling

- **Event Listeners** – An event listener is an interface in the View class that contains a single callback method. These methods will be called by the Android framework when the View to which the listener has been registered is triggered by user interaction with the item in the UI.
- **Event Listeners Registration** – Event Registration is the process by which an Event Handler gets registered with an Event Listener so that the handler is called when the Event Listener fires the event.
- **Event Handlers** – When an event happens and we have registered an event listener for the event, the event listener calls the Event Handlers, which is the method that actually handles the event.



# onClickListener() and onClick()

```
//---Button view---
```

```
Button btnOpen = (Button)
    findViewById(R.id.btnOpen);
    btnOpen.setOnClickListener(new View.OnClickListener() {

        public void onClick(View v) {
            DisplayToast("You have clicked the Open button");
        }

    });
```



Event Handler	Event Listener & Description
5  onClick()	<b>OnClickListener()</b> This is called when the user either clicks or touches or focuses upon any widget like button, text, image etc. You will use onClick() event handler to handle such event.
onLongClick()	<b>OnLongClickListener()</b> This is called when the user either clicks or touches or focuses upon any widget like button, text, image etc. for one or more seconds. You will use onLongClick() event handler to handle such event.
onFocusChange()	<b>OnFocusChangeListener()</b> This is called when the widget loses its focus ie. user goes away from the view item. You will use onFocusChange() event handler to handle such event.
onKey()	<b>OnFocusChangeListener()</b> This is called when the user is focused on the item and presses or releases a hardware key on the device. You will use onKey() event handler to handle such event.



Event Handler	Event Listener & Description
<code>onTouch()</code>	<b>OnTouchListener()</b> This is called when the user presses the key, releases the key, or any movement gesture on the screen. You will use <code>onTouch()</code> event handler to handle such event.
<code>onMenuItemClick()</code>	<b>OnMenuItemClickListener()</b> This is called when the user selects a menu item. You will use <code>onMenuItemClick()</code> event handler to handle such event.
<code>onCreateContextMenu()</code>	<b>onCreateContextMenuListener()</b> This is called when the context menu is being built(as the result of a sustained "long click")



# LIST VIEW



By: AbuBakar Ubaid

ANDROID

# ListView View

- The ListView displays a list of items in a vertically scrolling list.

```
public class MainActivity extends AppCompatActivity {  
  
    // Array of strings...  
    String[] mobileArray = {"Android", "iPhone", "WindowsMobile", "Blackberry",  
        "WebOS", "Ubuntu", "Windows7", "Max OS X"};  
    ListView listView;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        ArrayAdapter adapter = new ArrayAdapter<String>(this,  
            R.layout.activity_listview, mobileArray);  
  
        listView = (ListView) findViewById(R.id.mobile_list);  
        listView.setAdapter(adapter);  
  
        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
            @Override  
            public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {  
                Toast.makeText(getApplicationContext(), ""+listView.getItemAtPosition(i),  
                    Toast.LENGTH_SHORT).show();  
            }  
        });  
    }  
}
```





## LECTURE – 05 “Event Handling & List View”

THANK YOU 😊

