

CIS 142

COVER PAGE
INTRODUCTION TO JAVA
Dondi Hanson

Nick Messenger

UNIT 10 EXERCISES



TABLE OF CONTENTS

COVER PAGE..... 1

TABLE OF CONTENTS..... 2

NARRATIVE 3

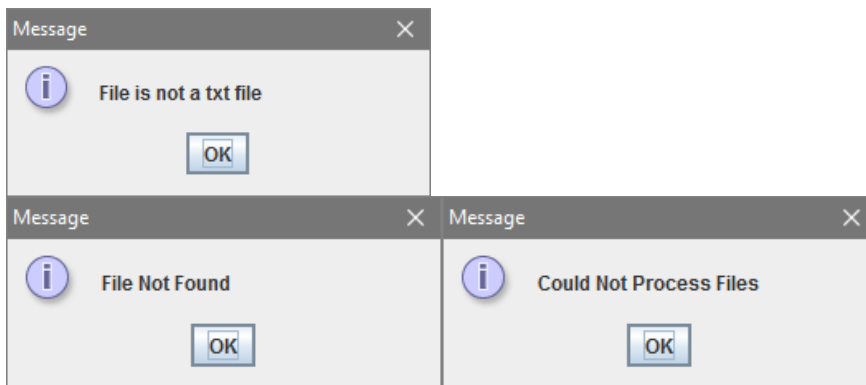
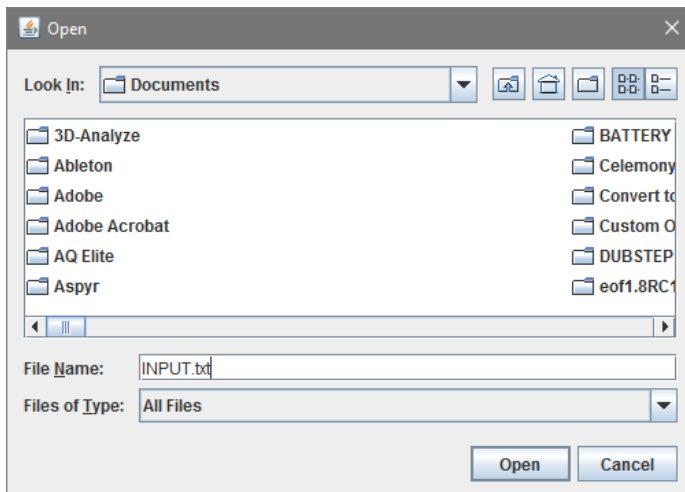
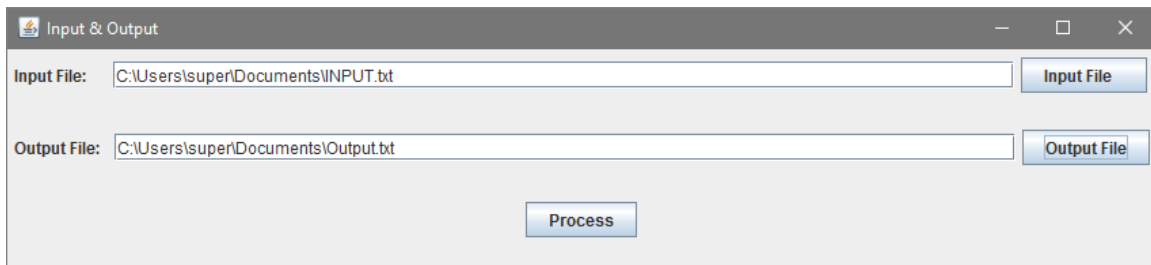
PROGRAM OUTPUT 3

SOURCE CODE..... 4

NARRATIVE

This unit was very cool to me. I have been wanting to know how to deal with files for a little while but I'm glad it was one of the last things we dealt with because it is easier to do with a GUI and exception handling for sure. This unit was ok for me. A little challenging but not too hard. The only problem I had was with the adding logic for outputting the sum of numbers to the output file. Other than that, I felt accomplished when I had a full working GUI with errors and output files being created. Thanks for a great quarter!

PROGRAM OUTPUT



SOURCE CODE

No matter what I do I cannot get this to format correctly within word. I have tried the settings you mentioned in the past within eclipse which normally works but this time it does not. I have also tried copying and pasting from several other programs as well but the lines keep wrapping over too much.

```
import java.awt.Container;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.HeadlessException;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.PrintWriter;
import java.util.Scanner;

import javax.swing.JButton;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class FileChooserProgram extends JFrame implements
ActionListener
{
    protected JTextField txtInFile;
    private JTextField txtOutFile;
    private JButton btnInFile;
    private JButton btnOutFile;
    private JButton btnProcess;
    private File inFile;
    protected File outFile;

    public FileChooserProgram()
    {

        Container canvas = this.getContentPane();
        canvas.setLayout(new GridLayout(3, 1));

        canvas.add(createInFilePanel());
        canvas.add(createOutFilePanel());
        canvas.add(createProcessPanel());

        this.setVisible(true);
        this.setTitle("Input & Output");
        this.setSize(863, 200);
        this.setLocationRelativeTo(null);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    }

    public JPanel createInFilePanel()
    {
```

Nick Messenger Unit 10 Exercises

```
JPanel panel = new JPanel(new FlowLayout(FlowLayout.LEFT));
panel.add(new JLabel("Input File: "));
txtInFile = new JTextField(60);
panel.add(txtInFile);
btnInFile = new JButton("Input File");
btnInFile.addActionListener(this);
panel.add(btnInFile);

return panel;
}

public JPanel createOutFilePanel()
{
    JPanel panel2 = new JPanel(new
FlowLayout(FlowLayout.LEFT));
    panel2.add(new JLabel("Output File: "));
    txtOutFile = new JTextField(60);
    panel2.add(txtOutFile);
    btnOutFile = new JButton("Output File");
    btnOutFile.addActionListener(this);
    panel2.add(btnOutFile);

    return panel2;
}

public JPanel createProcessPanel()
{
    JPanel panel3 = new JPanel(new
FlowLayout(FlowLayout.CENTER));
    btnProcess = new JButton("Process");
    btnProcess.addActionListener(this);
    panel3.add(btnProcess);

    return panel3;
}

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

@Override
public void actionPerformed(ActionEvent e)
{
    if (e.getSource() == btnInFile)
    {
        try
        {
            JFileChooser jfcInFile = new JFileChooser();
            if (jfcInFile.showOpenDialog(this) !=
JFileChooser.CANCEL_OPTION)
            {
                inFile = jfcInFile.getSelectedFile();

                txtInFile.setText(inFile.getAbsolutePath());
            }
        }
    }
}
```

```

        if (!inFile.exists())
        {
            throw new FileNotFoundException();
        }
        if (inFile.isDirectory())
        {
            throw new FileNotFoundException();
        }
        if (!inFile.getName().endsWith(".txt"))
        {
            throw new NotInTxtException();
        }
    } catch (NotInTxtException e4)
    {
        JOptionPane.showMessageDialog(null,
e4.getMessage());

        txtInFile.setText("");
        txtInFile.grabFocus();
    } catch (FileNotFoundException e1)
    {
        JOptionPane.showMessageDialog(null, "File Not
Found");

        txtInFile.setText("");
        txtInFile.grabFocus();
    }
}

if (e.getSource() == btnOutFile)
{
    try
    {
        JFileChooser jfcOutFile = new JFileChooser();
        if (jfcOutFile.showOpenDialog(this) !=
JFileChooser.CANCEL_OPTION)
        {
            outFile = jfcOutFile.getSelectedFile();

            txtOutFile.setText(outFile.getAbsolutePath());
        }

        if (outFile.isDirectory())
        {
            throw new FileNotFoundException();
        }
        if (!outFile.getName().endsWith(".txt"))
        {
            throw new NotInTxtException();
        }
    } catch (NotInTxtException e4)
    {
        JOptionPane.showMessageDialog(null,
e4.getMessage());

        txtOutFile.setText("");
        txtOutFile.grabFocus();
    } catch (FileNotFoundException e1)
    {

```

Nick Messenger Unit 10 Exercises

```
        JOptionPane.showMessageDialog(null, "File Not
Found");

        txtOutFile.setText("");
        txtOutFile.grabFocus();
    }

    }
    if (e.getSource() == btnProcess)
    {

        {

            {

                PrintWriter fout = null;
                Scanner fin = null;

                {

                    File file = new

File(txtInFile.getText());

                    try
                    {

                        fin = new Scanner(file);
                        fout = new

PrintWriter(outFile);

                        // Can't get logic to work
                        // zero because the first
                        // gone through all of the
                        // But if I put the adding
                        // as printing original
                        // occurs and there are weird

                        int sum = 0;
                        int num = 0;

                        while (fin.hasNextInt())
                        {
                            num = fin.nextInt();
                            fout.print(num + " +

");

                            sum += num;
                        }
                        fout.print("0 = ");
                        fout.print(sum);

                        JOptionPane.showMessageDialog(null, "Processed. Please Check
Output File");

                    }
                }
            }
        }
    }
}
```

```
        catch (FileNotFoundException e1)
        {
            JOptionPane.showMessageDialog(null, "Could Not Process Files");
        }

        finally
        {
            if (fin != null)
                fin.close();
            if (fout != null)
                fout.close();
        }
    }
}

import java.io.FileNotFoundException;

public class NotInTxtException extends FileNotFoundException
{
    public NotInTxtException()
    {
        this("File is not a txt file");
    }
    public NotInTxtException(String mess)
    {
        super(mess);
    }
}
```