

Turtle Graphics in Python

By Deborah R. Fowler



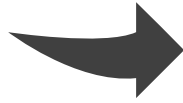
KEY CONCEPTS



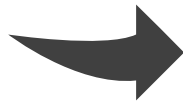
- variables



- truth statements



- looping



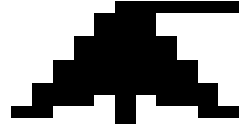
- functions

- I/O

- lists

- classes/objects

- OOP



Why I like Turtle Graphics:

Programming with visual output

Uses a Tkinter window (standard GUI –
Graphical User Interface)



<http://www.deborahrfowler.com/PythonResources/PythonTurtle.html>

<https://runestone.academy/runestone/books/published/thinkcspy/index.html>

or

<http://www.openbookproject.net/thinkcs/python/english2e/>

<https://docs.python.org/3.6/library/turtle.html>



Imagine a turtle with three attributes:

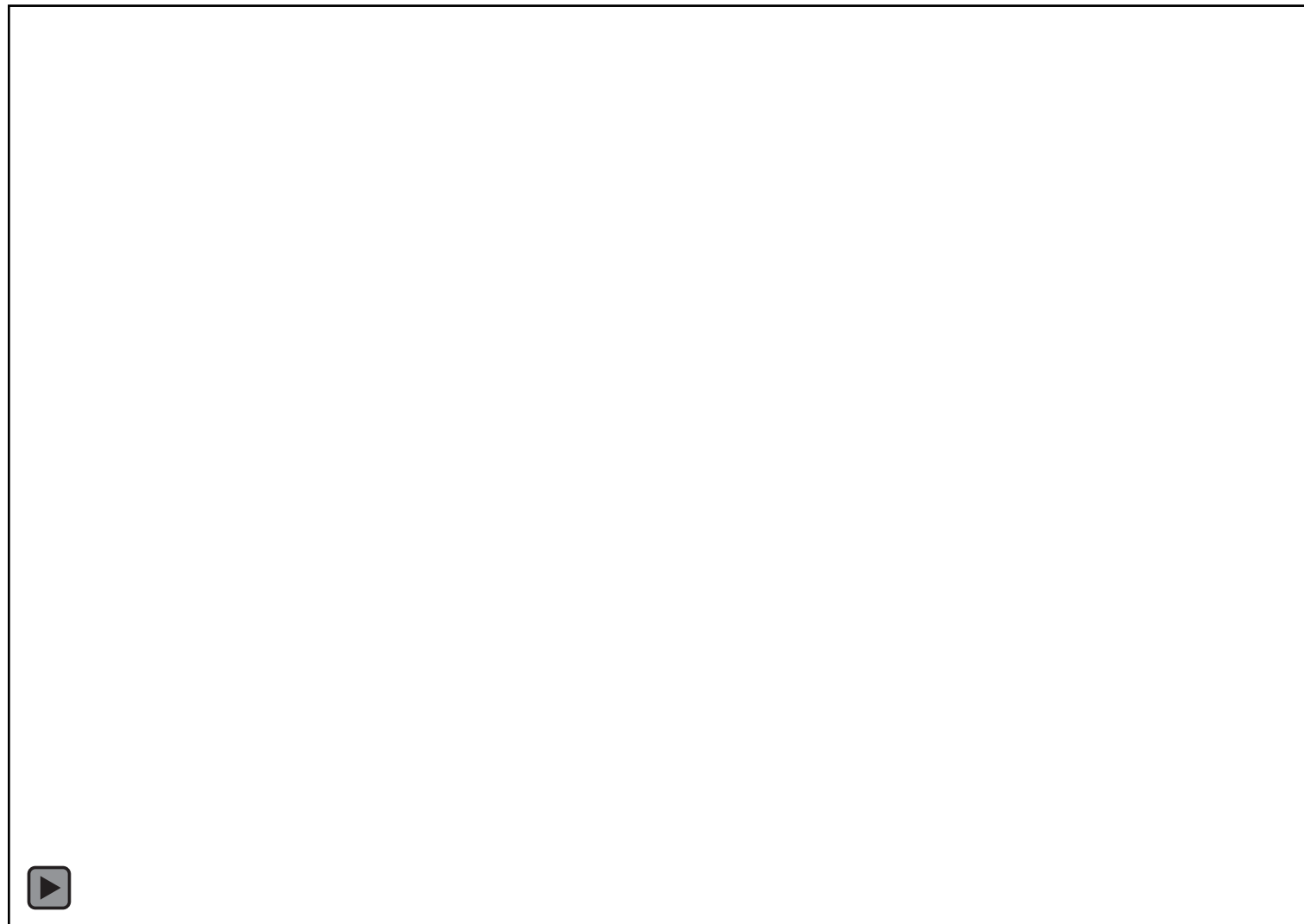
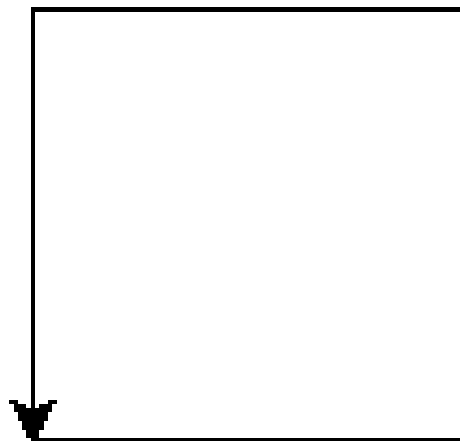
location

orientation

a pen (color, width/up/down)



Works exactly as is in 3.6 as well



in-class exercise:

Draw one of your initials using the turtle library



Looping – a way to repeat code

In python

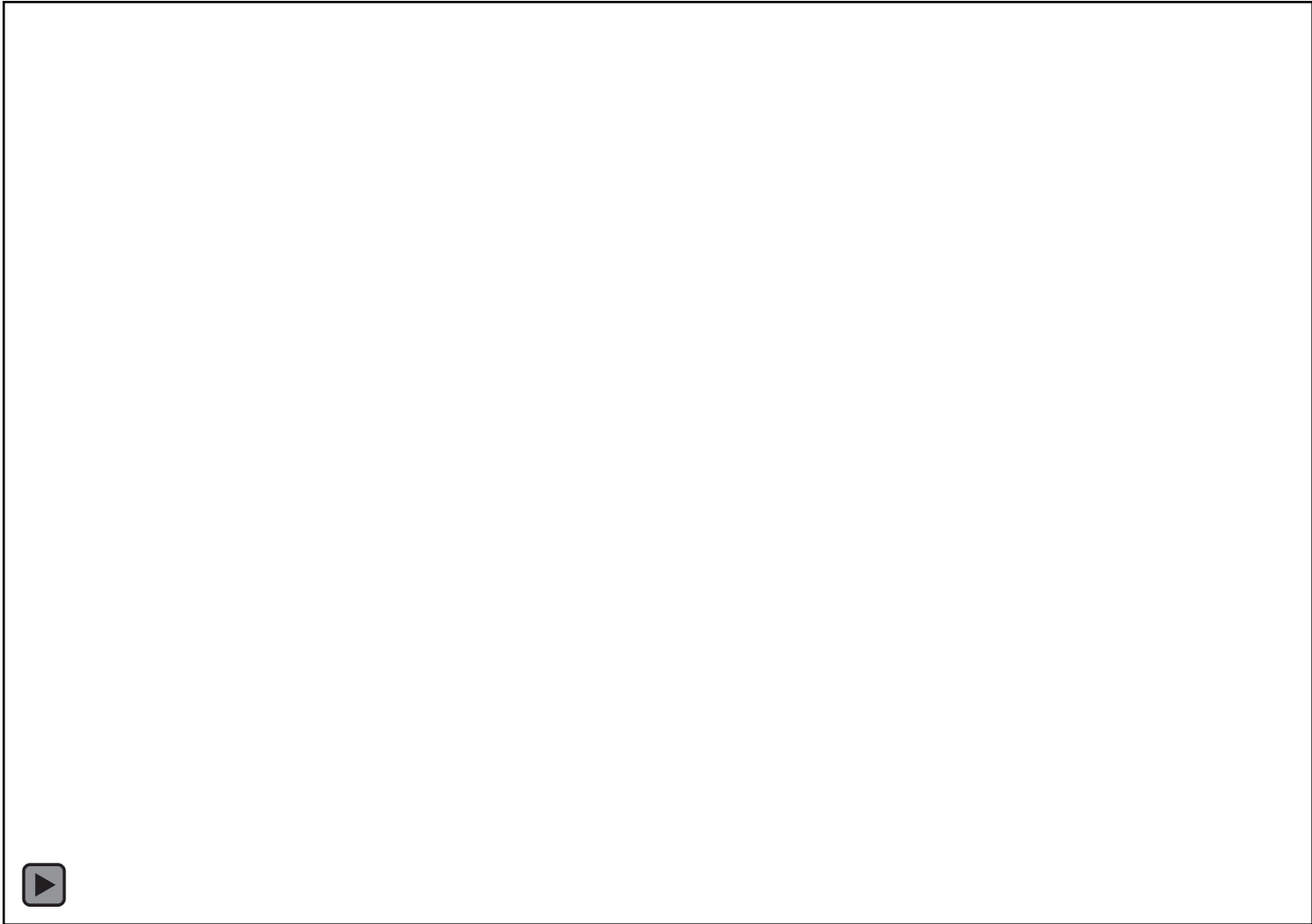
```
for i in range(0,2):  
    print("hello world")
```

```
Python 3.6.8 Shell  
File Edit Shell Debug Options Window Help  
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018, 00:00:00) [AMD64] on win32  
Type "help", "copyright", "credits" or "license()" for more  
>>> for i in range(0,2):  
        print("hello world")  
  
hello world  
hello world  
>>>
```




Works exactly as is in 3.6 as well

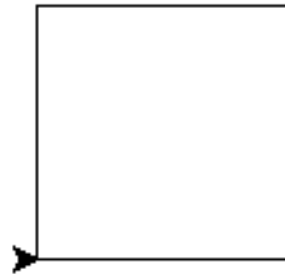
What would
our square
code look like?





Other code

What would happen if we left the last line off?



Works exactly as is in 3.6 as well

```
Python 2.7.14 Shell
File Edit Shell Debug Options Window He
Python 2.7.14 (v2.7.14:84471935ed, 2015-08-24) on win32
Type "copyright", "credits" or "license()" for more
>>> import turtle
>>> i = 0
>>> while i < 4:
>>>     turtle.forward(100)
>>>     turtle.left(90)
>>>     i = i + 1
>>> |
```



Draw more
than one
square?

We can nest
loops



in-class exercise:

Given:

`circle(radius)` – draws a circle of size `radius`

`fillcolor(colorname)` – sets the color attribute for fill

`begin_fill()` and `end_fill()` – similar to `pu()` and `pd()`

Create a snowman – have fun, be creative

Save your file and put it in the dropbox in a “Dailies” Folder

in-class exercise:

List is here:

<https://docs.python.org/3.6/library/turtle.html>



Functions

A group of code statements

Why?

Allows us to organize and build modularly

Allows easy repetition of code

Turning our
square into
a function

squarefn.py - C:/Users/Deborah/Desktop/testing/sq

File Edit Format Run Options Window Help

```
import turtle

def drawSquare():
    for i in range(0,4):
        turtle.forward(100)
        turtle.left(90)

drawSquare()
|
```


Adding
parameters

squarefn.py - C:/Users/Deborah/Desktop/testing/squa

File Edit Format Run Options Window Help

```
import turtle

def drawSquare(size):
    for i in range(0,4):
        turtle.forward(size)
        turtle.left(90)

drawSquare(100)
```

Calling with
variable
arguments

squarefn.py - C:/Users/Deborah/Desktop/testing/squarefn.py

File Edit Format Run Options Window Help

```
import turtle

def drawSquare(size):
    for i in range(0,4):
        turtle.forward(size)
        turtle.left(90)

mysize = 100
drawSquare(mysize)
|
```



KEY CONCEPTS

- ✓ • variables
- ✓ • truth statements
- ✓ • looping
- ✓ • functions
- I/O
- lists
- classes/objects
- OOP



homework:

Create a better snowman and read chapters 1-6 of the online resource:

<https://runestone.academy/runestone/books/published/thinkcspy/index.html>

You may start on the quilting exercise (E1) but it is not expected

Instructions on the link on the course notes