ECS10

2/1

Announcements

- □ Next assignment up, checkpoint due Thursday 2/7.
- □ Midterm programs returned in Section next week, scores up soon.
- □ Most people missed 3-4 MC, could write loop but not find highest/lowest
- □ PythonAnywhere – Web interpreter

Lists

- □ [“Zucchini”, “340”] is a list of strings

```
days = [31,28,31,30,31,30,31,31,31,31,30,31]
```

- □ days is a list of integers
- □ [“Milk”, 120] is also a fine list.
- □ len(days) produces the value 12.
- □ days[0] produces the value 31.
- □ This is called indexing.
- □ days[3]?
- □ days[-1]?

Lists and strings are sequences

- □ The len() function works on sequences
- □ Indexing works with sequences
- □ Concatenation works on sequences

```
>>> [1,2]+[4,5]
[1, 2, 4, 5]
```

The in operator

- □ The in operator operator works with sequences

```
FarmAnimals = [“cow”,”goat”,”pig”]
if “pig” in FarmAnimals:
        print(“He has a pig”)

if not “I” in “team”:
        print(“There is no I in team”)
```

- □ Produces a Boolean value

while loop on a list

```
FarmAnimals = [“cow”,”goat”,”pig”]
i = 0
while i < len(FarmAnimals):
        print(“He had some”, FarmAnimals [ i ])
        i = i+1
```

- □ The variable i is an index variable; it picks out the words in the list one by one, starting at the beginning.
- □ Has to be initialized to zero, incremented each time through the loop.

Data is in files.

- Files are on your disk.
- Files are read sequentially, just like you read a novel: one page at a time, on each page one line at a time, on each line one word at a time, in each word, one character at a time.
- This is called streaming.
- In Python, we can see it at the line level (other ways to do it too…)
- Main idea: get data out of files, into strings and lists. Process with loops.

Opening a file

```
inFile = open("menu.txt","r")
# variable inFile refers to the file
# in our program. Data type: I/O source
print(type(inFile))
```

- open() function connects program to file. "r" means read, "w" means write.
- Returns a "handle" to the data in the file, an I/O source (I/O means input/output).
- Knows that it's a text file.

Reading a line

```
line = inFile.readline()
length = len(line)
print(line)
print("Contains",length,"characters")
```

- readline() method reads one line from the file.
- Method is a kind of function. First input is before the period, not inside parentheses.
- readline() is an I/O method
- line is a string.
- When called at end of file, returns empty string "".

String and file methods

- format is a string method we've seen before.

```
'{:.2f}'.format( 1.8976437521 )
```

- This is an expression
- format() is a function. One of its inputs is in the parens.
- The other input is the string describing the format, which is stuck onto the beginning, with the period inbetween
- readline() is a file method (file before period)

The newline character

```
line = inFile.readline()
lastChar = line[-1]
print( lastChar == "\n" )
```

- A line from a file ends with a newline character.
- In Python, we write newline as "\n".

Removing the newline

```
line = line.strip()
```

- strip() method Removes the newline from the end of the line.
- strip() is a string method. Input is the string before the period.
- Here, result is assigned back into same variable.
- Similar to x = x+1