Assignment 2 - Simple Types

Instructions

Follow the instructions for submitting a Jupyter Notebook assignment in the submitting assignments documentation.

1. Numeric Operations (20 Points)

```
a = 402
b = 1855

x = 41.151309
y = -95.919741
```

Given the preceding variable definitions, answer complete the questions below.

a. Compute the absolute value of y

b. Add x and y and multiple the result by a

c. Calculate the remainder leftover after dividing b by a (i.e. b/a)

d. Calculate a to the power of 3

e. Show how to convert a to a floating point number

f. Multiple x by y and round the result to two signficant digits

g. Compute the bitwise *or* of a and b

h. Compute x divided by negative y

i. Compute a added to b divided by x minus y

j. Compute the floored quotient of b and x

```
In [38]:  # define vars
          a = 402
          b = 1855
          x = 41.151309
          y = -95.919741
```

```
In [39]:  # a. Compute the absolute value of y
          abs(y)
```

```
Out[39]:  95.919741
```

```
In [40]: # b. Add `x` and `y` and multiple the result by `a`
         (x+y) * a
```

Out[40]: -22016.909664000003

```
In [41]: # c. Calculate the remainder leftover after dividing `b` by `a` (i.e. `b/a`)
         (b / a) - (b // a)
```

Out[41]: 0.6144278606965177

```
In [42]: # d. Calculate `a` to the power of `3`
         a ** 3
```

Out[42]: 64964808

```
In [43]: # e. Show how to convert `a` to a floating point number
         float(a)
```

Out[43]: 402.0

```
In [44]: # f. Multiple `x` by `y` and round the result to two signficant digits
         round(x * y, 2)
```

Out[44]: -3947.22

```
In [45]: # g. Compute the bitwise *or* of `a` and `b`
         a | b
```

Out[45]: 1983

```
In [46]: # h. Compute `x` divided by negative `y`
         x / (-y)
```

Out[46]: 0.4290181413229629

```
In [47]: # i. Compute `a` added to `b` divided by `x` minus `y`
         a + b / x - y
```

Out[47]: 542.9972864068789

```
In [48]: # j. Compute the floored quotient of `b` and `x`
         import math
         math.floor(b / x)
```

Out[48]: 45

2. Integer Division (2 Points)

What is the difference between dividing using the `//` operator and the `/` operator? For instance, what is the difference between `4/2` and `4//2`?

answer

4 / 2 - is classic and true division, keeps floating-point numbers 4 // 2 - is floor division, truncates fractional remainders to floor

3. Number Representations (4 Points)

Pick an integer number between 33 and 126. Print the following information about this number.

1. Its binary representation
2. Its hexadecimal representation
3. Its octal representation
4. The character corresponding to its Unicode point code.

```
In [49]:  # define var b/w 33 and 126
          rand = 100
```

```
In [50]:  # 3.1 Its binary representation
          print(bin(rand))
```

```
0b1100100
```

```
In [51]:  # 3.2 Its hexadecimal representation
          print(hex(rand))
```

```
0x64
```

```
In [52]:  # 3.3 Its octal representation
          print(oct(rand))
```

```
0o144
```

```
In [53]:  # 3.4 The character corresponding to its Unicode point code
          str(rand).encode('utf8')
```

```
Out[53]:  b'100'
```

4. Variable Assignment (4 Points)

Consider the following two Python code examples. In both cases, we assign a value to variable a, assign variable b to a and then make changes variable a. Why is it that in the first example, changes to a do not affect b, but in the second example they do?

Example 1:

```
In [1]:   a = 1
          b = a
          a += 1
          print(b)

Out [1]:  1
```

Example 2:

```
In [2]:   a = [1, 2, 3]
          b = a
          a.append(4)
          print(b)

Out [2]:  [1, 2, 3, 4]
```

answer

In the first example, the variable b is pointing to the object originally created by a (the number value of 1). In the second example, list's are mutable objects that support in-place changes. Both list a and b share the list object originally created, b is not a copy of a - it is the same.

5. Dynamic Typing (6 Points)

Static typing vs. dynamic typing is one of computer programmings most bitter "holy wars (http://wiki.c2.com/?HolyWar)'. As a data scientist, it is important to understand the difference between static and dynamic typing and the pros/cons of each approach.

Answer each of the following questions in your own words.

a. What is the difference between static and dynamic typing?

b. What are the benefits of static typing over dynamic typing?

c. What are the benefits of dynamic typing over static typing?

answers

A. Static typing requires variables to be declared when created (e.g., int, str, etc) while dynamic typic determines the type at runtime.

B. Static typing allows to determine the use and application of certain variables. For example, you have to intend to use the variable in a certain way instead of getting lucky or having uncertain results later on.

C. Dynamic typing is more flexible and is more applicable in more contexts.

6. Garbage Collection (4 Points)

a. Explain what *garbage collection* means in connection to programming languages.

b. How does CPython implement garbage collection?

answers

A. Garbage collection is a way for the programming language to clean up or free memory back to the program.

B. Whenever an object is no longer refereneced the objected is deleted from the memory pool to free up more space. CPython does this automatically as part of the languages built-in features. This is very helpful for programmers not to have to get super technical to clean up as they go.