Name: Jack Shenfield
Course Name: Advanced System Analysis and Software Design
Course Code: ENSF 614
Assignment Number: Lab 06
Submission Date: 04/11/2025

Exercise A

Source code:

```
/* ENSF 614 - Lab 6 Exercise A
 * File Name: DemoDecoratorPattern.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.Font;
import java.awt.Graphics;

import javax.swing.JFrame;
import javax.swing.JPanel;

public class DemoDecoratorPattern extends JPanel {
	private static final long serialVersionUID = 1L; // recommended solution from eclipse error message I was getting.
	Component t;

	public DemoDecoratorPattern() {
		t = new Text("Hello World", 60, 80);
	}

	public void paintComponent(Graphics g) {
		int fontSize = 10;
```

```
                g.setFont(new Font("TimesRoman", Font.PLAIN, fontSize));

                // Now lets decorate t with BorderDecorator: x = 30, y = 30, width = 100,
and
                // height 100
                t = new BorderDecorator(t, 30, 30, 100, 100);

                // Now lets add a ColouredFrameDecorator with x = 25, y = 25, width =
110,
                // height = 110,
                // and thickness = 10.
                t = new ColourFrameDecorator(t, 25, 25, 110, 110, 10);

                // Now lets draw the product on the screen
                t.draw(g);
        }

        public static void main(String[] args) {
                DemoDecoratorPattern panel = new DemoDecoratorPattern();
                JFrame frame = new JFrame("Learning Decorator Pattern");
                frame.getContentPane().add(panel);
                frame.setSize(400, 400);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.setLocationRelativeTo(null);
                frame.setVisible(true);
        }
}
```

```java
/* ENSF 614 - Lab 5 Exercise A
 * File Name: Component.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.Graphics;

public interface Component {

	void draw(Graphics g);
}
```

```java
/* ENSF 614 - Lab 5 Exercise A
 * File Name: Text.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.Color;
import java.awt.Graphics;

public class Text implements Component {
```

```
        int x, y;
        String text;

        public Text(String text, int x, int y) {
                this.x = x;
                this.y = y;
                this.text = text;
        }

        public void draw(Graphics g) {
                g.setColor(Color.BLACK);
                g.drawString(text, x, y);
        }

}
```

```
/* ENSF 614 - Lab 5 Exercise A
 * File Name: Decorator.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.Graphics;

public abstract class Decorator implements Component {
```

```
        Component cmp;
        int x, y, width;
        public int height;

        public Decorator(Component cmp, int x, int y, int width, int height) {
                this.cmp = cmp;
                this.x = x;
                this.y = y;
                this.width = width;
                this.height = height;
        }

        public void draw(Graphics g) {
                cmp.draw(g);
        }

}
```

```
/* ENSF 614 - Lab 5 Exercise A
 * File Name: BorderDecorator.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.BasicStroke;
import java.awt.Color;
```

```
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.Stroke;

public class BorderDecorator extends Decorator {

	public BorderDecorator(Component cmp, int x, int y, int width, int height) {
		super(cmp, x, y, width, height);
	}

	public void draw(Graphics g) {
		super.draw(g);

		Graphics2D g2d = (Graphics2D) g;

		// Save the old stroke and color so we can restore them later
		Stroke oldStroke = g2d.getStroke();
		Color oldColor = g2d.getColor();

		// Set new stroke and color
		// dashed line here
		g2d.setStroke(new BasicStroke(2f, BasicStroke.CAP_BUTT,
BasicStroke.JOIN_BEVEL, 0, new float[] { 5 }, 0));
		g2d.setColor(Color.BLACK);

		// draw rectangle
		g2d.drawRect(x, y, width, height);
```

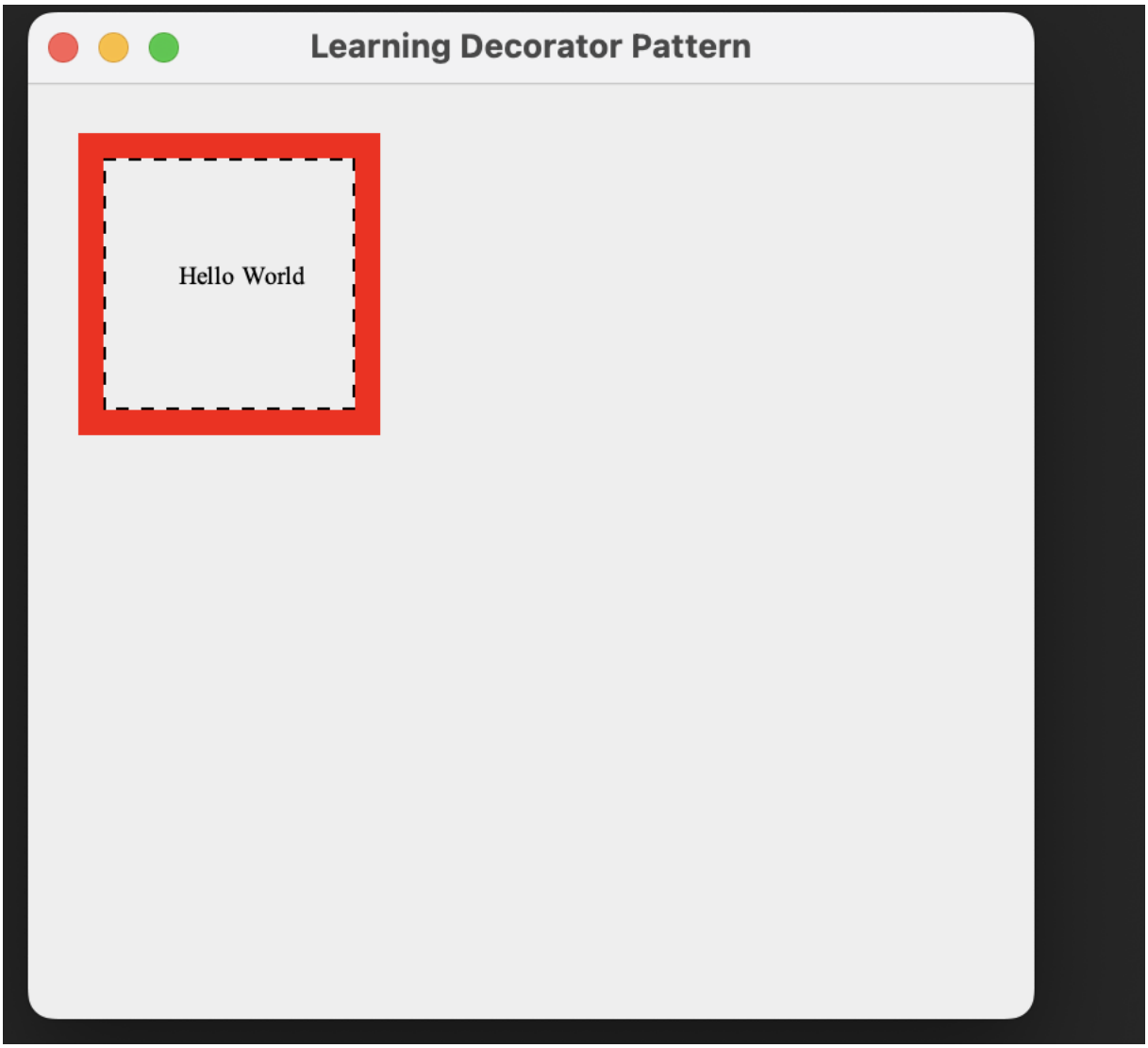
```
                // reset stroke and color
                g2d.setStroke(oldStroke);
                g2d.setColor(oldColor);
        }

}
```

```
/* ENSF 614 - Lab 5 Exercise A
 * File Name: ColourFrameDecorator.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 *
 */

import java.awt.BasicStroke;
import java.awt.Color;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.Stroke;

public class ColourFrameDecorator extends Decorator {

        float thickness;
```

```
        public ColourFrameDecorator(Component cmp, int x, int y, int width, int height,
float thickness) {
                super(cmp, x, y, width, height);
                this.thickness = thickness; // had to add this as my constructor was
missing the 5th integer argument
        }

        // example method given, slightly modified for my use
        public void draw(Graphics g) {
                super.draw(g);
                Graphics2D g2d = (Graphics2D) g;

                // save old stroke and colour
                Stroke oldStroke = g2d.getStroke();
                Color oldColor = g2d.getColor();

                // set stroke & colour
                // plain coloured line with thickness
                g2d.setStroke(new BasicStroke(thickness));
                g2d.setColor(Color.RED); // red as per assignment requirements

                g2d.drawRect(x, y, width, height);

                // Restore old stroke and color
                g2d.setStroke(oldStroke);
                g2d.setColor(oldColor);
        }
```

```
}
```

Output:



Exercise B:

Source code (Only added/modified parts):

Output:

Simply copy pasted the given code:

```java
/* ENSF 614 - Lab 6 Exercise A/B
 * File Name: DemoDecoratorPattern.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 */

import java.awt.Font;
import java.awt.Graphics;

import javax.swing.JFrame;
import javax.swing.JPanel;

public class DemoDecoratorPattern extends JPanel {
	private static final long serialVersionUID = 1L; // recommended solution from eclipse error message I was getting.
	Component t;

	public DemoDecoratorPattern() {
		t = new Text("Hello World", 60, 80);
	}

	public void paintComponent(Graphics g) {
		int fontSize = 10;
		g.setFont(new Font("TimesRoman", Font.PLAIN, fontSize));
		// GlassFrameDecorator info: x = 25, y = 25, width = 110, and height = 110
```

```
                t = new ColourGlassDecorator(
                                new ColourFrameDecorator(new BorderDecorator(t, 30, 30,
100, 100), 25, 25, 110, 110, 10), 25, 25, 110,
                                110);
                t.draw(g);
        }

        public static void main(String[] args) {
                DemoDecoratorPattern panel = new DemoDecoratorPattern();
                JFrame frame = new JFrame("Learning Decorator Pattern");
                frame.getContentPane().add(panel);
                frame.setSize(400, 400);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.setLocationRelativeTo(null);
                frame.setVisible(true);
        }
}
```

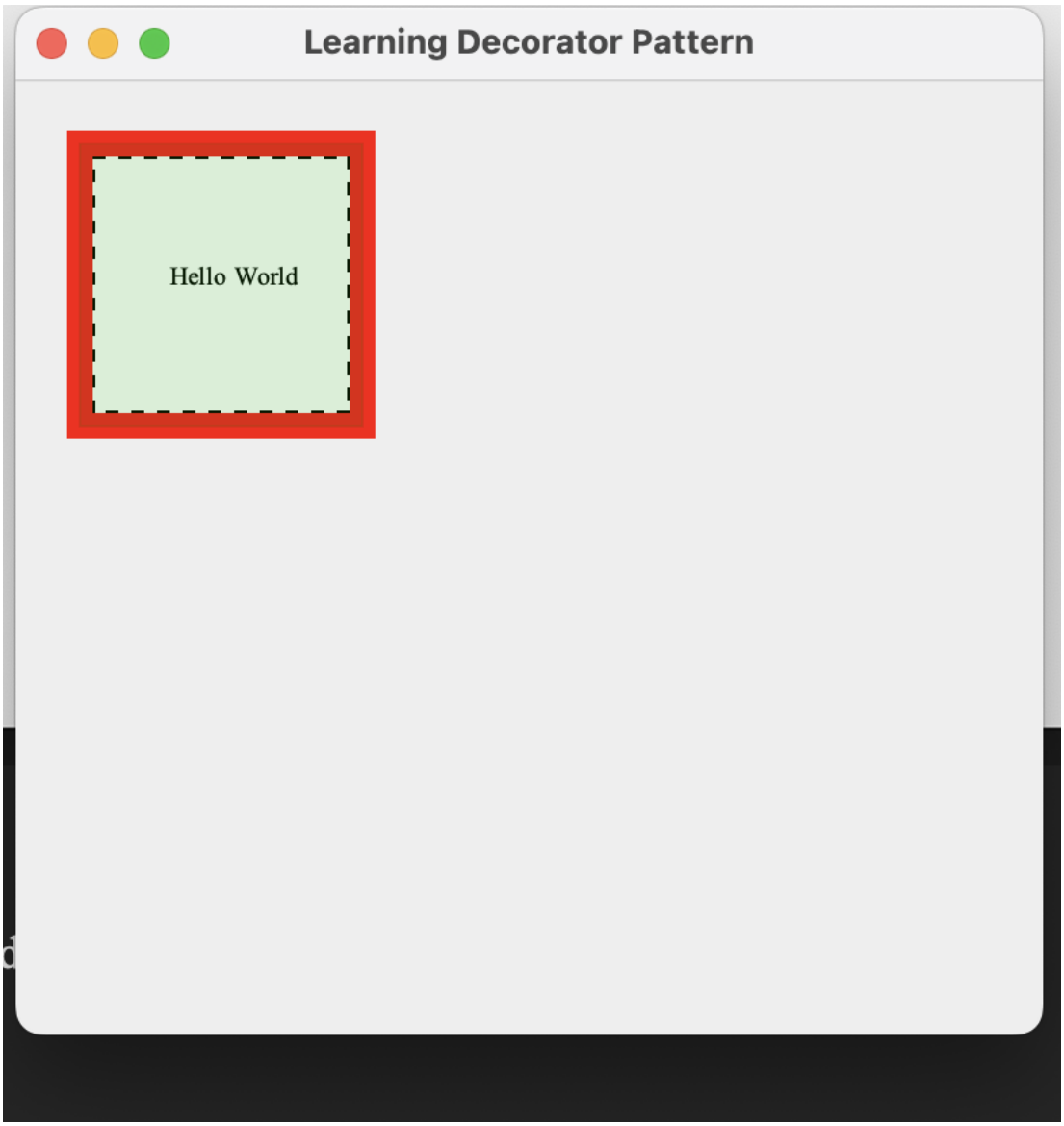
And created my new class, nearly identical to the other decorators. Just added the translucent rectangle logic given:

```
/* ENSF 614 - Lab 5 Exercise B
 * File Name: ColourFrameDecorator.java
 * M. Moussavi, October 2024
 * Lab Section: B01
 * Completed by: Jack Shenfield
 * Submission Date: November 4th, 2025
 *
 */
```

```
import java.awt.AlphaComposite;
import java.awt.Color;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.Stroke;

public class ColourGlassDecorator extends Decorator {

	public ColourGlassDecorator(Component cmp, int x, int y, int width, int height) {
		super(cmp, x, y, width, height);
	}

	// example method given, slightly modified for my use
	public void draw(Graphics g) {
		super.draw(g);
		Graphics2D g2d = (Graphics2D) g;

		// save old stroke and colour
		Stroke oldStroke = g2d.getStroke();
		Color oldColor = g2d.getColor();

		g2d.setColor(Color.GREEN);

	g2d.setComposite(AlphaComposite.getInstance(AlphaComposite.SRC_OVER, 1
* 0.1f));
		g2d.fillRect(25, 25, 110, 110);
```

```
            g2d.drawRect(x, y, width, height);

            // Restore old stroke and color
            g2d.setStroke(oldStroke);
            g2d.setColor(oldColor);
        }

}
```

Output:


Learning Decorator Pattern
Hello World