



OMIS 30: Intro to Programming (with Python)

Week 4, Class 1

Introduction to Programming
Instructor: Denis Vrdoljak



Learn programming with Python



Office Hours

Instructor	Days available
Yuan Wang (our TA)	M 2:30-3:30p, W 9-10a
Mike Davis (other section's instructor)	Tu 9:30-10:20a, Th 12-1p
Denis Vrdoljak	Tu 3:40-4:40p, W 12-1p/4-5p (this week only)



Course Topics

- 1. Computer setup - intros
- 2. Shell - cd, mkdir, move, rename, copy, pwd, touch, echo, nano, vim
- 3. Grep, bash, scripts
- 4. Python Basics - print, input, math, PEMDAS
- 5. Pseudo-code, algorithm design, comments
- 6. Loops & Nested Loops
- 7. Moving files around, input, export
- 8. Dates, times, epoch, time-series
- 9. Arrays, lists, dicts, sets, etc.
- 10. And, or, If, elif, try, except
- 11. Functions
- 12. Strings, upper, lower, regex
- 13. Computation time, flops, sorting
- 14. JSON, dicts, iteritems
- 15. Pandas
- 16. Jupyter, virtual environments
- 17. Web-apps/web-pages
- 18. Web-scraping
- 19. Plotting, graphing
- 20. Git



Goals from last week

- Understand and be able to use While and For loops
- Know how to write a standalone Python script, and how to run it from the Command Line
- Make progress on Project 1!



By the end of this week you should:

- Get some more practice with flow control (if, while, for)
- Understand objects and dot notation
- Know what mutability is and understand mutable vs immutable objects
- Know what a function is and how to use and define a basic one
- **Finish Project 1!**



FOR Loops (Review)

- For loops iterate through an iterable.
- This can be a list, a string...or a set, a tuple, a dictionary, etc
- For loops stop once the last element in the iterable is used
- break and continue work with for loops, just like with while loops

```
barn = ['cat', 'dog', 'elephant', 'giraffe', 'pig']
for animal in barn:
    print(animal + " found in the barn")
```

output:

```
cat found in the barn
dog found in the barn
elephant found in the barn
giraffe found in the barn
pig found in the barn
```

In [27]:



FOR Loops (Review)

- For loops iterate through an iterable.
- This can be a list, a string...or a set, a tuple, a dictionary, etc
- For loops stop once the last element in the iterable is used
- break and continue work with for loops, just like with while loops

output:

```
for letter in "animal":  
    print(letter)
```

```
a  
n  
i  
m  
a  
l
```

In [29]:



SANTA CLARA UNIVERSITY

Homework review



Homework review

This is the Loop that never Ends...

Create a standalone python script (.py file, not a .ipynb notebook) that asks a user for an input, does something (eg, prints user's input) repeatedly in an infinite loop, but quits if the first letter of the user input is a 'q' or a 'Q'.

https://github.com/denisvrdoljak/OMIS30_Fall2018/blob/master/wk3/wk3hw1.md



Homework review

CSV Parser

```
data = """  
Zip Code,Total Population,Median Age,Total Males,Total Females,Total Households,Average Household Size\n  
90001,57110,26.6,28468,28642,12971,4.4\n  
90002,51223,25.5,24876,26347,11731,4.36\n  
90003,66266,26.3,32631,33635,15642,4.22  
"""\n.strip()
```

https://github.com/denisvrdoljak/OMIS30_Fall2018/blob/master/wk2/OMIS30_wk2hw1.ipynb



Questions?



Dot Notation and Methods



Object dot notation

- There are two things that can be attached to an object
 - 1) An attribute, this stores data
 - 2) A bound method, this is a function that does things to the object
- An **attribute** is connoted with dot notation and no parenthesis
 - For example this object stores its color as an **attribute**
`cat_object.color`
- A **method** is connoted with dot notation and has parenthesis
 - For example this object has the **method** meow
 - Calling a bound method makes the object DO SOMETHING
`cat_object.meow()`



Bound Methods

- We use the term method to refer to a function that is defined for a specific object type
- The syntax is different than a regular generic function
- Consider sorting a list

A regular function

- `sorted(list)` is a function applied to a list

A bound method

- `list.sort()` leverages the list type's bound method



The sorted function

The sorted function makes a new list that is accessible when it is bound to an output variable

```
1 # since this is a series we can use methods that rely on order
2 farm_count=[200, 5, 2, 2, 10]
3
4 print (farm_count)
5
6 SortedLs=sorted(farm_count)
7
8 print (farm_count)
9
10 print (SortedLs)
```

```
[200, 5, 2, 2, 10]
[200, 5, 2, 2, 10]
[2, 2, 5, 10, 200]
```



The `.sort()` bound methods

- The sort bound method is a method for list objects and it modifies the list

```
2  
3 #the sort method changes the object itself  
4 print ('the bound method changes the list itself')  
5 farm_count.sort()  
6 print(farm_count)  
7
```

```
the bound method changes the list itself  
[2, 2, 5, 10, 200]
```



Dot Notation Activity:

Duck Soccer

https://github.com/denisvradoljak/OMIS30_Fall2018/tree/master/DuckSoccer



Mutability: Lists and Strings Revisited



List Methods [Mutable]

- There are other list methods including
 - *list.min()* get the min value
 - *list.len()* get the length of the list
 - *list.max()* get the max value
 - *list.index(element)* get the first matching element
 - *list.pop(index)* remove element at index and return it to output
 - *list.insert(element, index)* insert the element at
 - *list.extend()* add elements to the list
 - *list.append()* append single element



More String Methods and Operations (review)

- `.upper()`, `.lower()`, `.capitalize()`, `.title()`
 - `mystring = 'Abc dEf'`
 - `mystring.upper() = 'ABC DEF'`
 - Uppercase all letters
 - `mystring.lower() = 'abc def'`
 - Lowercase all letters
 - `mystring.capitalize() = 'Abc def'`
 - Capitalizes first letter of first word
 - `mystring.title() = 'Abc Def'`
 - Capitalizes first letter of each word



String Methods [immutable]

- There are string methods including
 - `str.min()` get the min value
 - `str.len()` get the length of the list
 - `str.max()` get the max value
 - `str.rindex()` get the first matching element
 - `str.slice()` slice by index
 - `str.upper()` make the string uppercase
 - `str.lower()` make the string lowercase
 - `str.title()` make the string title case
 - `str.split(substring)` split on the specified substring

<https://www.programiz.com/python-programming/methods/string>



Questions?



SANTA CLARA UNIVERSITY

Libraries



Import Statement

- Importing libraries is done with the “import” statement
- Imports must be at the top of the script
- Example:
 - `import random`



Import Statement

- Importing libraries is done with the “import” statement
- Imports must be at the top of the script
- Example:
 - `import random`
- Objects within the library can be accessed with dot notation:
 - `random.choose(mylist)`



Import Statement

- Importing libraries is done with the “import” statement
- Imports must be at the top of the script
- Example:
 - `import random`
- Objects within the library can be accessed with dot notation:
 - `random.choose(mylist)`
- Objects can also be imported as an import from:
 - `from random import choose`
 - `choose(mylist)`



Import Statement

- Importing libraries is done with the “import” statement
- Imports must be at the top of the script
- Example:
 - `import random`
- Objects within the library can be accessed with dot notation:
 - `random.choose(mylist)`
- Objects can also be imported as an import from:
 - `from random import choose`
 - `choose(mylist)`

More on the Random library can be found here:

<https://docs.python.org/3/library/random.html>



Homework 4.1

- Finish Project 1



Appendix

- Reserved keywords in Python:
 - <https://pentangle.net/python/handbook/node52.html>
 - DO NOT USE THESE as VARIABLE NAMES!
- Built-in Functions:
 - <https://docs.python.org/3/library/functions.html>
 - Review/Reference Material:
 - <https://developers.google.com/edu/python/lists>