**Workout – 2020-09-30 – Adapter & Façade Patterns Name: \_\_\_Sydney O’Dell\_\_\_\_\_\_**

Instructions: **This Workout is due at midnight tonight!** When you are finished with this workout, place the java files (Client, MyCanvas, MyNewCanvas, MyNewCanvasAdapter) and answers to the questions below in the directory. Then, zip up your workout directory and submit it to ASULearn. Make sure your name appears at the top of this document and in a comment atop the Client and MyNewCanvasAdapter classes.

# An Adapter Coding Problem

You have a drawing program with millions of lines of code that for years has been using a vendor-provided MyCanvas class with the following methods to draw shapes:

* clear();
* setLineColor(int rgb);
* setFillColor(int rgb);
* drawSquare(int xPosition, int yPosition, int length);  
   // x and y give the top-left corner
* drawRectangle(int xPosition, int yPosition, int topLength, int sideLength);   
  // x and y give the top-left corner
* drawTriangle (int[]xPosition, int[] yPosition)  
  // each array has 3 coordinate points for the 3 corners of the Triangle
* drawRightTriangle(int xPosition, int yPosition, int verticalLength, int horizontalLength);   
  //The first two parameters are the x/y coordinate is the coordinates of the right angle point. verticalLength and horizontalLength give the lengths of the two legs. verticalLength and horizontalLength can be positive or negative to give any orientation to the Right Triangle
* drawLine (int xStart, int yStart, int xEnd, int yEnd);

Your company was in the process of developing a new version of the application that used a new vendor-provided class. However, your project has been delayed by one year and the vendor is discontinuing support for the old class. You must quickly make the old code fit MyNewCanvas class. The application is about one million lines of code but I'm only showing you 9 sample lines here. ☺

MyNewCanvas has the following methods:

* clear();
* drawShape(int[] xPosition, int[] yPosition, int sides, int lineColor, int fillColor);   
  /\* The xPosition and yPosition arrays have the x and y coordinates for each  
   “corner” of the shape. They are “parallel arrays”.  
   sides gives the number of sides to the shape (1=a point, 2=a line, 3=triangle, 4=square, etc.  
  \*/

Your job is to develop an adapter that can be used by the drawing program to produce drawings using the new canvas class. Code to begin this exercise is in a zip file on ASULearn.

# Here are your specific tasks:

I started the process by making an interface named MyCanvas. It contains all the methods implemented in the old MyCanvas class. I renamed the old canvas to MyOldCanvas. I also started the canvas adapter named MyCanvasAdapter.

Before getting started on the new code, answer the following questions:

1. Is it possible that our new applications could be less efficient at runtime? Explain
   1. Depending on the size of the request, it possibly could. If we called many, and I mean MANY, draw requests it could potentially be less efficient at runtime because we have to go through the adapter.
2. What class will be the Adapter? The Adaptee?
   1. MyCanvasAdapter, MyNewCanvas
3. What is the company saving (in terms of work) by using this pattern?
   1. It is saved from having to change millions of lines of code to turn it into a “newer version”.

I started the MyCanvasAdapter. Answer the following questions about it:

1. In one sentence, state the basic responsibility of the MyCanvasAdapter class.
   1. Has methods for creation of objects that can be called inside the Client class.
2. Why did I need to use two fields, lineColor and fillColor? (Ask me if you need help.)
   1. LineColor and fillColor allow us to display the object when called. If we don’t declare color and fill it won’t be able to be seen when displayed.
3. Copy the code for drawSquare method. Line by line, describe what the method does.

public void drawSquare(int xPosition, int yPosition, int length) {

//takes as input an int for xPosition, int for yPosition, and an int for length

int[] xCoords = {xPosition, xPosition + length, xPosition + length, xPosition};

//declares an int array for X-Coordinates. There are 4 points to a square, so each point calculates all the x-coordinates using the xCoordinate and length.

int[] yCoords = {yPosition, yPosition, yPosition + length, yPosition + length};

//declares an int array for all 4 of the Y-Coordinates, using the yPosition and length declared.

canvas.drawShape(xCoords, yCoords, 4, lineColor, fillColor);

//calls drawShape and takes as input the xCoords array, yCoords arrsy, number of lines needed to create the shape, aswell as the linecolor and fillcolor so we can actually see the shape.

}

Now, change the Client's main so that it now makes calls to a MyCanvasAdapter object. To instantiate MyCanvasAdaptor you will need to pass its constructor a reference to a MyNewCanvas. You will have no more use for the MyOldCanvas object. Now test the client (only the rectangles and squares will be drawn).

Now, implement (in this order): drawTriangle, drawLine, and drawRightTriangle.

Mission accomplished. You just saved your company one million dollars. You will get a one thousand dollar bonus.