Android Notes – 01/22/2019

Android apps are separated into two parts:

Resources and Java Code

Resources:

-libraries, layouts, drawables (pngs, XML to describe shapes, etc), Strings, colors, menus, styles, and more

Java Code:  
-All the logic that runs

-the code you write

-code you get from the framework

Layouts and Views

-every UI “widget” is called a view bc it is derived from the view class

-views can be singular: buttons, checkboxes, etc.

Widget = view = control

Views can be used to layout other views

-called “view groups” or “layouts”

-can have child views

-control != view group or layout

“@” signals a resource

“@+id/name” says: we’re getting a resource with the following id (characters after the slash), and if it doesn’t exist, create it

!!using resources is how we increase uniformity – once we create a resource we can use it repeatedly anywhere in the code that we need to, thus making things consistent!!

Layouts:

-these describe how the screen looks

-you WILL have to play with the XML to get what you want

Linear Layout:

-views are shown in a single direction (horizontally or vertically) on the screen in the order that they appear in the XML

-when there is no more room in the row, the output drops to the next line

-all children of a LinearLayout are stacked one after the other, so a vertical list will only have one child per row

-respects margins and gravity between children

-very obsolete

Relative Layout:

-a view group that displays child views in positions relative to each other or the containing view group

-positions can be specified relative to sibling elements or the parent

-keeps your hierarchy flat

Constraint Layout (Preferred as of 2017)

-allows complex layouts with flat view hierarchy

-similar to relative layout in that all views are laid out with respect to relationships between siblings and the parent layout

-easier to use with Android Studio’s layout editor

-also allows for invisible guidelines that can provide extra control with simplicity