Hands-on Experiment # 9-2 : Worksheet

Section_______________ Date_____________________________________

No more than 3 students per one submission of this worksheet.

Student ID ____________________________________ Name____

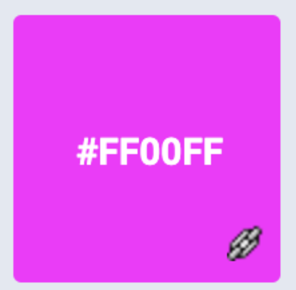
Student ID ____________________________________ Name____

Student ID ____________________________________ Name____

Part A: Understanding Given Resources/Backgrounds

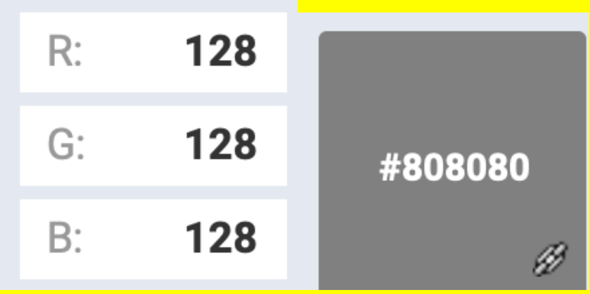
1) Get yourself familiar with the RGB color model. Play around with the color picker on http://www.colorpicker.com/ and answer the following questions.

   a. What color is it that has the maximal value in R, the maximal value in B, and 0 in G? Capture the picture of the color and post it here.



#FF00FF

   b. What are the requirements on the RGB values for all shades of gray?



R: 128

G: 128

B: 128

#808080

2) Read the API specification of the class *Java101ImageUtil* in *L09-2.pdf*.

   a. How many static methods are there in the class?

6 static methods

   b. How many overloaded methods are there in the class?

3 overloaded methods

   c. Write the "method signatures" of all the overloaded methods. (** Write only the signature) showViewer(int[][][] , String);

showViewer(int[][][] , int[][][] , String);

showViewer(int[][][][]);

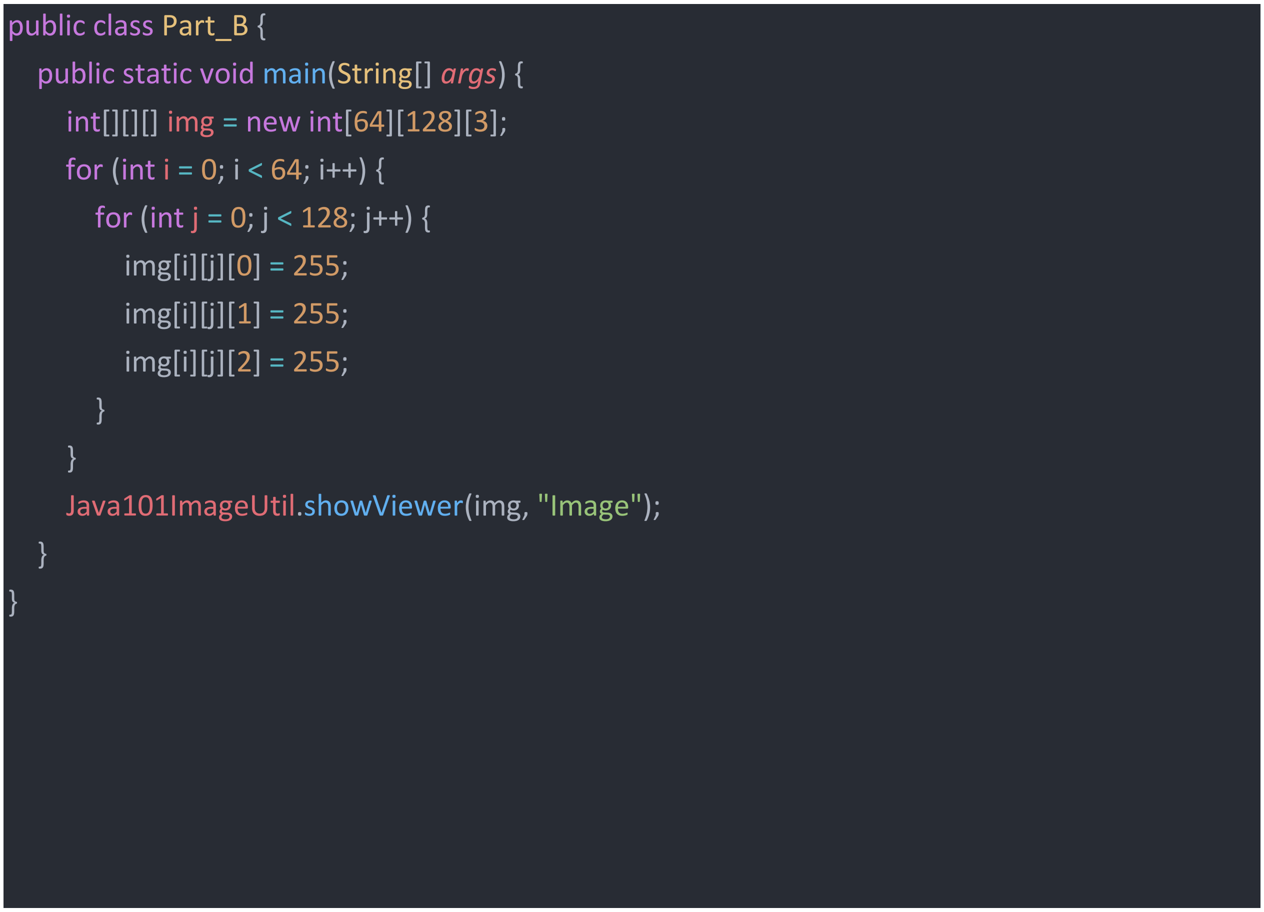
3) Read the source code of *Java101ImageUtilExample.java* and try executing the program. Briefly explain what the program does. (** It is recommended NOT TO open big images. The program was not optimized in any ways. Try the program on some images with a few hundreds of pixels in their width/height)

The purpose of this program is to open a picture file in the computer and show on the screen in three different ways. Firstly, the program just shows the picture itself. Secondly, it shows the flipped version of the image on the screen. Lastly, it shows three different images which are the original image, its flipped version, and the red image.

Part B: Creating RGB arrays for Desired Images

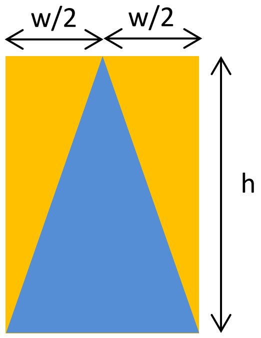
1) Write a program performing the following steps.
   a. Create a 3-D array of `int` that when used with `showViewer(int [ ][ ][ ],String)`, the program shows a 64-pixel x 128-pixel all-white image.
   b. Show the image with `showViewer(int [ ][ ][ ],String)`

   List your source code here.



```
public class Part_B {
  public static void main(String[] args) {
    int[][][] img = new int[64][128][3];
    for (int i = 0; i < 64; i++) {
      for (int j = 0; j < 128; j++) {
        img[i][j][0] = 255;
        img[i][j][1] = 255;
        img[i][j][2] = 255;
      }
    }
    Java101ImageUtil.showViewer(img, "Image");
  }
}
```

2) Write another program performing the following steps.
   a. Ask the user to input the value of w and h, which are integers in the range of 100 to 200.
   b. Show an image of a triangle as shown in the figure below. Use the colors of your choice.



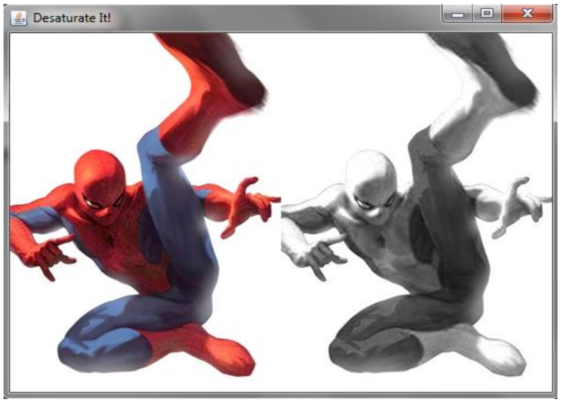
List your source code here.

```
Scanner kb = new Scanner(System.in);
    int w = kb.nextInt();
    int h = kb.nextInt();
    int[][][] img = new int[w][h][3];
    for (int i = 0; i < w; i++) {
      for (int j = 0; j < h; j++) {
        if (j > -h + 4 * i && j > h - 4 * i) {
          img[i][j][0] = 255;
          img[i][j][1] = 128;
          img[i][j][2] = 0;
        } else {
          img[i][j][0] = 0;
          img[i][j][1] = 128;
          img[i][j][2] = 255;
        }
      }
    }
    Java101ImageUtil.showViewer(img, "Image");
```

Part C: Image Manipulation

Modify *DesaturateIt.java* to obtain a Java program performing the following steps.

1) Ask the user to select a gif or a jpg file.
2) Show the original image and its “desaturated” (grayscale) version using `showViewer()`.



Explain how the grayscale values are computed.

The greyscale is set by finding the average of the integers of red, green, and blue, then initialize the data to be that value.

List your source code here.

```
public class DesaturateIt {
  public static void main(String[] args) {
    int[][][] rgb = Java101ImageUtil.getRGBArrayFromFile();
    if (rgb == null) {
      return;
    }
    int[][][] gray = desaturate(rgb);
    Java101ImageUtil.showViewer(rgb, gray, "Desaturate It!");
  }

  public static int[][][] desaturate(int[][][] rgb) {
    int w = rgb.length;
    int h = rgb[0].length;
    int[][][] img = new int[w][h][3];
    for (int i = 0; i < w; i++) {

      for (int j = 0; j < h; j++) {

        for (int k = 0; k < 3; k++) {

          img[i][j][k] = (rgb[i][j][0] + rgb[i][j][1] + rgb[i][j][2]) / 3;

        }

      }

    }

    return img;

  }

}
```

Submit this worksheet (by only one member of the group) via http://www.myCourseVille.com (Assignments > Hands-on Experiment # 9-2) **within the day after your lecture.**