



CS260 Intro to Java & Android

11.AndroidAdvUI (Part III)

Winter 2018

Creating TicTacToe for Android

- Thus far we have
 1. Created a TicTacToe board
 2. Captured onTouch events
 3. Hooked up our TicTacToe Java code to the Android code

Creating TicTacToe for Android

- Final steps
 1. Highlighting a rectangle
 2. Writing text centered in a rectangle
 3. Implement the game play

Drawing a Selected Rectangle on Screen Using Pixel Coordinates

- Let's work backward ... at some point we need to draw the selected rectangle in the onDraw
- Here's a hard coded example

```
private Rect mSelectedRectangle = new Rect ();  
private Paint mSelectedRectanglePaint = new Paint ();  
mSelectedRectangle.set (10, 10, 50, 50);  
mSelectedRectanglePaint.setColor (Color.BLUE);  
canvas.drawRect (mSelectedRectangle,  
                 mSelectedRectanglePaint);
```

Set Selected Rectangle on Screen Using Board Coordinates

- The selected rectangle area on the display needs to be set

```
private void setSelectedRectangle (int xCoordinate,
int yCoordinate, Rect rectangle)
{
    rectangle.set ((int) (xCoordinate * mRectangleWidth),
                  (int) (yCoordinate * mRectangleHeight),
                  (int) (xCoordinate * mRectangleWidth +
mRectangleWidth - 1),
                  (int) (yCoordinate * mRectangleHeight +
mRectangleHeight - 1));
}
```

Selecting Rectangle Using Board Coordinates

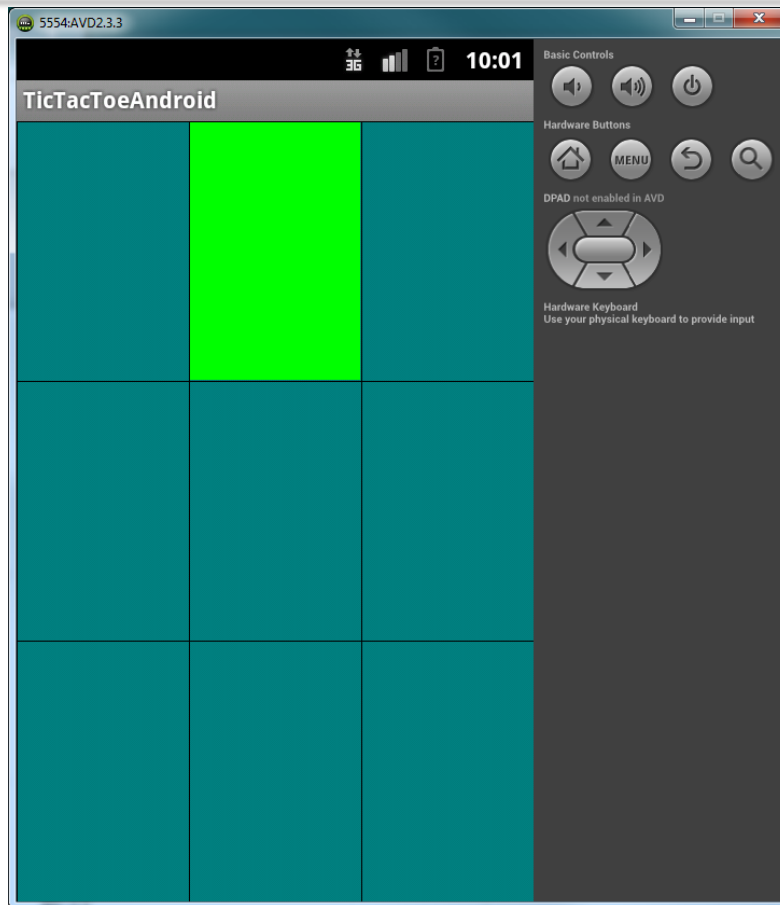
- Select the (x,y) coordinates of the rectangle on the screen

```
private void selectRectangle (int xCoord, int yCoord)
{
    invalidate (mSelectedRectangle); // invalidate old rectangle selection
    mXCoordSelectedRect = Math.min (Math.max(xCoord, 0),
                                    (int) mNUMBER_OF_RECTANGLES - 1);
    mYCoordSelectedRect = Math.min (Math.max(yCoord, 0),
                                    (int) mNUMBER_OF_RECTANGLES - 1);
    setSelectedRectangle (mXCoordSelectedRect, mYCoordSelectedRect,
                          mSelectedRectangle);
    invalidate (mSelectedRectangle); // invalidate new rectangle selection
}
```

Drawing a Selected Rectangle

- In the `onTouchEvent`, make a call to `select`
- Add additional logic so that the screen initially has no selected rectangles. As soon as the user hits a rectangle, highlight the selected rectangle
- See the following slide

Drawing a Selected Rectangle



Painting Text

- Finally, we need to paint some text on the screen. Code is based on [Hello, Android](#) by Ed burnette

```
private Paint mForeground = new Paint  
                        (Paint.ANTI_ALIAS_FLAG);
```

...

```
mForeground.setStyle (Style.FILL);  
mForeground.setTextSize (mRectangleHeight * 0.75f);  
mForeground.setTextScaleX (mRectangleWidth / mRectangleHeight);  
mForeground.setTextAlign (Paint.Align.CENTER);  
FontMetrics fontMetrics = mForeground.getFontMetrics ();  
float xFontCoord = mRectangleWidth / 2;  
float yFontCoord = mRectangleHeight / 2  
                - (fontMetrics.ascent + fontMetrics.descent) / 2;
```

Painting Text

- Drawing text into a rectangle is:

```
canvas.drawText (Integer.toString (i),  
x * mRectangleWidth + xFontCoord,  
y * mRectangleHeight + yFontCoord,  
mForeground);
```

- You are to draw the numbers 1 through 9 in the TicTacToe board tiles going upper-left to lower-right