

CSE 340

Winter Quarter 2020

1/22/2020

Professor: Dr. Jennifer Mankoff

Note-taker email: jial8@uw.edu

Link to Google Docs:

https://docs.google.com/document/d/1iSk4EmS2V1SQDRBmG4MR_eoqhAdFAztZWLSd7FeEkqE/edit?usp=sharing

Week 3 Lecture 1

Layout in Android

Side Notes

* Possible exam questions: If I draw this tree hierarchy, what does the screen look like?
What is missing from this diagram?

* **Most missed question in previous quiz: Draw a CircleView at (100,100) with a radius RAD=10** → `canvas.drawCircle(10,10,RAD,paint)`

- the center may technically be at (100,100) in the parent, **but the circle actually needs to start from the upper left square bounding the circle (from the child's perspective), NOT the parent!**

Answer to the right image: **(0,0)**

- **The rectangle doesn't care where it is in the parent.** The rectangle is the origin of itself!

How is position calculated by the toolkit architecture?

- Performs **Depth-First Traversal**
- Toolkit architecture invokes layout
- Containers do all the work
 - Manage resize, size, position of children

You have been asked to create a new `SquareView` object that can draw a square. The user wants to add a square with its top left corner at (150,50) and a width of (20). In `SquareView.onDraw()` you will need to call `canvas.drawRect(left, top, right, bottom, paint)` What values should you use for `left` and `top`?

- Enforce component abstraction (recursive traversal)
- Clip child after drawing is complete

Measuring

- **onMeasure()**: callback invoked by View.measure()
 - For each child... calculate max child width and height
 - `maxWidth = Math.max(..., child.getMeasuredHeight())`
 - `setMeasuredDimension()`
 - Get child's preferred layout → child knows parents' width and height
 - Should maintain aspect ratio!
 - Calculate each child's position in the stack, and its y position based on other images
- Decisions can be made by the child if they know how much space to expect
 - Guidance about size is given by parent view in form of MeasureSpec parameters
 - Ensures every child has width+height value set prior to a layout pass
 - Best toolkits specify:
 - Preferred size
 - Min, max size
 - Parents can ignore the above specifications of Child... but usually tries not to. Child component has to deal if not enough space → not display that part or truncate self... etc
- ImageView does not have a height before Layout is done... What to do?
 - HW2 Layout Part 3 → need to figure out which column to place image in without being unbalanced or look weird
- call **image.measure()** → `image.getMeasuredHeight()`, NOT `image.getHeight()`!
- Note: all images in the example column are at `x = 0`. Only need to change y values!

Drawing Process

- **In-order tree traversal** is done on the hierarchy.
 - Each view is responsible for drawing itself
- Drawing can be costly, only do when need to!
 - Redrawing "damaged" areas → `invalidate()`