Mobile Application Programming: Android

Custom Controls
Activities

- Apps are composed of activities

- Activities are self-contained tasks made up of one screen-full of information

- Activities start one another and are destroyed commonly

- Apps can use activities belonging to another app
Android Layout

View & ViewGroup

- TextView (Large, Medium, Small)
- Button
- CheckBox
- RadioButton
- CheckedTextView
- Spinner
- Switch

- abc
- Firstname Lastname
- .........
- 1...2...3
- user@domain
- (555) 0100
- Address
- Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
- 12:00am
- 1/1/2011

- GridLayout
- LinearLayout (Vertical)
- LinearLayout (Horizontal)
- RelativeLayout
- FrameLayout
- Include Other Layout
- Fragment
- TableLayout
- TableRow
- Space

- Alarm
- Text field
- Sync Browser
- 5/31/2012 4:58 PM

- Off
- On
- OFF
- ON

- Yes
- Maybe
- No
Custom Controls
3D Color Chooser
Gimbals
The Heist

Custom Controls

public class **ImageView**
extends View

**java.lang.Object**
  ↦ **android.view.View**
    ↦ **android.widget.ImageView**

- Known Direct Subclasses
  - **ImageButton**, **QuickContactBadge**

- Known Indirect Subclasses
  - **ZoomButton**
Creating a Custom Control

- Create subclass of View class
- Override:
  - `onDraw(Canvas c)`
  - `onMeasure(int wMeasure, int hMeasure)`
- Add listener interface and listener property for the interesting events the control generates and call on... methods when events occur
Creating a Custom Control

- Other Considerations:
  - Call `setMinimumWidth` and `setMinimumHeight` to give a suggested constraint for the view’s content’s size.
  - Consider view state persistence with `onSaveInstanceState` (lecture coming)