Section 3

May 6, 2021

Katherine Michel, Section Leader, Stanford Code in Place, 2021

Today’s Agenda

- Social Time (5 minutes)
- Recap/New (10 minutes)
- Running Total problem (10-15 minutes)
- FizzBuzz problem (15 minutes)
- Questions/comments/discussion (5 min)

Where We Are At

- This is the second week of Python
- Assignment 2 is due on Monday
- Diagnostic on Monday
- Lessons 1-8 have been published
- Tomorrow we start on images (super fun!)

Social Time

- Triumphs?
- Challenges?
- Doesn’t have to be about Code in Place!

Recap

- Control flow: comparison operators
- Control flow: “elif”
- Function definition/call
- Parameters and scope
- “print” versus “return”
- The building blocks of complex programs
- Real world: final project

Recap: Comparison Operators

Often used in if/while/else/elif statements to result in a Boolean value. Can be used to explicitly tell Python which choice to make.

Operator	Meaning	Example	Value
==	equals	if 1 == 1:	True
!=	does not equal	if 1 != 2:	True
<	less than	if 2 < 1:	False
>	greater than	if 2 > 1:	True
<=	less than or equal to	if 2 <= 1:	False
>=	greater than or equal to	if 2 >= 1:	True

Recap: “elif”

- “elif” is an abbreviation of “else if”
- Can use an unlimited number of “elif” statements

Using 2 “if” statements… not wrong

```
if user_input == 1:
    try_again()
if user_input == 2:
    print(“You win!”)
else:
    print(“You lose!”)
```

Better style/more Pythonic

```
if user_input == 1:
    try_again()
elif user_input == 2:
    print(“You win!”)
else:
    print(“You lose!”)
```

Recap: Function Definition/Call

```
def main():
    added = add_one(5)
    print(total)

def add_one(num):
    total = num + 1
    return total
```

Function call

Function definition

Recap: Parameters and Scope

```
def main():
    added = add_one(5)
    print(total)

def add_one(num):
    total = num + 1
    return total
```

Global scope… variables can be passed into helper functions as parameters

Local scope… in order for a variable contained within it to be used within the main body of the program, it has to be returned

Recap: “print” Versus “return”

```
def main():
    added = add_one(5)
    print(total)

def add_one(num):
    total = num + 1
    return total
```

This will print in the terminal… you will typically use the “print” statement to debug

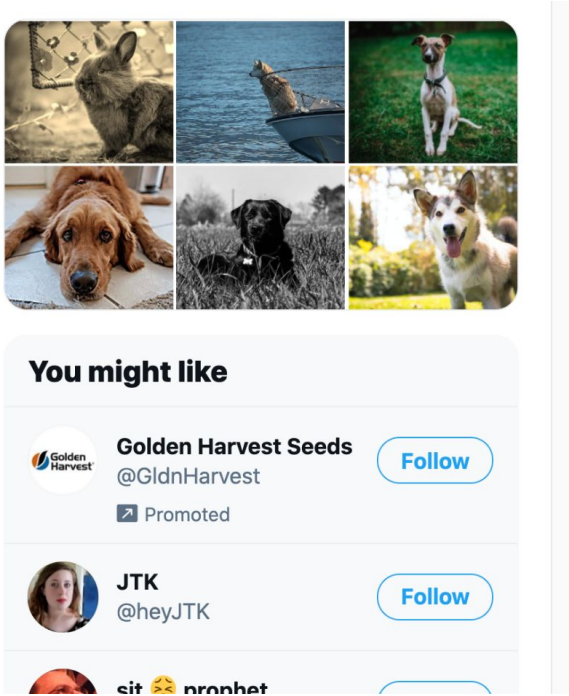
This will not print… to be used within the main body of the program, it has to be returned

The Building Blocks for Complex Programs

- Deconstruction
- Helper functions
- Scope
- Parameters
- Variables
- Python built-in library functions
- Third-party libraries

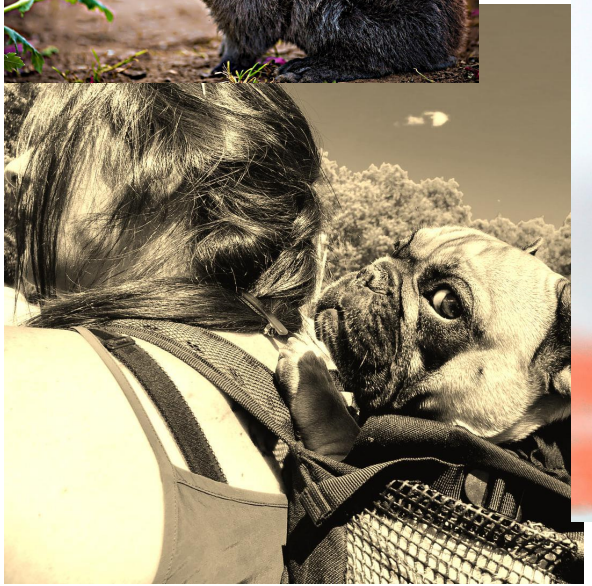
Katherine Michel, Section Leader, Stanford Code in Place, 2021

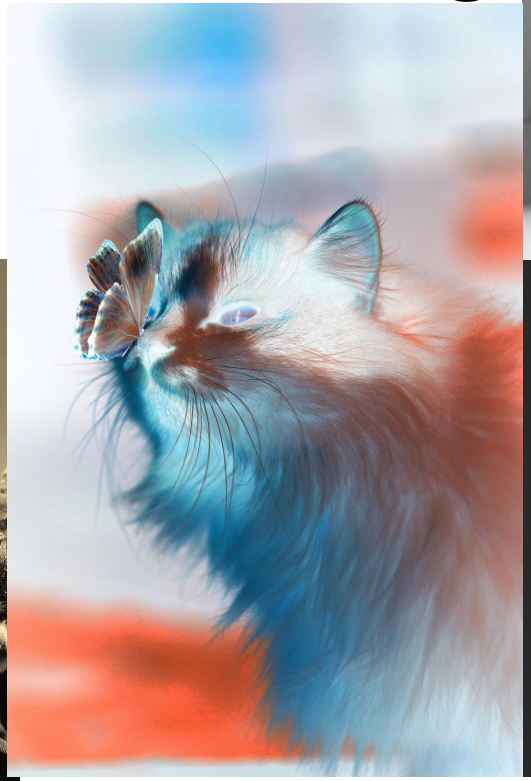
Real World: Program Result

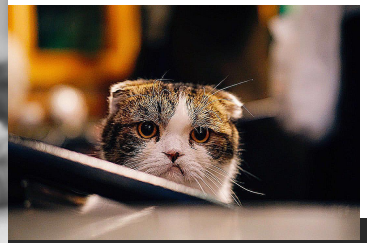


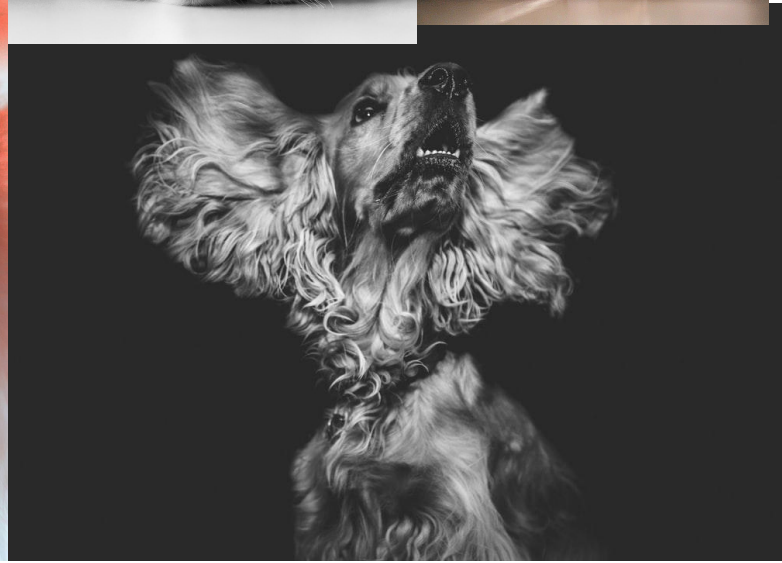
Katherine Michel, Section Leader, Stanford Code in Place, 2021

Real World: Program Result









Katherine Michel, Section Leader, Stanford Code in Place, 2021

Real World: Final Project

Using random and a third-party image library

"main" function

Choosing a random number to identify a random photo

```
import random
from PIL import Image, ImageFilter
# Other third-party libraries

def main():
    file_number = random.randrange(1, 278)
    file_url = ("https://source.unsplash.com/collection/2489501/" + str(file_number) + "/")
    file_path = str(file_number) + "." + "jpg"

# Download and save the photo file… a "new_image" variable is created
# Use random library again to randomly choose an algorithm from a list of algorithms
```

"Type casting" integer into string to create photo url and file path

Real World: Final Project

Helper functions/function definitions

Parameters from main/global scope

```
def black_and_white_algorithm(new_image, file_path):
    modified_image = new_image.convert("L")
    save_image(modified_image, file_path)

def save_image(modified_image, file_path):
    modified_image.save(file_path)
    return modified_image

# Return “modified_image” and tweet modified photo

if __name__ == "__main__":
    main()
```

Function call

Returning “modified_image” to main/global scope

“main” function is called

Answer Question from Last

What happens if random number does not match and if statement… program exits

This is something we normally want to avoid, but this is theoretical

8ball.py

```
This code corresponds to milestone 1 of the eight ball problem.
"""
answer_number = random.randint(6, 7)
if answer_number == 1:
    print(ANSWER_1)
if answer_number == 2:
    print(ANSWER_2)
if answer_number == 3:
    print(ANSWER_3)
if answer_number == 4:
    print(ANSWER_4)
if answer_number == 5:
    print(ANSWER_5)
```

/home/8ball.py 24:40 Spaces: 4 (Auto) All changes saved

Console Terminal Run Submit

```
Ask a yes or no question: Is Karel married?
```

Program exited with code 0

Randomly choose 6 of 7

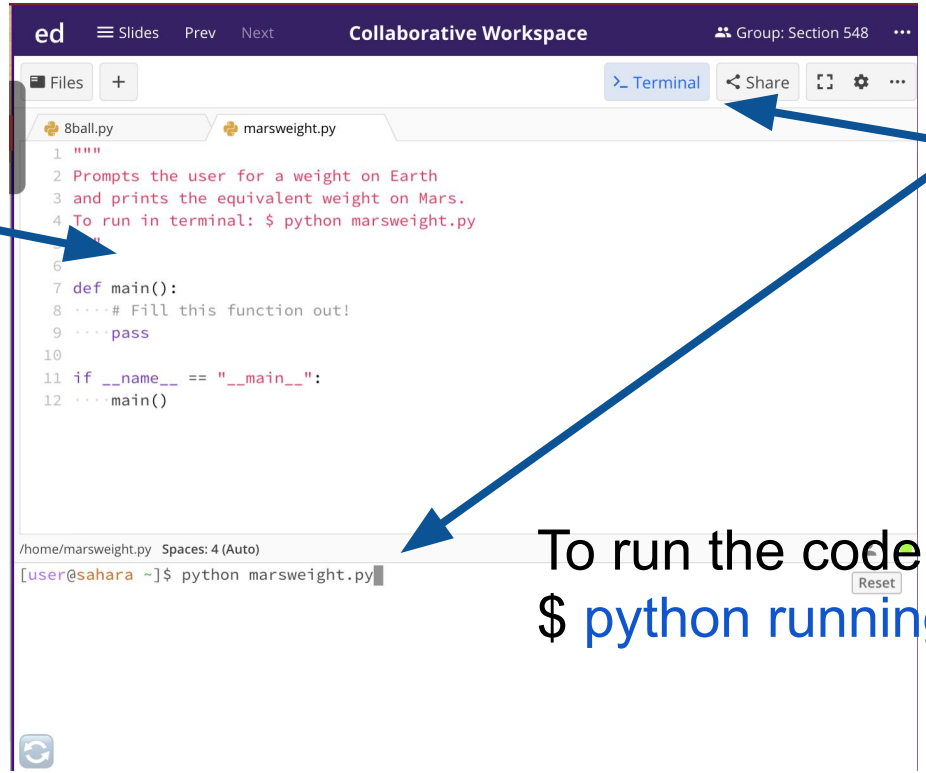
if statements only match 1, 2, 3, 4, or 5

Program exits

Katherine Michel, Section Leader, Stanford Code in Place, 2021

Collaborative Workspace

Everyone can see the same code and edit collaboratively



The “Terminal” button can be used to open a terminal…

To run the code type file name
$ python running.py

Katherine Michel, Section Leader, Stanford Code in Place, 2021

Running Total Problem

```
Enter a value: 7
Running total is 7

Enter a value: 3
Running total is 10

Enter a value: 5
Running total is 15

Enter a value: 12
Running total is 27

Enter a value: 0
```

Write a program that **asks a user to continuously enter numbers** and **print out the running total** (the sum of all the numbers so far).

Once you get the program working, see if you can modify it so that **the program stops when the user enters a 0**.

FizzBuzz Problem

```
Number to count to: 17
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
Fizzbuzz
16
17

Num fizzed: 4
Num buzzed: 2
Num fizzbuzzed: 1
```

In the game Fizz Buzz, players take turns counting up from one. If a player's turn lands on a number that's **divisible by 3, she should say fizz** instead of the number, and if it lands on a number that's **divisible by 5, she should say buzz** instead of the number. **If the number is both a multiple of 3 and of 5, she should say fizzbuzz** instead of the number.

It is an interesting problem in control flow and parameter usage. **Write a program that asks the user for an integer. The program should count up until and including n, fizzing and buzzing the correct numbers along the way. Once it's done, the program should print how many numbers were fizzed, buzzed, or fizzbuzzed along the way.**

Katherine Michel, Section Leader, Stanford Code in Place, 2021

Closing… Thoughts, Questions, Discussion?