Lists

by Deborah R. Fowler

KEY CONCEPTS

- variables
- truth statements
- looping
- functions
- I/O
- lists
- classes/objects
- OOP

Strings are lists

test3.py - D:/SCAD_ALL/BuildWebSite/SRCWebSite/PythonResources/programmingPDF/Class05-IO/test3.py (3.6.8)

File Edit Format Run Options Window Help

```
kermit = open("testdata.txt",'r')
for line in kermit:
    values = line.split()
    print(line)
    print(values[0] + values[1])

kermit.close()
```

print (values[0] + values[1])

would result in a
string that was **concatentated**

Python 3.6.8 Shell

File Edit Shell Debug Options Window Help

```
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 20
(AMD64)] on win32
Type "help", "copyright", "credits" or "license
>>>
 RESTART: D:/SCAD_ALL/BuildWebSite/SRCWebSite/I
ss05-IO/test3.py
10.6    11.5 40.6

10.611.5
30.0 50.6    50.0

30.050.6
10.0 50.8 45.7

10.050.8
>>>
```

test4.py - D:/SCAD_ALL/BuildWebSite/SRCWebSite/PythonResources/programmingPDF/Class05-IO/test4.py (3.6.8)

File Edit Format Run Options Window Help

```
kermit = open("testdata.txt",'r')
for line in kermit:
    values = line.split()
    print(line)
    print(float(values[0]) + float(values[1]))

kermit.close()
```

using float(argument) to convert

would result in 22.1

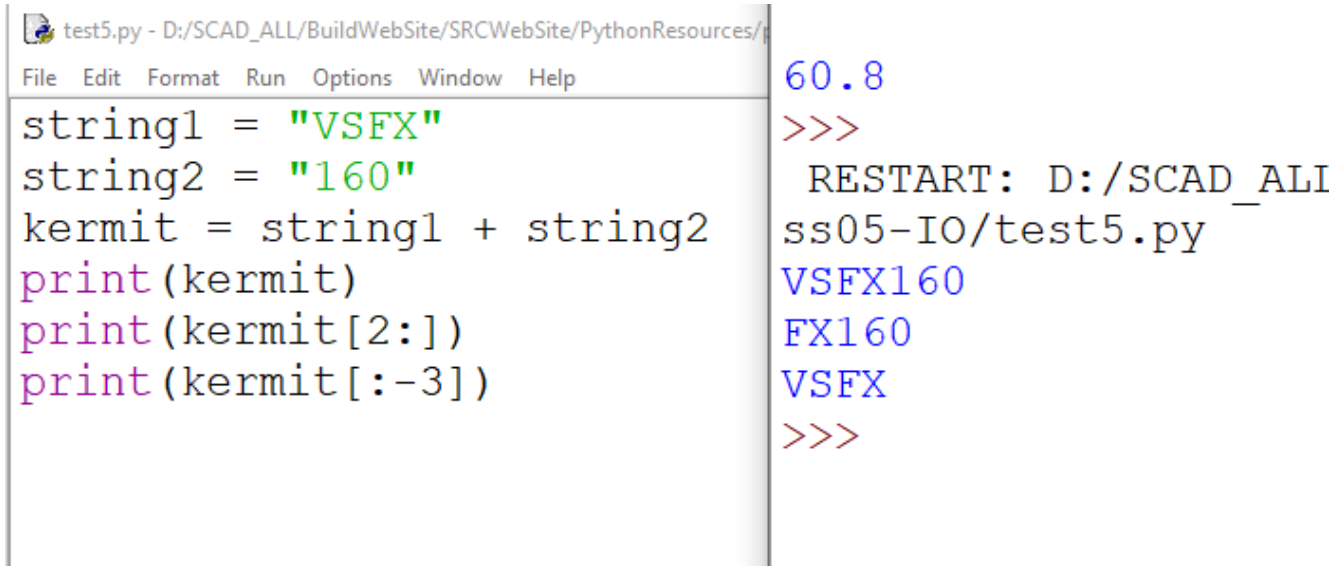
Python 3.6.8 Shell

File Edit Shell Debug Options Window Help

```
Python 3.6.8 (tags/v3.6.8:3c6b436a57,
(AMD64)] on win32
Type "help", "copyright", "credits" c
>>>
 RESTART: D:/SCAD_ALL/BuildWebSite/SR
ss05-IO/test4.py
10.6    11.5 40.6

22.1
50.0 50.6    50.0

80.6
10.0 50.8 45.7

60.8
>>>
```

Positive from left

Negative from right



rstrip

File Edit Format Run Options Window Help

```
kermit = open("testdataCommas.txt")
for line in kermit:
    print line
    modlist = line.rstrip().split(',')
    print float(modlist[0]) + float(modlist[1])


kermit.close()
```

In-class Exercise

Create a .txt file with a few lines of data, this time separate them with commas

Create a script to read the file and write out the second element of each line

Lists are defined by square brackets

[ ] is an empty list

ordered

changeable

Access an element or member of the list use an index (also called subscript)

mylist[1] will give you the second item on the list

remember to count from zero

Why use them?

Data structure that makes access easier,
For example the “for” loop can be used to iterate through a list

```
for item in mylist:
        print item
```

```
File  Edit  Shell  Debug  Options  Window  Help
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018, 00:16:47) [M
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more inf
>>> mylist = ["cavalier", "dog", "spaniel"]
>>> print(mylist[1])
dog
>>> for item in mylist:
        print(item)


cavalier
dog
spaniel
>>>
```

append

```
>>> mylist[4] = "king"
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    mylist[4] = "king"
IndexError: list assignment index out of range
>>> mylist.append("king")
>>> print(mylist)
['cavalier', 'dog', 'spaniel', 'king']
>>>
```

len

```
>>> print(mylist)
['cavalier', 'dog', 'spaniel', 'king']
>>> print(len(mylist))
4
>>>
```

Two other things python lists are capable of:

- inserting into the list at a given position
- differing types of items in a single list

insert

```
>>> mylist.insert(2,"charles")
>>> print(mylist)
['cavalier', 'dog', 'charles', 'spaniel', 'king']
>>>
```

remove

pop

del

```
>>> mylist.remove("dog")
>>> print(mylist)
['cavalier', 'charles', 'spaniel', 'king']
>>> mylist.pop()
'king'
>>> print(mylist)
['cavalier', 'charles', 'spaniel']
>>> mylist.pop(1)
'charles'
>>> print(mylist)
['cavalier', 'spaniel']
>>> del mylist[1]
>>> print(mylist)
['cavalier']
>>>
```

Mixed types

```
>>> print(mylist)
['cavalier']
>>> mylist.append(10.5)
>>> print(mylist)
['cavalier', 10.5]
>>> mylist.append(["this","is","another","list"])
>>> print(mylist)
['cavalier', 10.5, ['this', 'is', 'another', 'list']]
>>>
```

KEY CONCEPTS

- variables
- truth statements
- looping
- functions
- I/O
- lists
- classes/objects
- OOP

Homework:

Work on the Hurricane Exercise