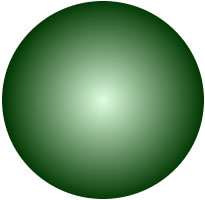
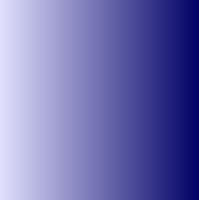
**GUI13 Gradient**

**Directions**

In graphic programming the term gradient is used to describe a gradual blend of color which can be considered as an even gradation from low to high values of color. Below are two examples of gradients.



Circle Gradient

The circle gradient can be created by drawing a series of increasingly smaller circles with each circle having a slightly lighter color than the previous circle.

Write a GUI program that draws a circle filled with a gradient color like the sample shown above.

You will need to use the following Graphics class methods:  
- **setColor**(Color c)  
- **fillOval**(int x, int y, int w, int h)

Copy the following code into your source file.

import java.util.\*;

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

public class GUI13 extends JFrame

{

// constructor

public GUI13()

{

// set frame attributes

setLayout(null);

setSize(500, 500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

getContentPane().setBackground(Color.white);

setVisible(true);

}

public void paint(Graphics page)

{

super.paint(page); // erase background

}

// main method

public static void main(String[] args)

{

GUI13 app = new GUI13(); // run program

}

}

Programming Notes

1. All of the code needs to be in the paint method.
2. Create a for loop that counts from 0 to 255. This loop will control the number of circles that will be drawn. The reason you want a loop that executes 256 times is because that is the range of possible color values.
3. When changing the color of the circle only select one of the RGB values to modify. For example, if I want to draw a red gradient circle my code to create a color might look like the following where the variable r represents the value of red.
4. page.setColor(new Color(r, 0, 0));

As r increases from 0 to 255 the color will change from black to increasingly lighter shades of red.

1. The circle should initially be drawn with the following values: x = 50, y = 50, width = 400, height = 400. This will center the circle in the middle of the screen.
2. To shrink the circle, repeatedly add 1 to x and y and subtract 2 from width and height.

**Source File**

GUI13.java