

Michael Stanco

3/17/21

Python Primer Assignment #2

Questions

Basics

1. What is an expression?

Expressions are representations of value, as opposed to statements, which do something.

2. What is a syntax error?

Syntax errors are the most basic type of error. They arise when the Python parser is unable to understand a line of code

3. What is PEP8?

PEP 8, sometimes spelled PEP8 or PEP-8, is a document that provides guidelines and best practices on how to write Python code.

4. What does a linter do?

Linting highlights syntactical and stylistic problems in your Python source code

5. What is the result of this expression: `"*" * 10`

6. What is CPython?

CPython is the implementation of the language called "Python" in C.

7. How is CPython different from Jython?

Jython is written using Java instead. It uses JVM implementation.

8. How is CPython different from IronPython?

IronPython is an open source implementation of Python for the .NET CLR and Mono

Primitive Types

1. What is a variable?

A Python variable is a reserved memory location to store values. In other words, a variable in a python program gives data to the computer for processing.

2. What are the primitive built-in types in Python?

Integer, float, boolean, string

3. When should we use `"""` (triple quotes) to define strings?

Python's triple quotes comes to the rescue by allowing strings to span multiple lines, including verbatim NEWLINES, TABs, and any other special characters, such as quotations themselves.

4. Assuming `(name = "John Smith")`, what does `name[1]` return?

o

5. What about `name[-2]`?

t

6. What about `name[1:-1]`?

John Smit

7. How to get the length of `name`?

`len(name)`

8. What are the escape sequences in Python?

Backslash (\) Single quote (') Double quote (") newline (\n) Backspace(\b)

9. What is the result of `f“{2+2}+{10%3}”`?

4 + 1

10. Given `(name = “john smith”)`, what will `name.title()` return?

John Smith

11. What does `name.strip()` do?

`strip()` removes spaces at the beginning or end of the string

12. What will `name.find(“Smith”)` return?

Finds the index of where the specified string starts. Assuming `name = “john smith”`, itll return 5.

13. What will be the value of `name` after we call `name.replace(“j”, “k”)`?

Assuming `name = “john smith”`, we will get “kohn smith”

14. How can we check to see if `name` contains “John”?

“John” in name

15. What are the 3 types of numbers in Python?

Int, float, complex

Control Flow

1. What is the difference between `10 / 3` and `10 // 3`?

`//` is integer division

`10/3 = 3.33`

`10//3 = 3`

2. What is the result of `10 ** 3`?

10 to the power of 3 = 1000

3. Given `(x = 1)`, what will be the value of after we run `(x += 2)`?

`1 += 2 = 3`

The answer will be 3.

4. How can we round a number?

`Round(number/var,decimal places)`

5. What is the result of `float(1)`?

Turns to float type - 1.0

6. What is the result of `bool(“False”)`?

False

7. What are the false values in Python?

`bool(False)` `bool(None)` `bool(0)` `bool(“”)` `bool(())` `bool([])` `bool({})`

8. What is the result of `10 == “10”`?

True, if types match

9. What is the result of `“bag” > “apple”`?

False

10. What is the result of `not(True or False)`?

True or false is true, so not true or false is false

11. Under what circumstances does the expression `18 <= age < 65` evaluate to

True?

If age is between 18 inclusive and 65 exclusive

12. What does `range(1, 10, 2)` return?

1,3,5,7,9

13. Name 3 iterable objects in Python.

Dict, tuple, list

Functions

1. What is the difference between a parameter and an argument?

Function parameters are the names listed in the function's definition. Function arguments are the real values passed to the function. Parameters are initialized to the values of the arguments supplied.

2. All functions in Python by default return ...?

int

3. What are keyword arguments and when should we use them?

Keyword arguments (or named arguments) are values that, when passed into a function, are identifiable by specific parameter names

4. How can we make a parameter of a function optional?

Using the *args parameter

5. What happens when we prefix a parameter with an asterisk (*)?

When defining a function, the * operator can be used to capture an unlimited number of positional arguments given to the function.

6. What about two asterisks (**)?

The ** will give you all keyword arguments except for those corresponding to a formal parameter as a dictionary.

7. What is scope?

Variables can only reach the area in which they are defined, which is called scope... depends on the access modifiers for your variables

8. What is the difference between local and global variables?

Local var - inside of a function

Global var - outside of a function

9. Why is using the **global** statement a bad practice?

They are bad practice because they can enable side effects.

Coding Exercises

1. Write a function that returns the maximum of two numbers.

2. Write a function called **fizz_buzz** that takes a number.

1. If the number is divisible by 3, it should return "Fizz".

2. If it is divisible by 5, it should return "Buzz".

3. If it is divisible by both 3 and 5, it should return "FizzBuzz".

4. Otherwise, it should return the same number.

3. Write a function for checking the speed of drivers. This function should have one parameter: speed.

1. If speed is less than 70, it should print "Ok".

2. Otherwise, for every 5km above the speed limit (70), it should give the driver one demerit point and print the total number of demerit points. For example, if the speed is 80, it

should print: "Points: 2".

3. If the driver gets more than 12 points, the function should print: "License suspended" 4. Write a function called **showNumbers** that takes a parameter called **limit**. It should print all the numbers between 0 and limit with a label to identify the even and odd numbers. For example, if the limit is 3, it should print:

- 0 EVEN
- 1 ODD
- 2 EVEN
- 3 ODD

5. Write a function that returns the sum of multiples of 3 and 5 between 0 and **limit** (parameter). For example, if limit is 20, it should return the sum of 3, 5, 6, 9, 10, 12, 15, 18, 20. 6. Write a function called **show_stars(rows)**. If **rows** is 5, it should print the following: ○ *

- **
- ***
- ****
- *****

7. Write a function that prints all the prime numbers between 0 and **limit** where limit is a parameter.

Answers*****

#1. Write a function that returns the maximum of two numbers.

```
def return_max(a,b):  
    maximum = max(a,b)  
    print(maximum)  
    return maximum  
return_max(5,23)
```

#2. Write a function called **fizz_buzz** that takes a number.

#-If the number is divisible by 3, it should return "Fizz".

#-If it is divisible by 5, it should return "Buzz".

#-If it is divisible by both 3 and 5, it should return "FizzBuzz".

#-Otherwise, it should return the same number.

```
def fizz_buzz():  
    number = int(input("Please enter a integer for Fizz_buzz function:"))  
    if number % 3 == 0:  
        print("fizz")  
    if number % 5 == 0:  
        print("buzz")  
    if (number % 5 == 0) & (number % 3 == 0):  
        print("fizzbuzz")  
    else:
```

```
    print(number)
fizz_buzz()
```

#3. Write a function for checking the speed of drivers. This function should have one parameter: speed.
#- speed is less than 70, it should print "Ok".
#-Otherwise, for every 5km above the speed limit (70), it should give the driver one demerit point and print the total number of demerit points.
For example, if the speed is 80, it should print: "Points: 2".
#-If the driver gets more than 12 points, the function should print:

```
    "License suspended" 4
def speed_drivers(speed):
    if speed < 70:
        print("OK")
    else:
        temp_num = speed - 70
        i = 0
        points = 0
        while True:
            i += 5
            if i <= temp_num:
                points += 1
            else:
                print(f"Points: {points}")
                Break
        if points >= 12:
            print("License suspended")

speed = int(input("Please enter speed of car:"))
speed_drivers(speed)
```

#4 Write a function called showNumbers that takes a parameter called limit. It should print all the numbers between 0 and limit with a label to identify the even and odd numbers. For example, if the limit is 3, it should print:

```
#-0 EVEN
#-1 ODD
#-2 EVEN
#-3 ODD
```

```

def showNumbers(limit):
    for i in range(limit):
        if i % 2 == 0:
            print(f"{i} EVEN")
        else:
            print(f"{i} ODD")
limit = int(input("Please enter integer limit to show even/odd numbers:"))
showNumbers(limit)

```

#5. Write a function that returns the sum of multiples of 3 and 5 between 0 and limit (parameter). For example, if limit is 20, it should return the sum of 3, 5, 6, 9, 10, 12, 15, 18, 20.

```

def sum_multiples(limit):
    sum_limit = 0
    for i in range(0,limit):
        i+=1
        if i % 3 == 0:
            sum_limit += i
        elif i % 5 == 0:
            sum_limit += i

    print(f"Sum of multiples of 3 and 5 between 0 and {limit}:
    {sum_limit}")

limit = int(input("Please enter integer limit to sum multiples of:"))
sum_multiples(limit)

```

#6. Write a function called show_stars(rows). If rows is 5, it should print the following: o *

```

# **
# ***
# ****
# *****

def show_stars(rows):
    stars = ""
    for i in range(rows):
        stars = stars + "*"
    print(stars)

```

```
rows = int(input("Please enter number of rows to show stars for:"))
show_stars(rows)
```

#7. Write a function that prints all the prime numbers between 0 and limit where limit is a parameter.

```
def print_prime(limit):
    for i in range(limit + 1):
        # all prime numbers are greater than 1
        if i > 1:
            for j in range(2, i):
                if (i % j) == 0:
                    break
            else:
                print(i)
```

```
limit = int(input("Please enter a number to show prime numbers for:"))
print_prime(limit)
```