Programming Club Meeting 10 Slides

...

F-Strings and ‘Count Even and Odd’

F-String Review

- Easy way to add other variables into a string
- Precede first quotation mark with and an ‘f’
- Easiest way to make complex strings in Python 3

Code

```
1 # Brief F-String Rvw
2 age = 19
3 name = "John Doe"
4 print(name + " is " + str(age) + " years old.")
5 print(f"{name} is {age} years old.")
```

Output

```
John Doe is 19 years old.
John Doe is 19 years old.
```

Practice Problem:

Count Even and Odd

- Src: https://www.geeksforgeeks.org/python-program-to-count-even-and-odd-numbers-in-a-list/
- Goal: Write a Python program that will count and output the number of even and odd numbers in a list. You can assume lists only include integer numbers.
- Ex:
  - List in - [2, 7, 5, 64, 14]
    - Output - Even = 3, Odd = 2
  - List in - [12, 14, 95, 3]
    - Output - Even = 2, Odd = 2

Modules

Module Basics

- AKA library
- Set of functions and variables (usually constants) that you can use in your code
- Can use this to separate your code in different files
- Allows code to be shared easily
- Just like functions, allows the creation of self-contained things that can be easily tested by themselves and copied to many programs
- Many organizations have their own internal libraries that they add to all programs
    - My one professor created programs for a company that did plumbing stuff, so their library had a few circle related functions and a pi constant

Custom Modules

- Have a Python file with functions / constants in it
- Use import to “copy” all of the code into yours
- To get functions / constants, use dot naming scheme

exModule.py > ...

```
1  # Named Constants
2  num1 = 12
3  num2 = 45
4
5  # Ex Function
6  def foo() -> str:
7      return "bar"
```

```
# Example Custom Module
import exModule
print(exModule.foo())
print(exModule.num2)
```

Output ->

```
bar
45
```

Keywords

- Keep imports at the top of program
  - Want them to be easily findable
  - Imported functions / constants can only be used after the import statement
- Use “from” to import one function or constant (don’t need dot)
- Rename module to alias with “as”
- “dir” function lists all functions and variables in module

```
# Keywords
from exModule import num1
print(num1)
import math as m
print(m.pi)
print(dir(exModule))
```

v Output

```
12
3.141592653589793
['__builtins__', '__cached__', '__doc__', '__file__', '__loader__', '__name__', '__package__', '__spec__', 'foo', '
num1', 'num2']
```

Common Libraries

Quick Note

- Each function has a ton of functions and variables, I only included the ones I felt were most useful
  - To see all of them, look at the documentation
- “File” is a common library, but we will cover it in a future meeting

Time

```
# Time Module
import time
print(time.perf_counter())
time.sleep(2)
print(time.time())
```

v Output

```
0.018457993
1673483247.871295
```

‘datetime’ is a module with many similar features that also includes the ability to work with days easier

Math

Code

```
# Math Module
import math
print(math.ceil(0.4))
print(math.fabs(-83))
print(math.factorial(3))
print(math.floor(0.8))
print(math.log(23))
print(math.log10(10_000))
print(math.sqrt(16))
print(math.sin(30))
# Also has cos, tan, asin, acos, atan
print(math.degrees(math.pi))
print(math.radians(180))
print(math.pi)
print(math.e)
print(math.inf)
```

Output

```
1
83.0
6
0
3.13549421592915
4.0
4.0
-0.9880316240928618
180.0
3.141592653589793
3.141592653589793
2.718281828459045
inf
```

Random

Code

```
1 # Random Module
2 import random
3 random.seed(3)
4 print(random.randint(0, 99))
5 options = ["a", "b", "c"]
6 print(random.choice(options))
7 random.shuffle(options)
8 print(options)
9 print(random.random())
```

Output

```
55
c
['c', 'b', 'a']
0.8929469543476547
```

External Libraries

- Can use pip to download modules from the internet
- Numpy is a common library with more math functions, won't need it unless the program is very mathematical
    - Often shortened with as to np

Practice

Practice Problem 1: JT Module

- Goal: Write a module for the Jim Thorpe High School which has several variables holding information about the school. Also write a program that will output this data.
- Relevant Information:
  - You will need to use VS Code to work from multiple files
  - Entitle the file “jtahsInfo.py”
  - Make variables to include the following:
    - Jim Thorpe Area High School
    - Jim Thorpe Area School District
    - 1 Olympian Way, Jim Thorpe, PA
    - (570) 325-3663

Practice Problem 2:

Random Number Guessing Game

Goal: Write a Python program that will pick a random whole number from 1 to 100 (inclusive) before allowing the user to guess the number until they are correct. The program should keep track of the number of guesses (which will be outputted when the number is correctly guessed) and tell the user whether they guessed too high or too low after each guess.

Practice Problem 3: Random Values

- Src: https://www.w3resource.com/python-exercises/modules/index.php
- Goal: Write a Python that will output a random RGB value, alphabetical string, and a random multiple of 7 between 0 and 70 (inclusive).
- Relevant Information:
  - The RGB value should be a list holding 3 numbers from 0 to 255 (inclusive)
  - The alphabetical string should be 10 characters long

Next Meeting: Basic Algorithms