**Chapter 7 – Week 10 – Exercises**

Exercises #1 – page 218

1. **Explain the importance of the interface of a class of objects**

A class interface is important because it is how the programmer interacts with the object. If, for instance, the interface contains hundreds of methods, the programmer will have a hard time finding the proper method to invoke. If the interface is lacking an important task, the programmer will not be able to make proper use of the object.

For these and other reasons, interface design is one of the most important skills in object-oriented programming.

1. **What is object instantiation? What are the options at the programmer’s disposal during this process?**

Object instantiation is the process of creating an object.

During instantiation, the programmer has the option to pass parameters to the constructor. (The constructor is the method responsible for creating and initialize an object.) These parameters are used by the constructor to initialize the state of the object for later use.

1. **Add a function named circle to the polygons module. This function expects the same arguments as the square and hexagon functions. The function should draw a circle. (Hint: the loop iterates 360 times.)**

def circle(t, length):

"""

Draws a circle. Actually, a 360 sided polygon

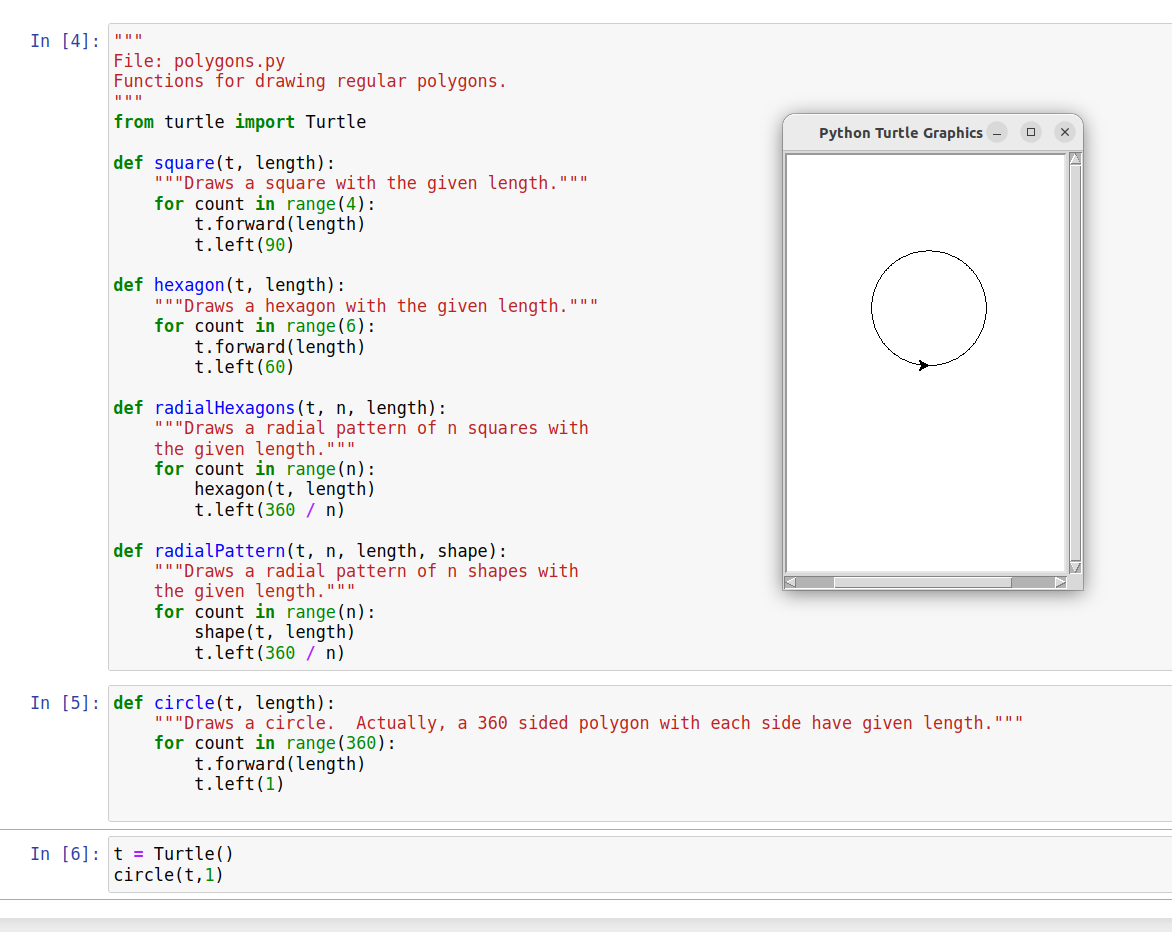
with each side have given length.

"""

for count in range(360):

t.forward(length)

t.left(1)



1. **The functions that draw polygons in the polygons module have the same pattern, varying only in the number of sides (iterations of the loop). Factor this pattern into a more general function named polygon, which takes the number of sides as an additional argument.**

def polygon(t, num\_sides, length):

"""

Draws a polygon with a given number of sides and

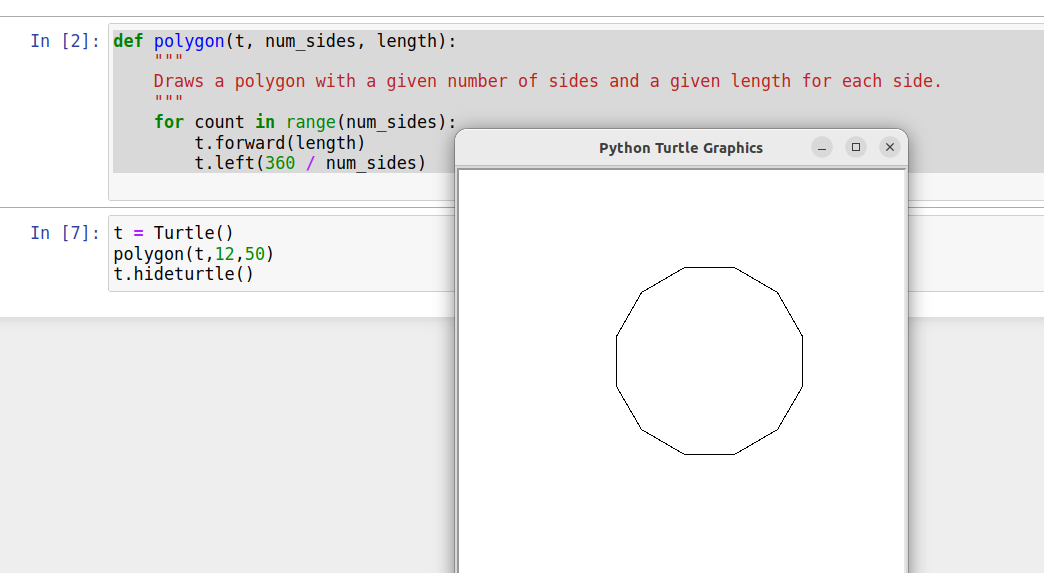
a given length for each side.

"""

for count in range(num\_sides):

t.forward(length)

t.left(360 / num\_sides)



1. **Turtle graphics windows do not automatically expand in size. What do you suppose happens when a Turtle object attempts to move beyond a window boundary?**

When a Turtle object moves beyond the window boundary, it will continue to draw, however, the drawing will not be visible. This is because a canvas has a size that can be bigger than the window. However, if somehow it is possible to exceed the limits of the canvas, I expect that either (1) the canvas will grow automatically to continue to contain the drawings or (2) an exception will be thrown.

1. **The Turtle class includes a method named circle. Import the Turtle class, run help(Turtle.circle), and study the documentation. Then use this method to draw a filled circle and a half moon.**

from turtle import Turtle

tur = Turtle()

tur.begin\_fill()

tur.circle(50)

tur.end\_fill()

tur.up()

tur.goto(-100,0)

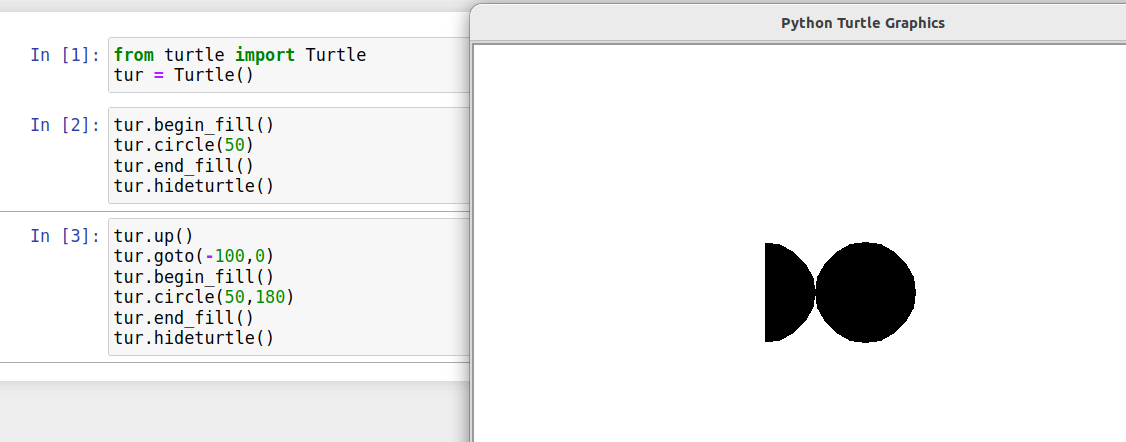
tur.begin\_fill()

tur.circle(50,180)

tur.end\_fill()

tur.hideturtle()

(See screenshot on next page)



Exercises #2 – page 237

1. **Explain the advantages and disadvantages of lossless and lossy image file-compression schemes.**

Lossless file-compression of an image has advantage of keeping all information that was available before compression. In so doing, it requires more space than an algorithm that is willing to lose some information.

Lossy file-compression of an image is the opposite. It is willing to lose some of the original information it an effort to save space (and potentially processing time).

As an additional note, because lossy file-compression looses data, the image will be of a lower quality and the lossless of a higher quality. In this case, a higher quality will look more closely like the original image.

1. **The size of an image is 1680 pixels by 1050 pixels. Assume that this image has been sampled using the RGB color system and placed into a raw image file. What is the minimum size of this file in megabytes? (Hint:There are 8 bits in a byte, 1024 bytes in a kilobyte, and 1000 kilobytes in a megabyte.)**

Space to store one pixel in RGB format: 3 bytes.

Pixels in a 1680x1050 pixel image: 1,764,000.

Total bytes 3 x 1,764,000 = 5,292,000 bytes

= 5168 kilobytes

= 5.047 megabytes (when 1024kb = 1mb)

1. **Describe the difference between Cartesian coordinates and screen coordinates.**

The axes of a Cartesian coordinates system have positive and negative values and the intersection of the axes is in the center of plane in question.

Screen coordinates on the other hand have axes that contain only positive values (and 0 of course). Additionally, the axes intersect at the top, left hand corner of the screen.

1. **Describe how a row-major traversal visits every position in a two-dimensional grid.**

A row-major traversal visits each position in a two-dimensional grid by starting on the first row and visiting each column in that row in order from the lowest column number to the highest. Once the first row is complete, the algorithm moves to the second row visiting each column there. This continues row by row until the last row has been completely visited.

1. **How would a column-major traversal of a grid work? Write a code segment that prints the positions visited by a column-major traversal of a 2-by-3 grid.**

Column-major traversal would start at the first column and visit each row in the column one by one. Once the first column is complete, the next column would be traversed. Each column in traversed until all columns have been traversed in order.

Code Segment:

width=2

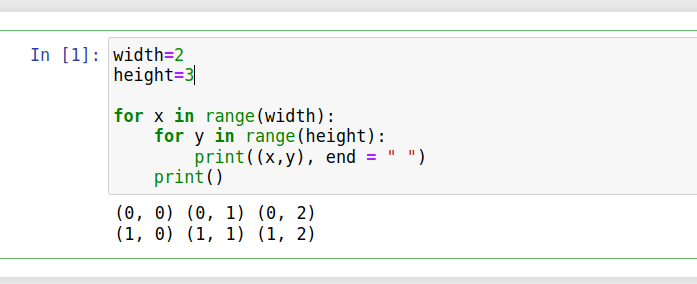
height=3

for x in range(width):

for y in range(height):

print((x,y), end = " ")

print()



1. **Explain why one would use the clone method with a given object.**

A programmer would use the clone method when a copy of an object is required AND the contents of modification of the copy must not effect the original value.

1. **Why does the blur function need to work with a copy of the original image?**

The blur function requires working with the original image and a copy of the original image. This is because as blur() traverses the image it must modify nearly every pixel in the image. The new value depends on the values of the original pixels. Without a copy, changing a pixel to the blurred value would cause its original value to be lost and the results of the blur function to be incorrect.