Python Exercises

Jason Antigua

Question 1:

print("Exercise 1: ");

print("\na. 5 / 3 = ",(5/3));

print("This method prints 1.67, which is the true value of 5/3, even though" +

" we did not set the values to float");

print("\nb. 5 % 3 = ", (5%3));

print("This prints out 5 mod 3, which is the remainder of when" +

" you do the operation 5 divdied by 3");

print("\nc. 5.0 / 3 = ", (5.0/3));

print("This also prints out the real value of 5/3, but instead it did a float value of 5.0" +

" divided by an integer value of 3.");

print("\nd. 5 / 3.0 = ", (5/3.0));

print("This again prints out the value of 5/3, but instead it turned 3 into a float valie," +

" showing it doesn't matter which of the two values are floats or integer it will still" +

" print the correct value.");

print("\ne. 5.2 % 3 = ", (5.2%3));

print("This prints out 5.2 mod 3, which is the remainder of when" +

" 5.2 gets divided by 3, which also shows you can do the remainder operation" +

" when two values are of different types, because 5 is a float and 3 is an integer.");

Question 2:

print("Exercise 2: ");

print("\na. 2000.3 \*\* 200 = overflow error");

print("This is supposed to print out 2000.3 to the power of 200 but" +

" there is an overflow error");

print("\nb. 1.0 + 1.0 - 1.0 = ", (1.0 + 1.0 - 1.0));

print("This adds to floating values of one which is 2.0" +

" and then subtracts a 1.0 from it");

print("\nc. 1.0 + 1.0e20 - 1.0e20 = ", (1.0 + 1.0e20 - 1.0e20));

print("This prints out 0, which is weird I would've assumed the output would be 1.0");

Question 3:

print("Exercise 3: ");

print("\na. float(123) = ", float(123) );

print("This turns 123 into a float type and then prints it out.");

print("\nb. float('123') = ", float('123') );

print("This turns the input of '123' into a float and prints the actual value of 123 without the apostrophies");

print("\nc. float('123.23') = ", float('123.23'));

print("This turns the input of '123.23' into a float and prints it out wihout the apostrophies");

print("\nd. float(123.23) = ", float(123.23));

print("This turns the input 123.23 into a float type and then prints it out");

print("\ne. int(123.23) = ", int(123.23));

print("This turns the input 123.23 into an int type and then prints out the value 123 instead since ints don't have decimals.");

print("\nf. int(float('123.23')) = ", int(float('123.23')));

print("This turns the input '123.23' into a float first and then into an int after, and then prints out the value which is 123.");

print("\ng. str(12) = ", str(12));

print("This turns the input 12 into a string type and prints it out, so on screen the 12 is of type int now");

print("\nh. str(12.2) = ", str(12.2));

print("This tunrs the input 12.2 into a string type and then proceeds to print it");

print("\ni. bool('a') = ", bool('a'));

print("This turns a into a boolean type and then it prints out True meaning the it's a true bool type");

print("\nj. bool(0) = ", bool(0));

print("This turns 0 into a boolean type and then printed out false, most likely because 0 always represents false and 1 represents true");

print("\nk. bool(0.1) = ", bool(0.1));

print("This turns the value 0.1 into boolean type and then printed out true");

Question 4:

print("Exercise 4: ");

print("\nThis is what gets outputted when executing range(5): ", range(5));

print("\nWhat i in range(5) means that i is true for any value in bewteen 0 to 5");

print("\nWe will execute (range(5)) to get a better understanding of the object: ", type(range(5)));

print("\nWhat gets printed out is <class 'range'>, which tells us range() command is of class range");

Question 5:

print("Exercise 5: ");

print("\nThis exercise finds the first 20 numbers that are" +

" divisible by 5, 7 & 11 and prints them");

def firstTwenty():

number\_found = 0;

x = 11;

while number\_found < 20:

divide\_five = (x % 5 == 0);

divide\_seven = (x % 7 == 0);

divide\_eleven = (x % 11 == 0);

if divide\_five and divide\_seven and divide\_eleven == True:

print(x);

number\_found += 1;

x += 1;

print("\nThe first 20 numbers are: ");

firstTwenty();

Question 6:

print("Exercise 6: ");

print("Part A");

def is\_prime(n):

if n <= 1:

return False;

else:

for p in range(2, n):

if n % p == 0:

return False;

return True;

print("\nTest Values:");

print("Is 0 prime? : ", is\_prime(0));

print("Is 7 prime? : ", is\_prime(7));

print("Is 24 prime? : ", is\_prime(24));

print("Is 41 prime? : ", is\_prime(41));

print("Is 53 prime? : ", is\_prime(53));

print("Is 100 prime? : ", is\_prime(100));

print("\nPart B");

def is\_prime(n):

if n == 2 or n == 3:

return True;

if n % 2 == 0 or n % 3 == 0:

return False;

if n <= 1:

return False;

j = 5;

k = 2;

while j \*\* 2 <= n:

if n % j == 0:

return False;

j += k;

k = 6 - k;

return True;

print("\nTest values:");

print("Is 2 prime? : ", is\_prime(2));

print("Is 14 prime? : ", is\_prime(14));

print("Is 23 prime? : ", is\_prime(23));

print("Is 47 prime? : ", is\_prime(47));

print("is 72 prime? : ", is\_prime(72));

print("\nPart C");

def primesUpTo(n):

x = 0;

while x <= n:

if is\_prime(x):

print(x);

x += 1;

print("\nTesting function primesUpTo(n)");

print("All prime numbers up to 45: ",);

print(primesUpTo(45));

print("\nPart D");

def firstPrimes(n):

total\_primes = n;

start\_point = 2;

while total\_primes > 0:

if is\_prime(start\_point):

total\_primes = total\_primes - 1;

print(start\_point);

start\_point += 1;

print("\nTesting function firstPrimes(n)");

print("First 10 prime numbers: ");

print(firstPrimes(10));

Question 7:

print("Exercise 7: ");

listOne = [1, 3, 5, 7, 9, 11]

listTwo = [2.2, 4.4, 6.6, 8.8, 10.10]

def printArray(list):

for x in range (len(list)):

print(list[x]);

print("\nPart A: ");

print("\nPrinting list One: ");

printArray(listOne);

print("\nPrinting list two: ");

printArray(listTwo);

def reverseArray(list):

for x in list[::-1]:

print(x);

print("\nPart B: ")

print("\nPrinting list one in reverse: ");

reverseArray(listOne);

print("\nPrinting list two in reverse: ");

reverseArray(listTwo);

def myLenFuntion(list):

countList = 0;

for x in list:

countList += 1;

return(countList);

print("\nPart C: ");

print("\nPrinting the length of list one: ", myLenFuntion(listOne));

print("\nPrinting the length of list two: ", myLenFuntion(listTwo));

Question 8:

print("Exercise 8: ");

print("\nPart A: ");

a = [5, 10, 15, 20];

print("\nPrinting the contents of list a: ", a);

print("\nPart B: ");

b = a;

print("\nWe set b = a, so what gets printed is: ", b);

print("Which is the exact same array as 'a'. ");

print("\nPart C & D: ");

b[1] = 25;

print("\nWe changed b[1] to equal 25, so we will print out the contents of both lists: ");

print("\nContents of list b: ", b);

print("\nContents of list a: ", a);

print("\nAs you can see the changes made to list b also affect the contents of list a.\nSo both of their index 1 values got changed.");

print("\nPart E: ");

c = a[:];

print("\nWe set c equal to a[:], and so we will print its content to see what it did. ", c);

print("\nPart F & G: ");

c[2] = 30;

print("\nWe made c[2] equal to 30, so we will print the list of a & c to see any changes: ");

print("\nContents of A: ", a);

print("\nContents of C: ", c);

print("\nPart H: ");

def set\_first\_elem\_to\_zero(l):

l[0] = 0;

return(l);

oList = [2, 4, 6, 8];

print("\nWe will be using a funtion to transform a list's first entry to 0.");

print("\nOriginal List: ", oList);

print("\nList after function: ", set\_first\_elem\_to\_zero(oList));

Question 9:

print("Exercise 9: ");

def merging\_sublists(list):

newList = [];

for i in list:

try:

i = int(i);

newList.append(i);

except:

for j in i:

newList.append(j);

return (newList);

listA = [[1, 2, 3], [4, 5]];

listB = [6, 12, 18, 24];

listC = [[1, 3], [3, 6]];

listD = [[10, 15, 20, 25], 30, 35];

print("\nList A: ", listA);

print("\nList A after funtion: ", merging\_sublists(listA));

print("\nList B: ", listB);

print("\nList B after function: ", merging\_sublists(listB));

print("\nList C: ", listC);

print("\nList C after function: ", merging\_sublists(listC));

print("\nList D: ", listD);

print("\nList D after function: ", merging\_sublists(listD));

Question 10:

import matplotlib.pyplot as plt

import numpy as np

print("Exercise 10: ");

x = np.arange(0.0, 2.0, 0.01);

y = np.power(np.sin(x-2),2) \* np.power(np.e,- (np.power(x,2)));

plt.xlabel("X-Axis");

plt.ylabel("Y-Axis");

plt.title("Function of sin^2(x-2)e^(-x^2)");

plt.xlim(0,2);

plt.ylim(0,1);

plt.plot(x, y);

plt.show();

Question 11:

print("Exercise 11: ");

list1 = [1, 2, 3, 4, 5];

l1 = len(list1);

list2 = [6, 7, 8, 9, 10];

l2 = len(list2);

def recursion(list, i):

if i == 0:

return (list[i]);

else:

return (list[i] \* recursion(list, i-1));

print("\nTesting two arrays for recursively getting the products of their elements");

print("\nContents of List 1: ", list1);

print("Product of List 1 using recursion: ", recursion(list1, l1-1));

print("\nContents of List 2: ", list2);

print("Product of the List 2 using recursion: ", recursion(list2, l2-1));

def iteration(list, i):

product = 1;

for x in range(i):

product = product \* list[x];

return product;

print("\nTesting two arrays for getting the products of their elements through iteration");

print("\nContents of List 1: ", list1);

print("Product of List 1 using iteration: ", iteration(list1, l1));

print("\nContents of List 2: ", list2);

print("Product of the List 2 using iteration: ", iteration(list2, l2));

Question 12:

print("Exercise 12: ");

def fibonacci(n):

if n <= 1:

return n;

else:

return fibonacci(n-1) + fibonacci(n-2);

print("\nTesting the fibonacci funtion:");

print("6th Fibonacci Number: ", fibonacci(6));

print("9th Fibonacci Number: ", fibonacci(9));

print("12th Fibonacci Number: ", fibonacci(12));

Question 13:

import re

print("Exercise 13: ");

file = open('emails.txt', 'r');

file = file.read();

emails = re.findall(r'([\w\S]+[@][a-z\S]+[.][a-z]+)', file);

print(emails);