Python PolyAngle Tkinter - Design a GUI

Contents

Time required: 120 minutes

Tutorial: GUI Design Process with PolyAngle

This tutorial idea comes from dead reckoning planning for a student robot project. We are going to create a Python console program that allows you to enter the number of sides in a regular polygon, then calculate the interior and exterior angles and display the type of polygon.

The Math

Start with solving the problem, create the algorithm. An algorithm is process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer.

Problem statement: Given the number of sides of a regular polygon, how do we calculate the interior and exterior angles? Let’s do the math.

The sum of interior angles in a triangle is 180°. To find the sum of interior angles of a polygon, multiply the number of triangles in the polygon by 180°. The formula for calculating the sum of interior angles is (n - 2) x 180° where n is the number of sides. All the interior angles in a regular polygon are equal.

The formula for calculating the size of an interior angle is:

interior angle of a polygon = sum of interior angles ÷ number of sides.

The sum of exterior angles of a polygon is 360°.

The formula for calculating the size of an exterior angle is:

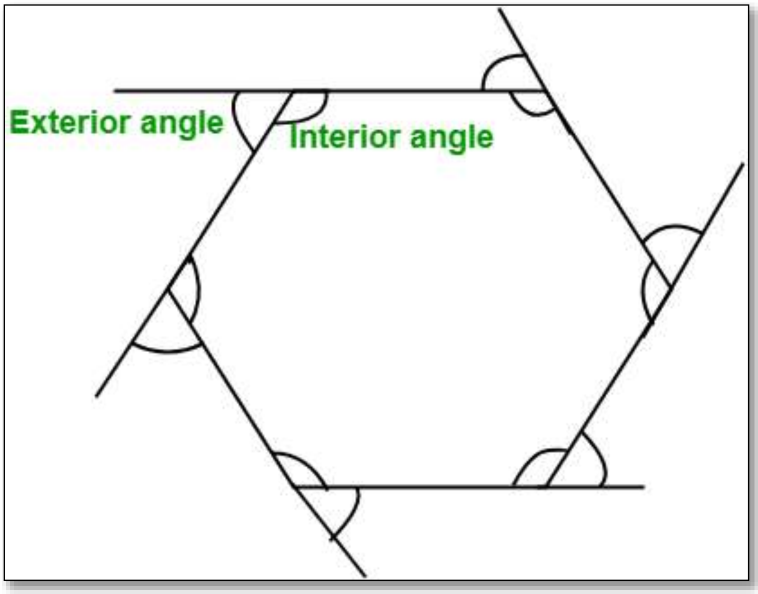
exterior angle of a polygon = 360 ÷ number of sides.

Interior Angle

The angle between two adjacent sides inside the polygon is known as the Interior angle.
Interior Angle = (n-2)180 / n

Exterior Angle

The angle formed by any side of a polygon and the extension of its adjacent side is known as the Exterior angle.
Exterior angle =360 / n



Pseudocode

The first step is a high-level look at the program. Think through what you want your program to do as if you were the user running your completed program.

```
Get the number of sides of a regular polygon from the user
Calculate the interior angle
Calculate the exterior angle
Display the results
```

Assignment: PolyAngle

Let’s convert our Python PolyAngle console program into an OOP tkinter GUI program. The Fahrenheit Tutorial conversion gives you a model for this process.

NOTE: Please program in the OOP style we have been using in past assignments.

GUI Design Process

Let’s go through a process to design and create a simple GUI tool to convert a distance in feet to the equivalent distance in meters.

The high-level pseudocode is the same for a console program as for a GUI program. With a GUI, we also have an interface to design.

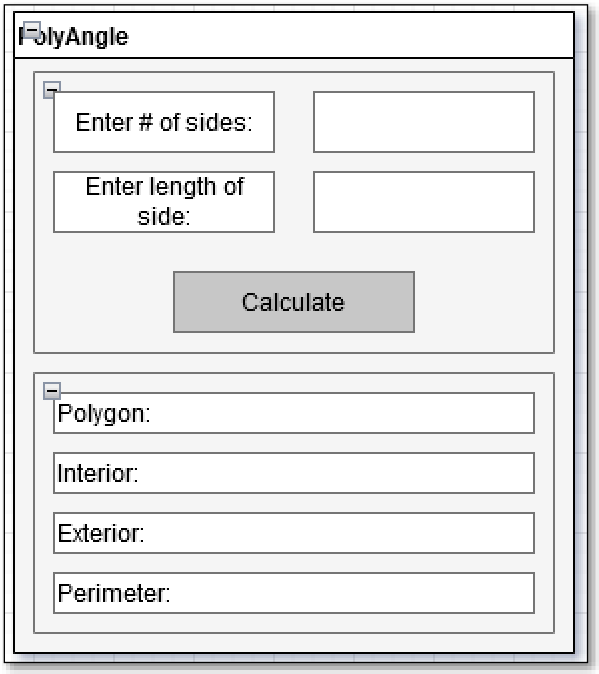
Pseudocode

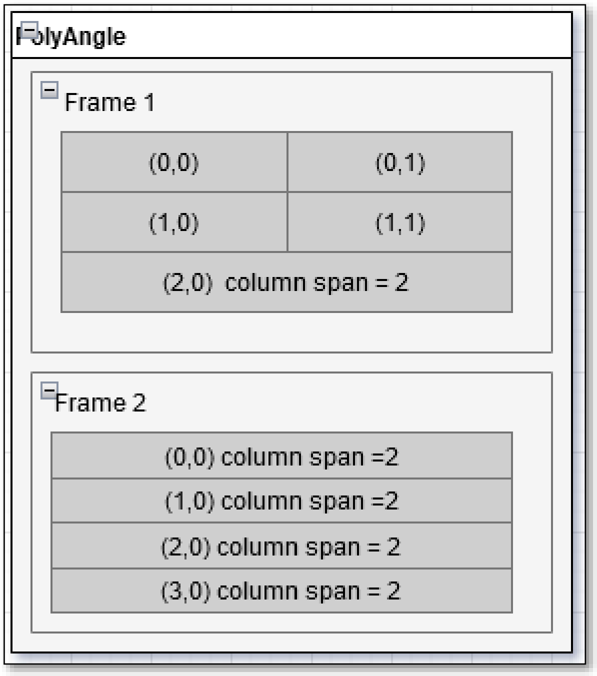
The first step if the high-level look at the program. Think through what you want your program to do if you were the user running your completed program.

```
Get the number of sides of a regular polygon from the user
Calculate the interior angle
Calculate the exterior angle
Display the results
```

Draw It Out

Draw out your sketch on paper. This example was created in Pencil, an open-source GUI prototyping tool that's available for ALL platforms. Your sketch does not have to look the same.




PolyAngle

Frame 1

(0,0)	(0,1)
(1,0)	(1,1)
(2,0) column span = 2	

Frame 2

(0,0) column span =2
(1,0) column span =2
(2,0) column span = 2
(3,0) column span = 2

The Tkinter grid layout manager works very well for most program designs. The following sketch shows how the design might look in a grid layout inside a frame. This visual guide makes it easy to keep track of which row and column each widget is in.

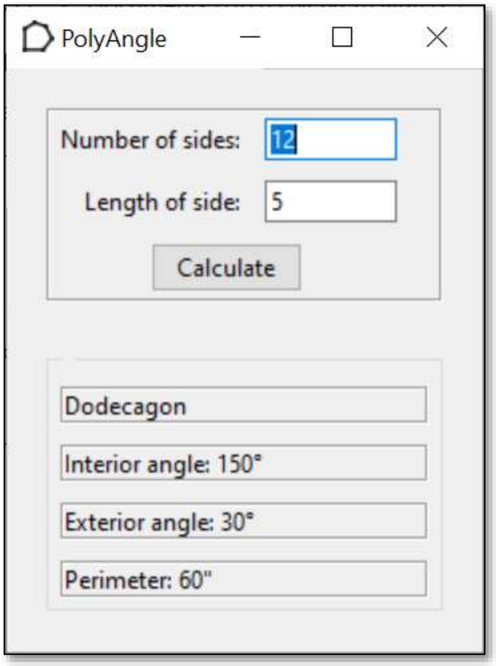
Here are the widgets we will need.

1. One LabelFrame and one Frame to hold all the widgets
2. Text entry to type in the number of sides
3. Calculate button will get the value out of that entry, perform the calculation, and put the result in the labels below the entry.
4. Display labels for the result of each calculation.

Use the techniques we learned in the Python GUI tutorial Temperature Converter and other Tkinter projects to create a tkinter program like the example run.

1. Create high level pseudocode
2. Break the pseudocode into smaller programmable steps
3. Create a sketch for your user interface
4. Use a previous Tkinter project as a starting point.
5. The Polygon names can be a list of names, or individual decisions using if elif.

Example run:



Number of sides: 12

Length of side: 5

```
Dodecagon
Interior angle: 150°
Exterior angle: 30°
Perimeter: 60"
```

Assignment Submission

1. Attach your user interface sketch to the assignment.
2. Attach all python program files to the assignment.
3. Attach screenshots showing the successful operation of the program.
4. Submit in Blackboard