Discover Python! Programming Virtual Turtles to draw Computer Graphics workshop

FAQS

Q: What is trinket.io?
A: Just one of many (partly) free online platforms you can learn to code various computer languages on!

Q: Why are we using trinket for today's workshop?
A: Setting up a local Python development environment e.g. on your personal computer/workstation, takes a bit of time & effort, mainly because the requirements typically vary significantly depending on factors like your operating system (Windows vs Mac vs Linux), which version of Python you want to use (e.g. Python 2 vs Python 3), and many more! It's not *hard*, there are thousands of free instructions online if you search for your specific circumstances, but using a standard, ready-to-code environment like trinket means everyone has the same set-up & can follow the same instructions for maximum learning!

Q: What are these Interactive vs Script modes for running Python code?
A: When programming in Python are two basic ways to run code:

1) **Interactive mode**, where you use a Python console (also known as Python interpreter or Python shell), an interactive programming command line tool that can be accessed from any local computer or server with Python installed. Specifically, it's a quick way to execute commands without creating a file (such as the 'main.py' file set-up in trinket), e.g. running single lines or blocks of code. *Note that the results of expressions are displayed in the console, whilst in 'script mode' they aren't and the 'print' statement is needed.*

2) **Script mode**, where you're writing more than a few lines of code, likely an actual program, and instead of running one line/block of code at a time, it's easier to write all your code in a single text file, also called a script, and then get the Python interpreter to run the entire file/code in one go! (e.g. how we're using the 'main.py' file set-up in trinket).

EXERCISES

EXERCISE A - Interactive Mode vs Script Mode

```
range(5)    # Note: this function does NOT include the end point

Learners =
Mentors =
Observers =
Total = Learners + Mentors + Observers
print(Total)
```

- Some arithmetic operators in Python: + | - | * | /
- Variables are objects that store a particular value in your program: <variable name> = <particular value>
- Python Standard Library - Built-in Functions, e.g. range(), print(): https://docs.python.org/3.6/library/functions.html

EXERCISE B - for loops

```
print("Hello World!")
```

```
for i in range(5):
    print(“YO World!”)
```

EXERCISE C - Python packages + random module

```
import random

random_integer = random.randint(0, 10)       # Note: this function DOES include the end point
print(random_integer)

colours = [ ]
random_colour = random.choice(colours)
print(random_colour)
```

- **PyPI - the Python Package Index**: repository of software for the Python programming language: https://pypi.python.org/pypi
- **Terminology**: Python packages are a collection, or directory, of Python modules - once you've selected & installed a package, you can 'import' individual modules from that package. For more details see official Python documentation: https://docs.python.org/3.6/tutorial/modules.html

EXERCISE D - turtle module

```
import turtle

turtle = turtle.Turtle()    # Make a turtle drawing robot
turtle.shape()              # Choose from "arrow", "turtle", "circle", "square", "triangle", "classic"
turtle.forward()
```

- Turtle module official Python documentation: https://docs.python.org/3.6/library/turtle.html

EXERCISE E - program turtle to draw triangle

```
length_of_side =
no_of_sides =
angle = 360/no_of_sides

turtle.forward(length_of_side)
turtle.right(angle)
```

EXERCISE F - program turtle to draw triangle using for loop

```
length_of_side =
no_of_sides =
angle = 360/no_of_sides

for i in range(no_of_sides):
    turtle.forward(length_of_side)
    turtle.right(angle)
```

EXERCISE G - program turtle to draw a firework

```
radius =
no_of_circles =
angle = 360/no_of_circles

for i in range(no_of_circles):
    turtle.circle(radius)
    turtle.right(angle)
```

EXERCISE H - program turtle to draw multiple fireworks in specific grid locations

```
turtle.goto(<x>, <y>)      # turtle operating on 2D grid/cartesian coordinates
```

EXERCISE I - program turtle to draw multiple fireworks in random grid locations

```
for i in range():
    turtle.penup()
    turtle.goto(random.randint(), random.randint())
    turtle.pendown()

    radius =
    no_of_circles =
    angle = 360/no_of_circles

    for i in range(no_of_circles):
        turtle.circle(radius)
        turtle.right(angle)
```

EXERCISE J - program turtle to draw varying-sized fireworks in random grid locations

```
for i in range():
    turtle.penup()
    turtle.goto(random.randint(), random.randint())
    turtle.pendown()

    radius = random.randint()
    no_of_circles = random.randint()
    angle = 360/no_of_circles

    for i in range(no_of_circles):
        turtle.circle(radius)
        turtle.right(angle)
```

EXERCISE K - program turtle to draw varying-sized + coloured fireworks in random grid locations

```
list_of_colours = []

for i in range():
    turtle.penup()
    turtle.goto(random.randint(), random.randint())
    turtle.pendown()
    turtle.pencolor(random.choice(list_of_colours))

    radius = random.randint()
    no_of_circles = random.randint()
    angle = 360/no_of_circles

    for i in range(no_of_circles):
        turtle.circle(radius)
        turtle.right(angle)
```

- Full list of 'named colors' in Python, e.g. 'navy', 'silver': http://www.tcl.tk/man/tcl8.5/TkCmd/colors.htm
- Python color swatches: http://www.wikipython.com/tkinter-ttk-tix/summary-information/colors/

EXERCISE L - optional add-ons - change pen size + set background colour

```
screen = turtle.getscreen()
screen.bgcolor()

turtle.pensize()
```

SOME VERY GENERAL RESOURCES - however, because Tech is such a broad field it's best to get personalised recommendations based on what you're specifically looking to do/learn

Free online textbook on the trinket website - Python for Everybody
https://books.trinket.io/pfe/

Free workshop - Learn Python & Django web framework with Django Girls non-profit org (March '18)
https://djangogirls.org/

Free Python tutorials from micro:bit
http://microbit.org/

One of Chiin's open source London-based tech communities for gender minorities - the AI Club
https://www.meetup.com/ai-club/

Website with range of free ebooks on different current tech skills + daily giveaway
https://www.packtpub.com/packt/free-ebook/

CHIIN CONTACT DEETS

Twitter @AnalyticsPanda | https://www.linkedin.com/in/chiin/ | chiin-rui@rhozeta.io