Tuples

Announcements

- Homework #4 due tomorrow @ 11pm

Outline

- Mid-semester feedback
- Warm-up: references and aliasing
- Tuples
- Programming practice

Warm-up: references and aliasing

For each of the following programs, draw a reference diagram and determine the program’s output.

a) odds = [1, 3, 5]
   odds[0] = 7
   nums = [7, 3, 5]
   print(odds, nums)
   print(odds == nums)
   print(odds is nums)
   Output:
   [7, 3, 5] [7, 3, 5]
   True
   False

b) tens = [10, 20, 30]
   nums = tens
   nums[2] = 40
   print(nums, tens)
   print(nums == tens)
   print(nums is tens)
   tenslice = tens[:]
   tenslice[1] = 50
   print(tenslice, tens)
   print(tenslice == tens)
   print(tenslice is tens)
   Output:
   [10, 20, 40] [10, 20, 40]
   True
   True
   [10, 50, 40] [10, 20, 40]
   False
   False
c) first = ['a', 'b']
    second = ['c']
    first.append(second)
    second.append(['d'])
    print(first, second)
    print(first[2] == second)
    print(first[2] is second)
    first.append(['e'])
    third = ['d']
    print(first, third)
    print(first[3] == third)
    print(first[3] is third)

Output:
['a', 'b', ['c', ['d']]] ['c', ['d']]
True
True
['a', 'b', ['c', ['d']], ['e']][ 'd']
False
False

d) def myfunc(b):
    b[0] = 2
    return b

    a = [1, 3, 5]
    c = myfunc(a)
    print(a, c)
    print(a is c)

Output:
[2, 3, 5] [2, 3, 5]
True

Tuples

- An unmodifiable collection of values
- Created using parenthesis instead of square brackets
  - professor = ("Aaron", "McGregory", 310)
- Cannot assign to an index or slice of a tuple!
  - professor[2] = 314
    TypeError: 'tuple' object does not support item assignment
  - professor[1:1] = "Gember-Jacobson"
    TypeError: 'tuple' object does not support item assignment
- Can assign a tuple of values to a tuple of variables
  - (name, building, room) = ("Aaron", "McGregory", 310)
    print("Name: " + name)
    print("Office: " + building + " " + str(room))

Output:
Name: Aaron
Office: McGregory 310
- (name, building, room) = professor
The number of variables on the left side of the assignment statement must equal the number of values on the right side of the assignment statement

- (firstname, lastname, building, room) = ("Aaron", "McGregory", 310)
  ValueError: not enough values to unpack (expected 4, got 3)

- Use tuples to return multiple values from a function

  - def stats(nums):
    total = 0
    for num in nums:
      total += num
    cnt = len(nums)
    avg = total / cnt
    return (cnt, avg)

  (count, average) = stats([1,2,3,4,5])
  print("Count: " + str(count))
  print("Average: " + str(average))
  Output:
  Count: 5
  Average: 3.0

Programming practice

a) Write a function called unique_characters that returns a list of every distinct character that appears in a string. For example, unique_characters("mississippi") returns ['m', 'i', 's', 'p'].

def unique_characters(string):
  chars = []
  for char in string:
    if char not in chars:
      chars.append(char)
  return chars

b) Write a function called double that takes a list of numbers and doubles each number in the list.

def double(nums):
  for i in range(len(nums)):
    nums[i] = nums[i] * 2

c) Write a function called double_preserve that takes a list of numbers and returns a list of numbers where each number in the original is doubled. The list passed as a parameter must remain unchanged!

def double_preserve(nums):
  clone = nums[:]
  for i in range(len(clone)):
    clone[i] = clone[i] * 2
  return clone