

Java Bridges Lab: The ColorGrid Class and Creating Patterns

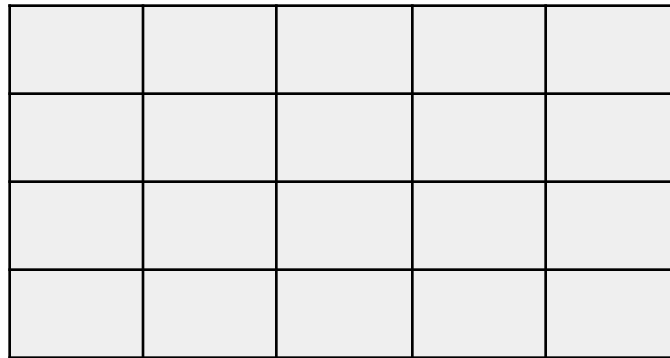
Introduction

Here's a picture of an example **ColorGrid** object.

How many rows does it have? _____

How many columns does it have? _____

Label each row number and each column number, remembering that numbering starts with 0.



For this project, you will write seven methods, each method creating a different pattern. As you work on the problems, remember how you previously used nested loops in order to create patterns. You will apply that knowledge to these new problems.

For your review, here's some **pseudocode** for **iterating** through a grid (or 2D array).

```
for row 0 to the last row of the grid
{
    for column 0 to the last column of the grid
    {
        do things to cells , perhaps based on conditions
    }
}
```

Understanding Bridges Classes

In order to be able to set up our loops and "do things to cells", we should understand the Bridges **ColorGrid** and **Grid** classes. Check out the APIs for these two classes ([ColorGrid API](#), [Grid API](#)). Notice that **ColorGrid** *inherits from* **Grid**.

ColorGrid is the _____ class and **Grid** is the parent (or super) class.

From the ColorGrid API, answer the following:

What method will you use to get the number of rows? _____

What method will you use to get the number of columns? _____

From the *Grid API*, answer the following:

What method will you use to change the color of a cell? _____

Programming Assignment

1. Access the starter Pattern.java class [here](#).
OR
create a new Java class called Pattern.java. Then copy and paste the starter code below into your class.
2. Solve the seven problems detailed within the starter code comments

```
import bridges.connect.Bridges;
import bridges.validation.RateLimitException;

import java.io.IOException;

import bridges.base.Color;
import bridges.base.ColorGrid;

public class Pattern
{
    public static void main(String[] args) throws RateLimitException, IOException
    {
        // instantiate a Bridges object with your:
        // assignment number
        // your username,
        // and your API key
        Bridges bridges = new Bridges(assignment number, "username", "APIkey");

        // instantiate a ColorGrid object with 15 rows and 15 columns
        ColorGrid grid = new ColorGrid(15, 15);

        // SAMPLE PROBLEM: draw sample pattern (two meager beady eyes)
        drawSample(grid, bridges);

        // Continue on, writing methods for each
        // PROBLEM 1: Two colors as parameters. Make every other row a different color.

        // PROBLEM 2: Two colors as parameters. Make every other column a different color.

        // PROBLEM 3: Two colors as parameters. Create a checkerboard pattern

        // PROBLEM 4: Two colors as parameters. Make all the border cells one color and all the interior cells another color

        // PROBLEM 5: Two colors as parameters. Make an "X" design

        // PROBLEM 6: Two colors as parameters. Along the Slope = -1 diagonal, split the grid in half, one color each

        // PROBLEM 7: Five colors as parameters. Show the 4-quadrants in a different color for each. If the ColorGrid has
```

```

        // an odd number of rows and an odd number of columns, show the x and y axes in yet
        // another color. Otherwise, the four quadrants will show and the x and y axes will merely
        // be suggested by the color changes

    } // end main method

    // SAMPLE METHOD (use this as a template)
    public static void drawSample(ColorGrid grid, Bridges bridges) throws RateLimitException, IOException
    {
        //Display a two-eyeballs pattern to the grid (two meager beady eyes -- how lame)
        // first eyeball
        grid.set(5, 6, new Color("yellow"));
        // second eyeball
        grid.set(5, 9, new Color("yellow"));
        // call to the Bridges object called 'bridges' -- set the data structure to the Grid type
        bridges.setDataStructure(grid);
        // call to the Bridges object called 'bridges' -- show the ColorGrid object called 'grid'
        bridges.visualize();
    } // end drawSample

} // end class

```

Note:

Before trying any of the problems, make sure that your `Pattern.java` class works well as is. That means, you can compile it, and that when you run it, your visualization gives you a picture of two yellow beady eyes looking right back atcha.

Then, go ahead and try problem #1 (draw stripes, switch colors every other row)

Each time you want to test a method, comment out any other method calls that you have in the main method. Thus, before you try your `drawStripes()` method, comment out your main method call to `drawSample()`.