**SimpleCanvas API**

A SimpleCanvas represents a window on the computer screen where lines and shapes can be drawn.

**Constructors**

* SimpleCanvas(int width, int height)
  + Creates a canvas in a new window on the screen of the given width and height.
* SimpleCanvas(int width, int height, String title)
  + Same as above, but lets you specify the title of the window.

**Instance Methods**

* void drawCircle(int centerX, int centerY, int radius)
  + Draws a circle centered at (centerX, centerY) of the given radius.
* void drawOval(int centerX, int centerY, int radiusX, int radiusY)
  + Draws an oval centered at (centerX, centerY) with the given x-radius and y-radius.
* void drawRectangle(int topLeftX, int topLeftY, int width, int height)
  + Draws a rectangle with the top left corner at (topLeftX, topLeftY) and the given width and height.
* void drawFilledCircle(int centerX, int centerY, int radius)
* void drawFilledOval(int centerX, int centerY, int radiusX, int radiusY)
* void drawFilledRectangle(int topLeftX, int topLeftY, int width, int height)
* void drawLine(int x1, int y1, int x2, int y2)
  + Draws a line from the point (x1, y1) to (x2, y2).
* void drawString(int x, int y, String text)
  + Writes the specified text on the screen at the coordinates (x, y).
* void setLineThickness(int size)
  + Sets the thickness of the lines used for drawing lines and non-filled shapes.
* void drawImage(int x, int y, String filename)
  + Draws an image on the canvas with the top-left corner at (x, y). Supports JPG or PNG (maybe others too).
* void setPenColor(Color c)
  + Sets the color of the "pen" used for drawing lines and shapes.
* void show()
  + Show the canvas window on the screen. Automatically draws all shapes since the last update.

* void hide()
  + Hide the canvas window on the screen (make it invisible).
* void update()
  + Update the canvas drawing to draw everything since the last update.
* int getHeight()
  + Get the height on the canvas in pixels.
* int getWidth()
  + Get the width of the canvas in pixels.
* Color getPixelColor(int x, int y)
  + Get the color of a specific pixel on the canvas.
* void setPixelColor(int x, int y, Color c)
  + Set the color of a specific pixel on the canvas.

**Color API**

The Color class represents a red-green-blue color that can be displayed on the screen.

**Constructors**

* [Color](https://docs.oracle.com/en/java/javase/11/docs/api/java.desktop/java/awt/Color.html" \l "%3Cinit%3E(int,int,int))​(int r, int g, int b)
  + Creates a new RGB color with the specified values. Each value should be between 0 and 255, inclusive.

**Instance Methods**

* int getRed()
  + Returns the red component of this color (0-255).
* int getGreen()
  + Returns the green component of this color (0-255).
* int getBlue()
  + Returns the blue component of this color (0-255).
* Color brighter()
  + Returns a color of the same hue that is brighter than this color.
* Color darker()
  + Returns a color of the same hue that is darker than this color.
* boolean equals(Color otherColor)
  + Returns true if this color is the same color as otherColor.