Chapter 9, #21 – Image manipulation.

```
package guiprogramming;
import java.awt.*;
import javax.swing.*;

public class PhotoOpDrawingPanelHomeWork extends JPanel {
	private Image bimage;
	private double angle = 0.0; // rotate image by this angle
	private float scaleValue = 1.0f; // resize image by this amount

	// fields representing horizontal and vertical shear
	private float horizontalShear = 0.0f;
	private float verticalShear = 0.0f;

	// fields representing horizontal and vertical shift
	private int horizontalShift = 0;
	private int verticalShift = 0;


	public PhotoOpDrawingPanelHomeWork() {
		bimage =  new javax.swing.ImageIcon("image/squirrelMonkey.jpg").getImage();
	}

	public void paintComponent(Graphics g) {
		super.paintComponent(g);
		Graphics2D     g2 = (Graphics2D) g;

		// rotate image
		g2.rotate(angle, 250, 250);

		// scale image
		g2.scale(scaleValue, scaleValue);

		// shear image
		g2.shear(horizontalShear, verticalShear);

		// shift image
		g2.drawImage(bimage, horizontalShift, verticalShift, null);
	}

	public void moveImageUp(int x) {
		horizontalShift -= x;
		repaint();
	}

	public void moveImageDown(int y) {
		verticalShift -= y;
		repaint();
	}

	public void moveImageRight(int x) {
		horizontalShift += x;
		repaint();
	}

	public void moveImageLeft(int y) {
		verticalShift = y;
		repaint();
	}

	public void rotateImage(double a) {
		angle += a;
		repaint();
	}

	public void scaleImage(float s) {
		scaleValue *= s;
		repaint();
	}

	// method to shear an image by the given amounts
	public void shearImage(float hshear, float vshear) {
		horizontalShear = hshear;
		verticalShear = vshear;
		repaint(); // calls the paintComponent method of this class
	}
```

```
	public void loadImage(Image i) {
		bimage = i;
		repaint();
	}

}

--- --- --- --- --- ---

package guiprogramming;
import javax.swing.*;
import java.awt.*;
import javax.swing.border.*;
import java.awt.event.*;
import java.util.Random;
import javax.swing.filechooser.FileNameExtensionFilter;
import java.io.File;


public class PhotoOpHomeWork {
	private JFrame window;
	private JPanel topPanel, groupPanel;
	private PhotoOpDrawingPanelHomeWork drawingPanel;
	private JPanel titlePanel;
	private JLabel titleLabel;
	private JFormattedTextField scaleField;
	private JCheckBox shearBox1;
	private JCheckBox shearBox2;
	private JFileChooser chooser;

	public PhotoOpHomeWork() {
		// create the window
		window = new JFrame();

		// create the panels
		createPanels();

		// panel to hold photo's title
		titlePanel = createTitlePanel();

		// create a label
		titleLabel = new JLabel("squirrelMonkey.jpg");
		titlePanel.add(titleLabel);

		// create shift panel
		groupPanel.add(createShiftPanel());

		// create rotate panel
		groupPanel.add(createRotatePanel());

		// create scale panel
		groupPanel.add(createScalePanel());

		// create shear panel
		groupPanel.add(createShearPanel());

		// create menu bar
		window.setJMenuBar(createMenuBar());

		// set topPanel as the content pane of this window
		window.setContentPane(topPanel);

		// when window is closed, terminate the program as well
		window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

		// set window size
		window.setSize(600, 600);

		// set window title
		window.setTitle("Photo Op");

		// make window visible
		window.setVisible(true);
	}

	public void createPanels() {
		topPanel = new JPanel(new BorderLayout());
		drawingPanel = new PhotoOpDrawingPanelHomeWork();
```

```
        groupPanel = new JPanel();

        // add drawingPanel to topPanel
        topPanel.add(drawingPanel, BorderLayout.CENTER);
        topPanel.add(groupPanel, BorderLayout.EAST);

        groupPanel.setLayout(new BoxLayout(groupPanel, BoxLayout.Y_AXIS));
        groupPanel.setBorder(BorderFactory.createEtchedBorder(EtchedBorder.RAISED));
        groupPanel.setBackground(Color.lightGray);
        topPanel.setBackground(Color.lightGray);
}

public JPanel createTitlePanel() {
        JPanel titlePanel = new JPanel();
        topPanel.add(titlePanel, BorderLayout.NORTH);
        titlePanel.setBorder(BorderFactory.createEtchedBorder(EtchedBorder.RAISED));
        titlePanel.setBackground(Color.lightGray);
        return titlePanel;
}

public JPanel createRotatePanel() {
        JPanel rotatePanel = new JPanel();

        // create two buttons to rotate image
        JButton rotateButton1 = new JButton(new ImageIcon("image/leftButton.jpg"));
        JButton rotateButton2 = new JButton(new ImageIcon("image/rightButton.jpg"));
        rotatePanel.add(rotateButton1);
        rotatePanel.add(rotateButton2);

        rotateButton1.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        // rotate image to the left by 45 degrees
                        drawingPanel.rotateImage(-Math.PI/4);
                }
        });

        rotateButton2.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        // rotate image to the right by 45 degrees
                        drawingPanel.rotateImage(Math.PI/4);
                }
        });
        rotateButton1.setToolTipText("Rotate left by 45 degrees");
        rotateButton2.setToolTipText("Rotate right by 45 degrees");
        rotatePanel.setMaximumSize(new Dimension(200, 60));
        rotatePanel.setBackground(Color.lightGray);
        return rotatePanel;
}

public JPanel createShiftPanel() {
        JPanel shiftPanel = new JPanel();

        // create four buttons to rotate image
        JButton shiftButton1 = new JButton(new ImageIcon("image/leftButton.png"));
        JButton shiftButton2 = new JButton(new ImageIcon("image/upButton.png"));
        JButton shiftButton3 = new JButton(new ImageIcon("image/rightButton.png"));
        JButton shiftButton4 = new JButton(new ImageIcon("image/downButton.png"));
        shiftPanel.add(shiftButton1);
        shiftPanel.add(shiftButton2);
        shiftPanel.add(shiftButton3);
        shiftPanel.add(shiftButton4);

        shiftButton1.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        drawingPanel.moveImageUp(10);
                }
        });

        shiftButton2.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        drawingPanel.moveImageDown(10);
                }
        });

        shiftButton3.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        drawingPanel.moveImageRight(10);
```

```
                }
        });

        shiftButton4.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        drawingPanel.moveImageLeft(10);
                }
        });

        shiftButton1.setToolTipText("Shift left by 10 pixels");
        shiftButton2.setToolTipText("Shift right by 10 pixels");
        shiftButton3.setToolTipText("Shift up by 10 pixels");
        shiftButton4.setToolTipText("Shift down by 10 pixels");
        shiftPanel.setMaximumSize(new Dimension(200, 120));
        shiftPanel.setBackground(Color.lightGray);
        return shiftPanel;
}

public JPanel createScalePanel() {
        JPanel scalePanel = new JPanel();
        JLabel scaleLabel = new JLabel("Scale:");

        // create a formatted text field called scaleField
        scaleField = new JFormattedTextField(new Float(100));
        scaleField.setColumns(3);
        JLabel percentLabel = new JLabel("%");
        scalePanel.add(scaleLabel);
        scalePanel.add(scaleField);
        scalePanel.add(percentLabel);

        // actionListener for scaleField
        scaleField.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        // scale the image by the value in scaleField
                        float scale = (Float) scaleField.getValue()/100.0f;
                        drawingPanel.scaleImage(scale);
                }
        });
        scalePanel.setMaximumSize(new Dimension(200, 60));
        scalePanel.setBackground(Color.lightGray);
        return scalePanel;
}

public JPanel createShearPanel() {
        JPanel shearPanel = new JPanel();
        shearPanel.setLayout(new BoxLayout(shearPanel, BoxLayout.Y_AXIS));

        // create two panels for holding each check box.
        JPanel shearPanel1 = new JPanel(new FlowLayout(FlowLayout.LEFT, 6, 6));
        JPanel shearPanel2 = new JPanel(new FlowLayout(FlowLayout.LEFT, 6, 6));
        shearPanel.add(shearPanel1);
        shearPanel.add(shearPanel2);
        JLabel shearLabel = new JLabel("Shear:");

        // create check box for horizontal shear
        shearBox1 = new JCheckBox("Horizontal");
        shearPanel1.add(shearLabel);
        shearPanel1.add(shearBox1);

        // create check box for vertical shear
        shearBox2 = new JCheckBox("Vertical");
        shearPanel2.add(Box.createHorizontalStrut(37));
        shearPanel2.add(shearBox2);

        // add event handlers to the check boxes
        shearBox1.addItemListener(new ItemListener() {
                public void itemStateChanged(ItemEvent e) {
                        shearAction();
                }
        });
        shearBox2.addItemListener(new ItemListener() {
                public void itemStateChanged(ItemEvent e) {
                        shearAction();
                }
        });

        shearPanel.setBackground(Color.lightGray);
        shearBox1.setBackground(Color.lightGray);
```

```
        shearBox2.setBackground(Color.lightGray);
        shearPanel1.setBackground(Color.lightGray);
        shearPanel.setBackground(Color.lightGray);
        shearPanel2.setBackground(Color.lightGray);
        shearPanel.setMaximumSize(new Dimension(200, 60));
        return shearPanel;
}

public void shearAction() {
        Random r = new Random();
        int value = r.nextInt(100);

        // shears the image by a random value
        if(shearBox1.isSelected() && shearBox2.isSelected())
                drawingPanel.shearImage(value/100.0f, value/100.0f);
        else if(shearBox1.isSelected())
                drawingPanel.shearImage(value/100.0f, 0);
        else if(shearBox2.isSelected())
                drawingPanel.shearImage(0, value/100.0f);
        else
                drawingPanel.shearImage(0, 0);
}

public JMenuBar createMenuBar() {
        // create a menuBar
        JMenuBar menuBar = new JMenuBar();

        // add a menu called File to menuBar
        JMenu menu = new JMenu("File");
        menuBar.add(menu);

        // add a menu item called Open to File
        JMenuItem menuItem = new JMenuItem("Open Image");
        menu.add(menuItem);

        menuItem.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        selectFile();
                }
        });

        // add a menu called Image to menuBar
        menu = new JMenu("Image");
        menuBar.add(menu);

        // add a menu items
        JMenuItem menuItemRotate = new JMenuItem("Rotate");
        menu.add(menuItemRotate);

        menuItemRotate.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        rotateImage();
                }
        });

        JMenuItem menuItemShear = new JMenuItem("Shear");
        menu.add(menuItemShear);

        menuItemShear.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        shearImage();
                }
        });

        JMenuItem menuItemScale = new JMenuItem("Scale");
        menu.add(menuItemScale);

        menuItemScale.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                        scaleImage();
                }
        });

        menuBar.setBackground(Color.lightGray);
        return menuBar;

}
```

```
	private void rotateImage() {
		String rotateAngle = JOptionPane.showInputDialog(window, "Enter the angle: ", "Rotate Image",
JOptionPane.PLAIN_MESSAGE);
		double value = Double.valueOf(rotateAngle);
		drawingPanel.rotateImage(Math.toRadians(value));
	}

	private void shearImage() {
		JPanel shearPanel = new JPanel();
		shearPanel.setLayout(new BoxLayout(shearPanel, BoxLayout.Y_AXIS));

		// create two panels for holding each check box.
		JPanel shearPanel1 = new JPanel(new FlowLayout(FlowLayout.LEFT, 26, 6));
		JPanel shearPanel2 = new JPanel(new FlowLayout(FlowLayout.LEFT, 26, 6));

		// create a field for horizontal shear
		JTextField xField = new JTextField(5);
		shearPanel1.add(new JLabel("Enter the horizontal sheare value: "));
		shearPanel1.add(xField);
		shearPanel.add(shearPanel1);

		shearPanel.add(Box.createHorizontalStrut(15)); // a spacer

		// create a field for vertical shear
		JTextField yField = new JTextField(5);
		shearPanel2.add(new JLabel("Enter the vertical sheare value: "));
		shearPanel2.add(yField);
		shearPanel.add(shearPanel2);

		float result = JOptionPane.showConfirmDialog(window, shearPanel, "Shear image",
JOptionPane.PLAIN_MESSAGE);
		// shears the image by the input value
		float x = Float.valueOf(xField.getText());
		float y = Float.valueOf(yField.getText());
		drawingPanel.shearImage(x/100.0f, y/100.0f);
	}

	private void scaleImage() {
		String scaleSize = JOptionPane.showInputDialog(window, "Enter the size in %: ", "Scale Image",
JOptionPane.PLAIN_MESSAGE);
		float value = Float.valueOf(scaleSize)/100.0f;
		drawingPanel.scaleImage(value);
	}

	public void selectFile() {
		chooser = new JFileChooser();

		// This file filter allows the user to select JPEG files only
		FileNameExtensionFilter filter = new FileNameExtensionFilter("JPEG files", "JPG", "JPEG");
		chooser.setFileFilter(filter);
		int returnVal = chooser.showOpenDialog(window);
		if (returnVal == JFileChooser.APPROVE_OPTION) {
			//open a dialog box to select files
			File file = chooser.getSelectedFile();
			System.out.println(file.getPath());

			// load the image from the file and put it in drawing panel
			Image image =  new javax.swing.ImageIcon(file.getPath()).getImage();
			drawingPanel.loadImage(image);

			// update the title of the image
			titleLabel.setText(file.getName());
			titlePanel.repaint();
		}
	}


	public static void main(String[] args) {
		new PhotoOpHomeWork();
	}
}
```


Photo Op
File
Image
Rotate
Shear
Scale
squirrelMonkey.jpg
Scale: 100 %
Shear: Horizontal
Vertical

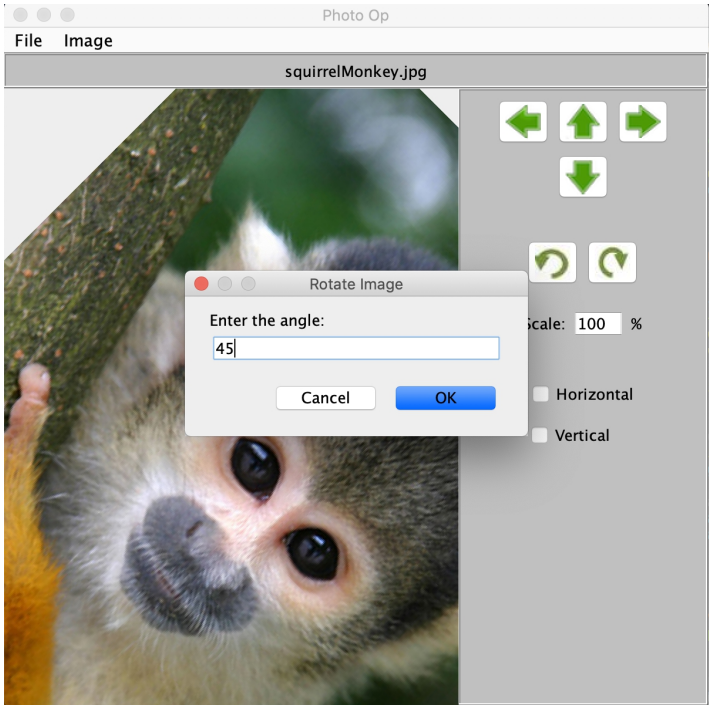
Photo Op
File
Image
squirrelMonkey.jpg
Rotate Image
Enter the angle:
45
Cancel
OK
Horizontal
Vertical

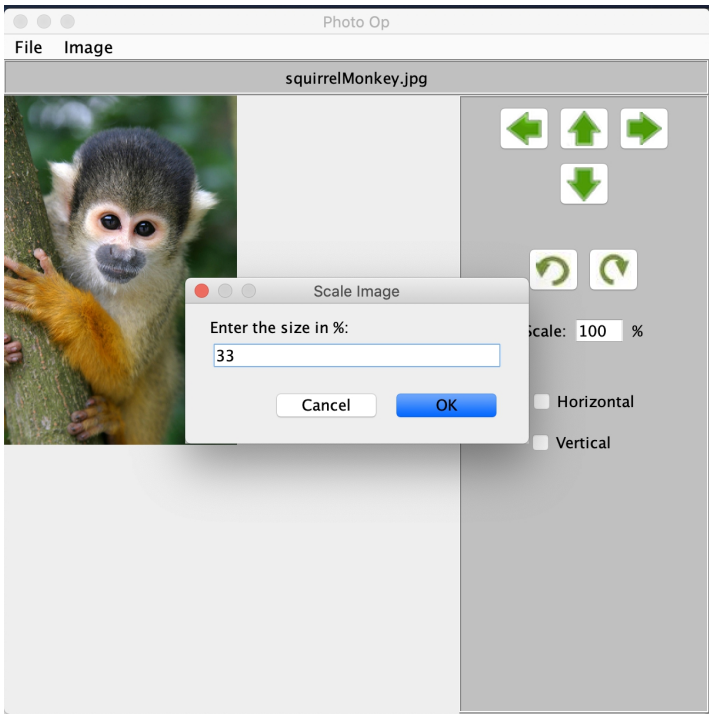
Photo Op
File
Image
squirrelMonkey.jpg
Scale Image
Enter the size in %:
33
Cancel
OK
Horizontal
Vertical

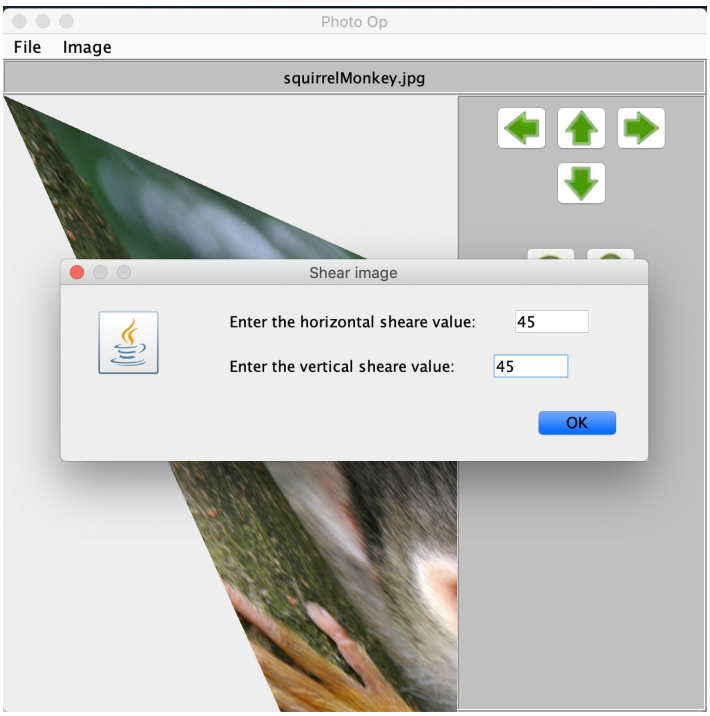
Photo Op
File
Image
squirrelMonkey.jpg
Shear image
Enter the horizontal sheare value:
45
Enter the vertical sheare value:
45
OK