Colours in Java

When you use a *setColor()* or *setBackgroundColor()*, or any other method to tell Java what color you want to use, you have to put something in the brackets to specify the color. The thing you put in brackets will specify an existing color, or create a new color for Java to use.

When you change the colour, you do not change the colour of something already on the screen. Instead it works as if you are picking up a new coloured pen. Everything you draw from now on will be in the new colour. So (i) set the colour, then (ii) draw something.

You use an existing color object like this:

gc.setColor(Color.red); or

gc.setBackgroundColor(Color.BLUE);

➔ *Do not use "gc.setBackground(Color.BLUE);" as it will not work.*
➔ *Recall (from the previous document) that you have to do* ***c.clear()*** *after you set the background colour.*

Possible color names are:

BLACK, BLUE, CYAN, DARK_GRAY, GRAY, GREEN, LIGHT_GRAY, MAGENTA, ORANGE, PINK, RED, WHITE, AND YELLOW.

You can see what the colours are, etc. by Googling "java color class". Try it now

Or you can create your own color object like this:

c.setColor(new Color(128, 138, 255)); or

c.setBackgroundColor(new Color(0, 100, 0)); or

Color ballColour = new Color(55,66,77); //this is now a colour variable

c.setColor(ballColour);

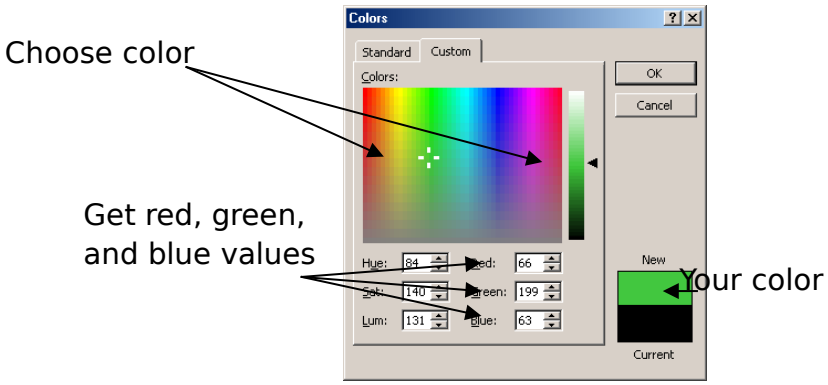
The three numbers specify how much of the colors **red**, **green** and **blue** to mix.
➔ 0 is the minimum and 255 is the maximum amount. ➔
Anything else will cause an error.
new Color(0, 0, 0) is black,
new Color(255, 255, 255) is white,
new Color(255, 0, 0) is bright red *(100% red, zero green, zero blue)*
new Color(100, 0, 0) is dull red, and so on.

How do you find out the red/green/blue values for a good color?

One way to do this is to go into any windows program, then any color tool, select **more colors**, and the **custom** tab. Then choose the color you want, and copy down the red, green, and blue values shown.

Basically, the three primary colours for any lights (TV, computer monitors, stage lighting)
are red, green and blue.

R + B = magenta B + G = cyan R + G = yellow R + G + B = white



Transparency

If you create a new color with a fourth 0-255 value, it will be used as the "alpha" value, which controls how transparent the color is. When you draw in a partially transparent color, instead of covering over other shapes and text on the screen, they will still show through. Try it, it's cool!

c.setColor(new Color(128, 138, 255, 127));

The 127 is the transparency. A value of 127 or 128 is 50%.
A value of 0 is completely transparent (so you won't see anything that you draw).
A value of 255 is completely opaque, as if you had no transparency at all.