Portfolio Project: Turtle

Recommended Resource - Python turtle API: https://docs.python.org/3.4/library/turtle.html#overview-of-available-turtle-and-screen-methods

In week 1 we drew interesting shapes. Today we'll use python's turtle module to draw shapes using loops, functions, and variables to draw pictures.

An important skill as a coder is the ability to learn how to use an existing API library by reading documentation, finding good API usage examples on the internet, and adapting those examples to meet your needs. In this activity, you will begin building these.

Step 1. Draw a line

Create a new file for your turtle program. A basic turtle program looks like the following:

```
import turtle
turtle.shape("turtle")   # optional
turtle.speed(0)          # optional

turtle.forward(90)
turtle.left(90)
turtle.forward(90)

turtle.Screen().exitonclick()
```

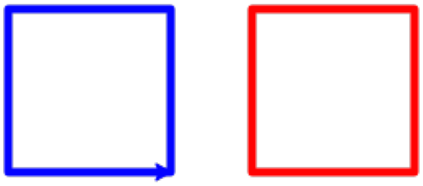
The first & last lines should be in every turtle program you write. **NOTE**: Do NOT name your program turtle.py (or python will lose access to the original turtle package and try to use your file for drawing turtles instead).

Step 2. Draw a square

1. Draw a square on the screen.
2. Change its color.
3. Make the line thicker by changing the pensize or width.
4. Change the speed the turtle draws.

Step 3. Draw two squares

It's possible to move the turtle without drawing. Draw two squares on the screen that don't touch. See the example on the right. To speed up testing, set the turtle speed to be 0.



Step 4. Create a square function
Revise your program to include a square function that takes the length of each side as a parameter & draws a square using the turtle. Call this function instead of the lines you used to draw the squares in the prior two steps. ***Using a loop is recommended.***

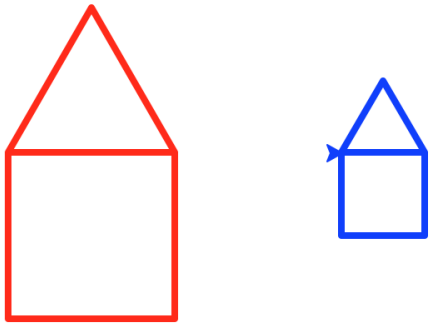
Step 5. Create a rectangle function
Next we are going to generalize our square function by creating a rectangle function that takes a width & a height as parameters and uses the turtle to draw a rectangle.

Step 6. Draw a picture
Using your newly created square & rectangle functions, draw an interesting picture using a **for loop** that calls rectangle and/or square. In your picture, try to include ***at least*** one rectangle and one square. ***Your program should not take any user input.***

Step 7. Draw a picture with functions
Create a program **`house.py`** that has the following functions:

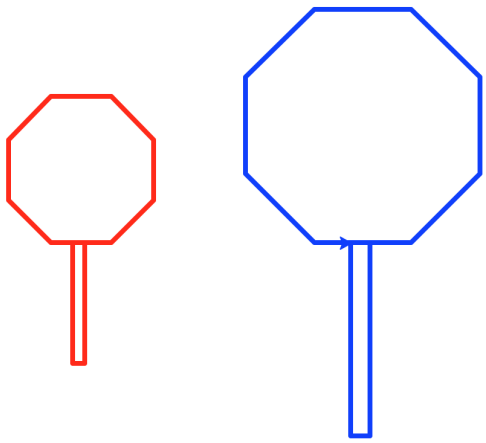
- **square**: takes the length of a side as a parameter and draws a square (each interior angle is 90°)
- **triangle**: takes the length of a side as a parameter and draws an equilateral triangle (each exterior turn is 120°)
- **house**: takes the length of a line as a parameter and draws a house by calling your triangle and square functions.

Make sure that your square and triangle functions use a for loop. **Test** that your functions are working by drawing multiple houses of different sizes. For example:



Next, create a program **`stop.py`** that has the following functions:

- **rectangle**: takes the width and height as parameters and draws a rectangle (each interior angle is 90°)
- **octagon**: takes the side length as a parameter and draws an octagon (each exterior turn is 45°)
- **stop**: takes the octagon side length as a parameter and draws a stop sign by calling octagon, moving forward 3/8 of the side length, and drawing a rectangle sign post that is 1/5 of the side wide and has a height that is double a side.



Note that your shape functions ***must*** use a for loop. **Test** that your functions are working by drawing multiple stop signs of different sizes. For example:

Celebrate your progress!

Take a screenshot of your drawing and share it in the group:

- Mac: http://www.wikihow.com/Take-a-Screenshot-in-Mac-OS-X
- Windows: http://www.wikihow.com/Take-a-Screenshot-in-Microsoft-Windows