Quiz Section 3

loops

2019-04-18

`Python` - loops.

Loops

> Loops allow code to be repeated

In `Python` there are two kinds of loops
- `for`
- `while`

Today we are going to talk about `for` loops.

`for` loops

`for` loops let you repeatedly apply the same code to all elements in list (roughly).

Example: print all items in a list

```
l = [4, 5, 6]
print(l[0])  # 4
print(l[1])  # 5
print(l[2])  # 6
```

```
l = [4, 5, 6]
for i in l:
  print(i)
```

General structure of a `for` loop

```
for <target> in <object>:
  <statement>
  <statement>
...
<statement>
```

notice the : and the indents

`for` loops and strings

```
DNA = 'ATG'
for base in DNA:
  print(base)
```

Example: length of list

```
counter = 0
for item in l:
  counter = counter + 1
print(counter)
```

Example

Take in a DNA string and print the sequence by base

input: `'AGTCGA'`

output:

```
base 0 is A
base 1 is G
base 2 is T
base 3 is C
base 4 is G
base 5 is A
```

Solution

```
index = 0
for base in DNA:
  print("base {0} is {1}".format(index, base))
  index = index + 1
```

Example: sum numbers in a list

input: 1, 2, 3, 4
output: 10

Solution

```
num_list = [1, 2, 3, 4]
sum = 0
for num in num_list
  sum = sum + num
print(sum)
```

`range` function

input: start, stop, step
output: iterator (functionally a list) of those numbers

`range([start,] stop [, step])`

- start and step are optional (default 0 and 1)
- negative step reverses

Range exercises

- print every number from 0 to 10 (`range(11)`)
- print every number from 1 to 10 (`range(1, 11)`)
- print every *even* number from 2 to 12 (`range(2, 13, 2)`)
- print every *third* number from 90 to 100 *backwards* (`range(100, 80, -3)`)

Nested loops

example: print all pairs of [A, T, C, G]

```
nts = ['A', 'T', 'C', 'G']
for i in nts:
  for j in nts:
    print(i,j)
```

Hamming Distance problem

Given two strings calculate the hamming distance.
input: `CATS HATS`
output: 1

Hamming distance solution

```
import sys

s1 = sys.argv[1]
s2 = sys.argv[2]

if len(s1) != len(s2):
    print("s1 and s2 are different lengths!")
else:
    dist = 0
    for i in range(len(s1)):
        if s1[i] != s2[i]:
            dist = dist + 1
    print("The hamming distance is {0}".format(dist))
```

Sample problem

Write a program `add-arguments.py` that reads any number of integers from the command line and prints the cumulative total for each successive argument.

```
> python add-arguments.py 1 2 3
1
3
6
> python add-arguments.py 1 4 -1
1
5
4
```

Solution

```
nums = sys.argv[1:] # all inputs as list

total = 0
for num in nums:
  total = total + num
  print(total)
```