Mobile Application Programming: Android
UI and Layout
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
- Much more to come...
Informational Views

- TextView
- ImageView
- ProgressBar
- AnalogClock
- DigitalClock
- Chronometer
- VideoView
Input Controls

- Buttons
- Check Boxes
- Toggle Buttons
- Radio Buttons
- Text Fields
Buttons

- Class - `android.widget.Button`
- Principal Event - `onClick`  Principal Property - `title`
- Accepting Event
  - Call `setOnClickListener` with an anonymous class as the argument
  - Define `public void onClick(View v)` in the anonymous class
Check Boxes

- Class - `android.widget.CheckBox`
- Principal Event - `onClick`  Principal Property - `checked`
- Accepting Event
  - Call `setOnCheckedChangeListener` with an anonymous class as the argument
  - Define `onCheckedChanged(...)` in the anonymous class
Toggle Buttons

- Class - `android.widget.ToggleButton`
- Principal Event - `onClick`  Principal Property - `checked`
- Accepting Event
  - Just like CheckBox as both are actually subclasses of `android.widget.CompoundButton`
Radio Buttons

- Principal Event - `onClick`  Principal Property - `checked`
- Creation - Instantiate `RadioButton` then and add to a `RadioGroup`
- Accepting Event
  - Create an anonymous on click listener class as with a CheckBox and store it in a variable
  - Call `setOnCheckedChangeListener` passing variable
  - More simply, you can also set one on the `RadioGroup` instead
Text Fields

- Class - `android.widget.TextView` & `android.widget.EditText`
- Principal Event - `onKey`    Principal Property - `text`
- Accepting Event
  - Call `setOnKeyListener` (defined by the `View` class) with an anonymous class as the argument
  - Define `onKey(....)` in the anonymous class
AWT Layout Managers

http://java.sun.com/docs/books/tutorial/uiswing/layout/visual.html
Android Layout Approach

LinearLayout

- Single-Row Layout
- Horizontal / Vertical
- Dividers
- Baseline Alignment
- Weight
- Gravity
LinearLayout Params

- Weight
  - Distributes extra space across children
  - Defaults to 0 (leave empty)
- Layout Gravity
  - How to position smaller child within larger container
  - View.setGravity different
RelativeLayout

- Children Relative to Each Other or to Parent
- Uses Rules in Params
- Rules Require View ID
- Gravity
RelativeLayout Params

- Above / Below
- Left Of / Right Of
- Left / Right / Top / Bottom
- Align Parent
  - Left / Right / Top / Bottom
- Center in Parent
  - Vertical / Horizontal / Both
ListView

- Lists data provided by an Adapter
- Use ArrayAdapter or a custom class to provide data
- Set OnItemClickListener to react to clicks on rows
GridView

- Lists data provided by an Adapter
- Use ArrayAdapter or a custom class to provide data
- Set OnItemClickListener to react to clicks on rows
- Call setNumColumns to change the grid resolution