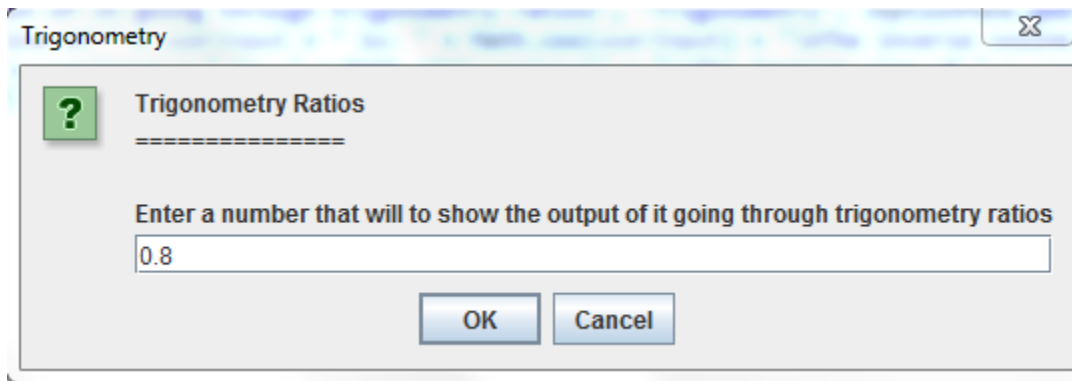


```

import javax.swing.JOptionPane;
/*
 * Justin Mendes
 * Created: September 14, 2017
 * Last Edited: September 14, 2017
 * Unit 1 Activity 1 Program/Question 1
 * This program will display the outputs of a number used in trigonometry functions
 (USING MATH CLASSES)
 */
public class Trigonometry
{
    public static void main(String[] args)
    {
        //Variable Declarations and Initializations
        double userInput;
        userInput = Double.parseDouble(JOptionPane.showInputDialog(null,
"Trigonometry Ratios\n=====\n\nEnter a number that will "
+ "to show the output of it going through trigonometry
ratios", "Trigonometry", JOptionPane.QUESTION_MESSAGE));
        System.out.println("The cosine of " + userInput + " is: " +
Math.cos(userInput) + "\nThe inverse cosine of " + userInput + " is: " +
Math.acos(userInput) +
"\n\nThe sine of " + userInput + " is: " +
Math.sin(userInput) + "\nThe inverse sine of " + userInput + " is: " +
Math.asin(userInput) +
"\n\nThe tangent of " + userInput + " is: " +
Math.tan(userInput) + "\nThe inverse tangent of " + userInput + " is: " +
Math.atan(userInput));
    } //end main
} //end class

```



```

The cosine of 0.8 is: 0.6967067093471654
The inverse cosine of 0.8 is: 0.6435011087932843

The sine of 0.8 is: 0.7173560908995228
The inverse sine of 0.8 is: 0.9272952180016123

The tangent of 0.8 is: 1.0296385570503641
The inverse tangent of 0.8 is: 0.6747409422235527

```

```

/*
 * Justin Mendes
 * Created: July 29, 2017
 * Last edited: September 14, 2017
 * Unit 1 Activity 1 Program/Question 2
 * This program will find the date of the Easter in the given year
 */
import javax.swing.*;
public class EasterVariation2
{
    public static void main(String[] args)
    {
        //a - m - the quotients and remainders needed
        int a, b, c, d, f, g, h, j, k, n, p, q;
        int month; //Easter month
        int day; //Easter day
        int year; //Easter's year
        String monthStr;

        //Ask user for year
        String yearStr = JOptionPane.showInputDialog("Easter "
            +"Date Calculator\nPlease enter a year");

        //Convert the year from a String to an int
        year = Integer.parseInt(yearStr);

        //Calculations
        a = year / 100;
        b = year % 100;
        c = (3 * (a + 25)) / 4;
        d = (3 * (a + 25)) % 4;
        q = (8 * (a + 11)) / 25;
        f = (5 * a + b) % 19;
        g = (19 * f + c - q) % 30;
        h = (f + 11 * g) / 319;
        j = (60 * (5 - d) + b) / 4;
        k = (60 * (5 - d) + b) % 4;
        n = (2 * j - k - g + h) % 7;
        p = (g - h + n + 114) % 31;
        month = (g - h + n + 114) / 31;
        day = p + 1;

        //determine the month in String format
        if(month == 3)
        {
            monthStr = "March";
        } //end if
        else
        {
            monthStr = "April";
        } //end else

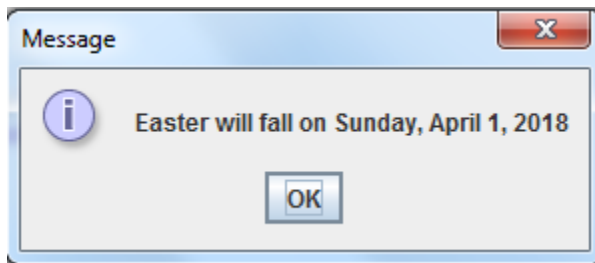
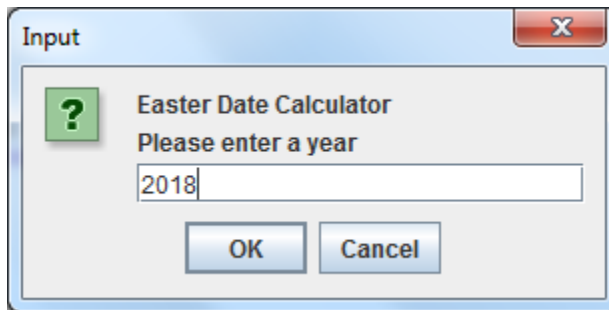
        //Display the results
        JOptionPane.showMessageDialog(null, "Easter will fall"
            +" on Sunday, "+monthStr+" "+day+"", "+year");
    }
}

```

```

    }//end main
} //end class

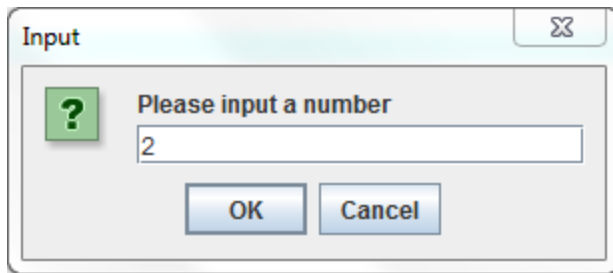
```



```

/*
 * Justin Mendes
 * Created: July 29, 2017
 * Last Edited: September 14, 2017
 * Unit 1 Activity 1 Program/Question 3
 * This program will find the square of the square root of any number and find the
round-off error in the number
 */
import javax.swing.JOptionPane;
public class RoundingError
{
    public static void main(String[] args)
    {
        //Variable Declarations and Initializations
        double number;
        number = Double.parseDouble(JOptionPane.showInputDialog(null, "Please
input a number", "Input", JOptionPane.QUESTION_MESSAGE));
        System.out.println("The square of the square root = " +
Math.pow(Math.sqrt(number), number));
        System.out.println("The round off error = " + (number -
Math.pow(Math.sqrt(number), number)));
    } //end main
} //end class

```



```
<terminated> RoundingError [Java Application] C:\Program Files\Ja
The square of the square root = 2.0000000000000004
The round off error = -4.440892098500626E-16
```

```
import java.text.DecimalFormat;
import java.util.Scanner;
import javax.swing.JOptionPane;
/*
 * Justin Mendes
 * Created: July 29, 2017
 * Last Edited: September 15, 2017
 * Unit 1 Activity 1 Program/Question 4
 * This program will calculate the area of a scalene triangle using either Heron's
 * formula or using trigonometry.
 */
public class AreaOfTriangle
{
    public static void main(String[] args)
    {
        //for restart function
        int restart = 1;
        while (restart == 1)
        {
            //Variable Declarations and Initializations
            double sides[] = new double[3], area;
            int method;
            DecimalFormat twoDigit = new DecimalFormat("###.##");
            Scanner sc = new Scanner(System.in);
            method =
Integer.parseInt(JOptionPane.showInputDialog(null, "Choose which method you would like
to use/n"
                                + "to calculate the area of the scalene
triangle\n\n1 - Heron's Formula\n2 - Trigonometry", "Area Method",
JOptionPane.QUESTION_MESSAGE));
            for(int i = 0; i < 3; i++)
            {
                sides[i] =
Double.parseDouble(JOptionPane.showInputDialog(null, "Please enter side " + (i + 1) +
" of the SCALENE TRIANGLE"
                                                + "\n(no two sides should be the same!)",
"Scalene Input", JOptionPane.QUESTION_MESSAGE));
            } //end loop
            System.out.println("Area of Triangle\n=====");
        }
    }
}
```

```

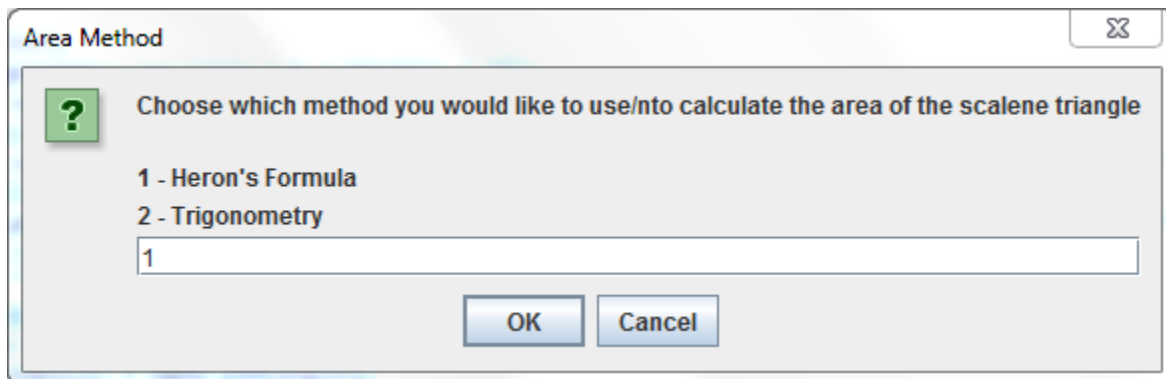
        if (sides[0] + sides[1] > sides[2] && sides[0] + sides[2] >
sides[1] && sides[1] + sides[2] > sides[0] &&
        sides[0] != sides[1] && sides[1] != sides[2] &&
sides[2] != sides[0])
        {
            switch(method)
            {
                case 1: area = calcHeron(sides);
                break;
                default: area = calcTrig(sides);
            } //end switch
            System.out.println("The area of the " + sides[0] + "x" +
sides[1] + "x" + sides[2] + " triangle is: " + twoDigit.format(area) +
"units\u00B2");
        } //end if
    else
    {
        System.out.println("These sides can NOT make a SCALENE
triangle.");
    } //end else
    System.out.println("\nPlease press 1 to try again\nPress anything
else to exit");
    restart = sc.nextInt();
} //end restart loop
} //end main
public static double calcHeron(double[] sides)
{
    double semiP = 0, inSqrt;
    System.out.println("Heron's Formula way:\nThe square root of s(s - a)(s
- b)(s - c), Where semiperimeter = s = (a + b + c) / 2");
    System.out.println("s = (" + sides[0] + " + " + sides[1] + " + " +
sides[2] + ") / 2");
    semiP = (sides[0] + sides[1] + sides[2]) / 2;
    System.out.println("s = " + semiP);
    // \u221A is the symbol for square root which only works with text file
encoding UTF-8
    System.out.println("area = \u221A" + semiP + "(" + semiP + " - "
+sides[0] + ")(" + semiP + " - " +sides[1] + ")(" + semiP + " - " +sides[2] + ")");
    inSqrt = semiP * (semiP - sides[0]) * (semiP - sides[1]) * (semiP -
sides[2]);
    System.out.println("area = \u221A" + inSqrt);
    System.out.println("area = " + Math.sqrt(inSqrt));
    return Math.sqrt(inSqrt);
} //end method calcHeron
public static double calcTrig(double[] sides)
{
    double angle = 0, inBracket;
    System.out.println("Trigonometry way:\n(a + b + Sin(C)) / 2");
    System.out.println("C = cos-1((a\u00B2 + b\u00B2 - c\u00B2) / 2ab)");
    // \u00B2 unicode for squared superscript
    System.out.println("C = cos-1((" + sides[0] + "\u00B2 + " + sides[1] +
"\u00B2 - " + sides[2] + "\u00B2) / 2(" + sides[0] + ")(" + sides[1] + ")");
    //SOLVING FOR C with COSINE LAW
    angle = Math.acos((Math.pow(sides[0], 2) + Math.pow(sides[1], 2) -
Math.pow(sides[2], 2)) / (2 * sides[0] * sides[1])) * 180 / Math.PI;

```

```

        // *180/PI is needed to convert the radian into degrees (that comes from
Math.acos function)
        System.out.println("C = " + angle + "\u00B0");
        // \u00B0 unicode for degrees
        System.out.println("area = (" + sides[0] + " + " + sides[1] + " + Sin("
+ angle + "\u00B0)) / 2");
        inBracket = sides[0] + sides[1] + Math.sin(angle);
        System.out.println("area = " + inBracket + " / 2");
        System.out.println("area = " + (inBracket / 2));
        return inBracket / 2;
    } //end method calcTrig
} //end class

```



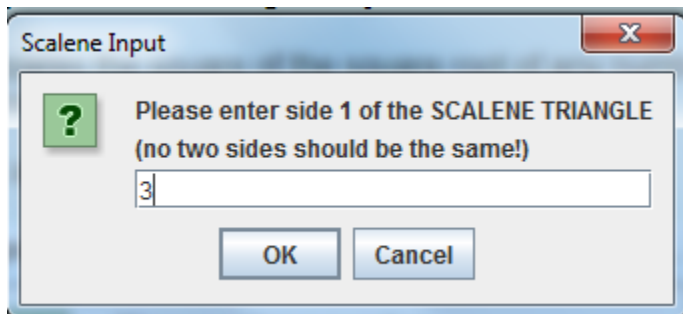
Area Method

Choose which method you would like to use/nto calculate the area of the scalene triangle

- 1 - Heron's Formula
- 2 - Trigonometry

1

OK Cancel

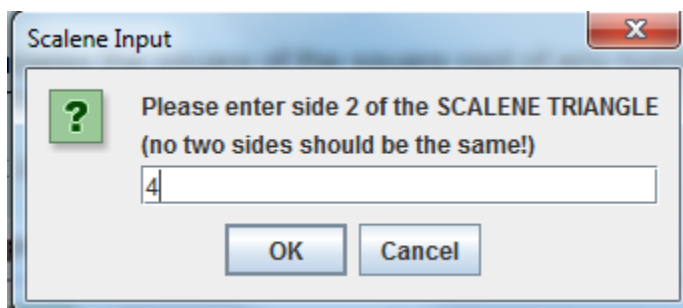


Scalene Input

Please enter side 1 of the SCALENE TRIANGLE
(no two sides should be the same!)

3

OK Cancel

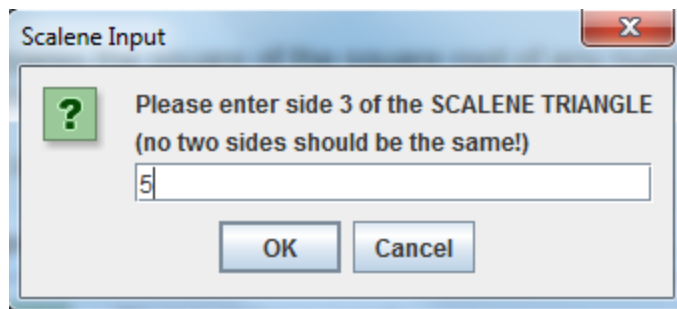


Scalene Input

Please enter side 2 of the SCALENE TRIANGLE
(no two sides should be the same!)

4

OK Cancel



AreaOfTriangle [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Sep 17, 2017, 9:43:06 PM)

Area of Triangle

=====

Heron's Formula way:

The square root of $s(s - a)(s - b)(s - c)$, Where semiperimeter = $s = (a + b + c) / 2$

$s = (3.0 + 4.0 + 5.0) / 2$

$s = 6.0$

$area = \sqrt{6.0(6.0 - 3.0)(6.0 - 4.0)(6.0 - 5.0)}$

$area = \sqrt{36.0}$

$area = 6.0$

The area of the 3.0x4.0x5.0 triangle is: 6units²

Please press 1 to try again

Press anything else to exit