Daniel Gopal

**Level 1: Dialogue Based Output**

1. Use the Eclipse (Juno or later) IDE. Use Eclipse from the applications folder on the classroom computers.

2. Follow the instructions in the “Hello World SWT” tutorial

a. Create a new Java project

b. Import SWT into your project following the tutorial

c. Display a dialog box with a “Hello World” title (or something more useful)

import org.eclipse.swt.SWT;  
import org.eclipse.swt.widgets.Display;  
import org.eclipse.swt.widgets.Shell;  
import org.eclipse.swt.widgets.Text;  
  
public class HelloWorldSWT {  
  
public static void main (String [] args) {  
 //The above SWT example will create a TextBox and display it as “Hello World”.  
 //This is the display screen for the SWT  
 Display display = new Display ();   
 //Stores the display into the shell variable  
 Shell shell = new Shell(display);  
 //All application’s GUI are rendered in display.  
 Text Character = new Text(shell, SWT.NONE);  
 Character.setText("Hello World"); //Sets the text 'HelloWorld'   
 Character.pack();  
   
 shell.pack();//Tells the SWT application to auto resize the widget (shell windows) to its preferred size,   
 //It uses only as much space based on resolution and platform rendering  
 shell.open ();  
 while (!shell.isDisposed ()) {  
 if (!display.readAndDispatch ()) display.sleep (); //Display.readAndDispatch()keeps track of user events in applications   
 //like when closing windows  
 }  
 display.dispose ();  
 }  
 }

**Level 2: Dialogue Based Input**

1. Research how to use SWT to read user input into your program.

2. Read string data

a. Read a user typed string from a dialogue box

b. Display the string in another dialogue box

import org.eclipse.swt.SWT;  
import org.eclipse.swt.widgets.Display;  
import org.eclipse.swt.widgets.Shell;  
import org.eclipse.swt.widgets.Text;  
  
import java.util.Scanner;  
  
  
public class StringSWT {  
 //Static allows for the variable to be used in any method created within this class  
 static String Word;//Allows for the string to be used outside global variables  
   
 static Scanner sc = new Scanner(System.in);//Allows for the Scanner to be used outside global variables  
  
public static void main (String [] args) {  
 //Ask the user to type a word  
 System.out.println("Please type a word: ");  
 Word = sc.nextLine();  
   
 //The above SWT example will create a TextBox and display it as “Hello World”.  
 //This is the display screen for the SWT  
 Display display = new Display ();   
 //Stores the display into the shell variable  
 Shell shell = new Shell(display);  
 //All application’s GUI are rendered in display.  
 Text Character = new Text(shell, SWT.NONE);  
 Character.setText(Word); //Stores the word  
 Character.pack();  
   
 shell.pack();//Tells the SWT application to auto resize the widget (shell windows) to its preferred size,   
 //It uses only as much space based on resolution and platform rendering  
 shell.open ();  
 while (!shell.isDisposed ()) {  
 if (!display.readAndDispatch ()) display.sleep (); //Display.readAndDispatch()keeps track of user events in applications   
 //like when closing windows  
 }  
 display.dispose ();  
 }  
 }

3. Read integer data

a. Read a user typed integer number from a dialogue box

b. Multiply the number by 5

c. Display the result in another dialogue box

import org.eclipse.swt.SWT;  
import org.eclipse.swt.widgets.Display;  
import org.eclipse.swt.widgets.Shell;  
import org.eclipse.swt.widgets.Text;  
  
import java.util.Scanner;  
  
  
public class IntSWT {  
 //Static allows for the variable to be used in any method created within this class  
 static int number;//Allows for the integer to be used outside global variables  
   
 static int TimesBy5;//This will be used to multiply the user input by 5  
   
 static String NumtoString; //Allows for the string to be used outside global variables  
   
 static Scanner sc = new Scanner(System.in);//Allows for the Scanner to be used outside global variables  
  
public static void main (String [] args) {  
 //Ask the user to type a word  
 System.out.println("Please type a number: ");  
 number = sc.nextInt();  
   
 //Multiplies the integer by 5  
 TimesBy5 = number \* 5;   
   
 //Converts the integer to a String so that it can be outputted as a text type  
 NumtoString = Integer.toString(TimesBy5);   
   
 //The above SWT example will create a TextBox and display it as “Hello World”.  
 //This is the display screen for the SWT  
 Display display = new Display ();   
 //Stores the display into the shell variable  
 Shell shell = new Shell(display);  
 //All application’s GUI are rendered in display.  
 Text Character = new Text(shell, SWT.NONE);  
 Character.setText(NumtoString); //Stores the number  
 Character.pack();  
   
 shell.pack();//Tells the SWT application to auto resize the widget (shell windows) to its preferred size,   
 //It uses only as much space based on resolution and platform rendering  
 shell.open ();  
 while (!shell.isDisposed ()) {  
 if (!display.readAndDispatch ()) display.sleep (); //Display.readAndDispatch() keeps track of user events in applications   
 //like when closing windows  
 }  
 display.dispose ();  
 }  
 }

**Level 3: Factorial Calculator**

1. Research the meaning of a “factorial” related to mathematics. Research the equation to calculate a factorial.

2. Design an algorithm (program method) using variables, loops, and conditionals to calculate a factorial for any number.

a. Resource:<https://www.java-made-easy.com/java-for-beginners.html>

3. Write a program to calculate a factorial as follows:

a. Get user input for a number using a dialogue box

b. Calculate the factorial using your algorithm (method)

c. Display the result in a dialogue box

//A dialog box that displays a message to the user and allows for inputs within the same window  
import javax.swing.JOptionPane;  
  
import org.eclipse.swt.SWT;  
import org.eclipse.swt.widgets.Display;  
import org.eclipse.swt.widgets.Shell;  
import org.eclipse.swt.widgets.Text;  
  
public class FactorialCalculatorSWT {  
  
 public static void main(String[] args) {  
 //Allows for the integer to be used within the JOptionPane dialog box string based input  
 int number = Integer.parseInt(JOptionPane.showInputDialog("Please enter a number"));  
   
 //This will help find the factorial of the inputted number  
 int fact = 1;  
 for(int i = 1; i <= number; i++)  
 {  
 fact = fact \* i;  
 }  
 //Converts the integer to a String so that it can be outputted as a text type  
 String NumtoString = Long.toString(fact);   
   
 //Prints the factorial/answer in another dialog box  
 JOptionPane.showMessageDialog(null, "The factorial is " + fact);  
   
 }  
 }