Pictures and For-each Loops

Pixel Method	Description
`int getRed()`	Get the red intensity (an integer from 0-255)
`int getGreen()`	Get the green intensity (an integer from 0-255)
`int getBlue()`	Get the blue intensity (an integer from 0-255)
`void setRed(int)`	Set the red intensity to a value from 0-255
`void setGreen(int)`	Set the green intensity to a value from 0-255
`void setBlue(int)`	Set the blue intensity to a value from 0-255
`int getX()`	Get the x coordinate where this pixel is located in the image
`int getY()`	Get the y coordinate where this pixel is located in the image
`void setColor(int red, int green, int blue)`	Set all three color values at once

0,0,0 = black

255,255,255 = white

For-Each Loops

```java
for (<datatype> <variable-name> : <collection>)
{
    // body of for-each loops
}

// pixel example
for (Pixel pix : image.getPixels())
{
    pix.setRed(255);
}
```

Picture Method	Description
`new Picture(String)`	Use this constructor to create a `Picture` from an image file by providing the file name in double-quotes
`new Picture(int width, int height)`	Use this constructor to create a new, blank `Picture` with the specified dimensions
`int getWidth()`	Get the width of this image, in pixels
`int getHeight()`	Get the height of this image, in pixels
`Pixel getPixel(int x, int y)`	Get the pixel at the specified location
`Pixel[] getPixels()`	Get all the pixels in the image in a form suitable for use in a for-each loop

`void show()`	Show this picture on the screen
`void repaint()`	Update the on-screen image shown using `show()`
`void hide()`	Hide the image shown on the screen using `show()`
`void explore()`	Show the image using an image explorer view that allows you to inspect the color of any pixel in the image
`void reload()`	If this image was loaded from the file, throw away any changes made to the image and reload it fresh from the original file to restore it to its original appearance