

# Handling UI Events

# A Quick Review

- Remember, an Activities UI is defined using View and ViewGroup nodes and is represented by a hierarchy tree.
- Question: What is the hierarchy tree for MyAddition?

# Rendering

- What is the purpose of `setContentView ()`?
- How is the screen rendered?

# Event-handling

- Events are created through user interaction
- Events are captured from a View object the user interacts with

Example: When a View such as a Button is touched, the method `onTouchEvent ()` is called on the touched object

# How to handle the event?

- If a Button is touched, how do we do we provide “our” functionality for the Button?
- Answer: We could extend the class and override the method
- Question: What is the down side of this method?

# Nested Interfaces

- The View class contains nested interfaces that are easily definable.
- Of particular interest for our present application is View.OnClickListener
- Let's go to <http://developer.android.com/reference/android/view/V> and look at Nested Classes and then look at View.OnClickListener

# Let's add a listener

- Problem: Let's add a listener for our CLEAR Button.
- First we need a Button variable:

```
private Button mButtonClear;
```

private instance variables are typically preceded with a m. Please use this convention.

# What might a Button look like?

```
class Button
{
    private View.OnClickListener listener;
    public Button ()
    {
        listener = null;
    }
    public void setOnClickListener (View.OnClickListener L)
    {
        listener = L;
    }
    private void handleEvent(Event e)
    {
        paintButton();
        if( listener != null)
        {
            listener.onClick (this);
        } ...
    }
}
```



# Add Clear Button Functionality

1. Add `implements View.OnClickListener` to Activity
2. Add to onCreate

```
mButtonClear = (Button) findViewById  
                (R.id.buttonClear);  
mButtonClear.setOnClickListener (this);
```

# Write the code for OnClick

```
public void onClick (View view)
{
    if (view == mButtonClear)
    {
    }
}
```

# Problem

- You are to write the code that will clear each editText field if the CLEAR Button is pressed. Fill the editText fields and test away!!!