CompSci 101
Introduction to Computer Science

October 9, 2014

Prof. Rodger

Thanks to Prof. Azhar and Yossra Hamid for giving this lecture!
Announcements

• Reading for next time on calendar page
  – en.wikibooks.org/wiki/Python_Programming/Sets
  – RQ

• APT 4 is due today
  – APT 5 is out today

• Exam 1 was handed out Tuesday, grades are on Sakai, you will need to see Prof. Rodger next week to get your test back

• Today Sets

• Prof. Rodger is at a conference this week
  – http://gracehopper.org/
Python Sets

• Set – unordered collection of distinct items
  – Unordered – can look at them one at a time, but cannot count on any order
  – Distinct - one copy of each

• Operations on sets:
  – Modify: add, clear, remove
  – Create a new set: difference(-), intersection(&), union (|), symmetric_difference(^)
  – Boolean: issubset <=, issuperset >=

• Can convert list to set, set to list
Summary (from wikibooks)

- set1 = set()                   # A new empty set
- set1.add("cat")                # Add a single member
- set1.update(["dog", "mouse"])  # Add several members
- set1.remove("cat")             # Remove a member - error if not there
- print set1
- for item in set1:              # Iteration AKA for each element
    print item
- print "Item count:", len(set1)  # Length AKA size AKA item count
- isempty = len(set1) == 0       # Test for emptiness
- set1 = set(["cat", "dog"])    # Initialize set from a list
- set3 = set1 & set2             # Intersection
- set4 = set1 | set2             # Union
- set5 = set1 - set3             # Set difference
- set6 = set1 ^ set2             # Symmetric difference (elements in either
set but not both)
- issubset = set1 <= set2        # Subset test
- issuperset = set1 >= set2      # Superset test
- set7 = set1.copy()             # A shallow copy (copies the set, not the
elements)
- set8.clear()                   # Clear AKA empty AKA erase
Creating and changing a set

colorList = ['red', 'blue', 'red', 'red', 'red', 'green']
colorSet = set(colorList)
smallList = list(colorSet)
colorSet.clear()
colorSet.add("yellow")
colorSet.add("red")
colorSet.add("blue")
colorSet.add("yellow")
colorSet.add("purple")
colorSet.remove("yellow")

• See setsEasy.py
Set Operations

UScolors = set(["red", "white", "blue"])  
dukeColors = set(["blue", "white"])  
print dukeColors.union(UScolors)  
print dukeColors | UScolors  
print dukeColors.intersection(UScolors)  
print dukeColors & UScolors  
print dukeColors.difference(UScolors)  
print dukeColors - UScolors  
print UScolors - dukeColors  
print dukeColors ^ UScolors  
print UScolors ^ dukeColors

• See setsEasy.py
poloClub = set(['Mary', 'Laura', 'Dell'])

rugbyClub = set(['Fred', 'Sue', 'Mary'])

Question 1:
print [w for w in poloClub.intersection(rugbyClub)]

Question 2:
print [w for w in poloClub.union(rugbyClub)]
lista = ['apple', 'pear', 'fig', 'orange', 'strawberry']
listb = ['pear', 'lemon', 'grapefruit', 'orange']
listc = [x for x in lista if x in listb]
listd = list(set(lista)|set(listb))

Question 1:
print listc

Question 2:
print listd
More Set Examples

```python
s = set(lista)
t = set(listb)

problem1 = (s-t) | (t-s)
print problem1

problem2 = (s|t) - (s&t)
print problem2

problem3 = (s|t|(s&t))
print problem3
```

lista = ['apple', 'pear', 'fig', 'orange', 'strawberry']
listb = ['pear', 'lemon', 'grapefruit', 'orange']
Set Operations from pictures
bit.ly/101fall14-1009-03

Question: Which picture is which operation?

A) 

B) 

C) 

D) 

E)
Problems — snarf setExample.py

• Given a list of strings that have the name of a course (one word), followed by last names of people in the course:
  – Convert list into lists of strings of names for each course
  – Find total number of people taking any course
  – Find number of people taking just one course

["compsci101 Smith Ye Li Lin Abroms Black", \
 "math101 Green Wei Lin Yavatkar Delong Noell Ye Smith", ...]
Data for example

"compsci101 Smith Ye Li Lin Abroms Black",
"math101 Green Wei Lin Yavatkar Delong Noell Ye Smith",
"econ101 Abroms Curtson Williams Smith",
"french1 Wills Wrigley Olson Lee",
"history230 Black Wrigley"]
Set Picture of Data

People in CompSci 101?
People Taking both Math101 & CompSci101?
People Taking both Math101 & CompSci101

Intersection

ECON101

COMPSCI101

MATH101

FRENCH1

HISTORY230
Part 1 — processList

bit.ly/101fall14-1009-04

• Given a list of strings that have the name of a course (one word), followed by last names of people in the course:
  – Convert list into lists of strings of names for each course

"compsci101 Smith Ye Li Lin Abroms Black", \
"math101 Green Wei Lin Yavatkar Delong Noell Ye Smith", ...

[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'],
['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong', 'Noell', 'Ye', 'Smith'], ...]
def processList(stringList):
   '''
   process the list of strings to
   return a list of lists
   and remove the course number since it is not needed
   '''
   allLists = []
   for line in stringList:
       linelist = line.split()
       # append after removing course number
       allLists.append(linelist[1:])
   return allLists
Process List: Data Visualization

["compsci101 Smith Ye Li Lin Abroms Black", \
"math101 Green Wei Lin Yavatkar Delong Noell Ye Smith", \
"econ101 Abroms Curtson Williams Smith", \
"french1 Wills Wrigley Olson Lee", \
"history230 Black Wrigley"]

[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'],
['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong', 'Noell', 'Ye', 'Smith'],
['Abroms', 'Curtson', 'Williams', 'Smith'],
['Wills', 'Wrigley', 'Olson', 'Lee'],
['Black', 'Wrigley']]
Part 2 — peopleTakingCourses
bit.ly/101fall14-1009-05

• Given a list of strings that have the name of a course (one word), followed by last names of people in the course:
  
  – Find total number of people taking any course

```javascript
["compsci101 Smith Ye Li Lin Abroms Black", 
"math101 Green Wei Lin Yavatkar Delong Noell Ye Smith", .. ] 

[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'], ['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong', 'Noell', 'Ye', 'Smith'], .. ] 
```

Answer: 11 for this data (union will remove duplicates)

Which ones did I miss?
People taking Courses - Union

- ECON101
  - Curtson
  - Williams

- HISTORY230
  - Black
  - Wrigley

- COMPSCI101
  - Smith
  - Li

- MATH101
  - Ye
  - Lin
  - Green
  - Noell
  - Wei
  - Yavatkar

- FRENCH1
  - Wills
  - Lee
  - Olson

Total Number Is 17
def peopleTakingCourses(data):
    ''' data is a list of lists of names for each course, 
    return list of all strings that appear at least once ''
    names = set([])
    for lista in data:
        seta = set(lista)
        names = names | seta
    return list(names)
    # return [w for w in names]
peopleTakingCourses: Data Visualization

["compsci101 Smith Ye Li Lin Abroms Black", \
"math101 Green Wei Lin Yavatkar Delong Noell Ye Smith", \
"econ101 Abroms Curtson Williams Smith", \
"french1 Wills Wrigley Olson Lee", \
"history230 Black Wrigley"]

['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black', 'Green', 'Wei', 'Yavatkar', 'Delong', 'Noell', 'Curtson', 'Williams', 'Wills', 'Wrigley', 'Olson', 'Lee']

17 people taking one or more courses
Part 3 – bit.ly/101fall14-1009-06

- Given a list of strings that have the name of a course (one word), followed by last names of people in the course:
  - Find number of people taking just one course
- BUT FIRST, let's write this helper method
  unionAllSetsButMe

```javascript
[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'],
 ['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong', 'Noell', 'Ye', 'Smith'],
 ['Abroms', 'Curtson', 'Williams', 'Smith'],
 ['Wills', 'Wrigley', 'Olson', 'Lee'],
 ['Black', 'Wrigley']]```
Union all sets
But French1
unionAllSetsButMe

```
[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'],
 ['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong', 'Noell', 'Ye', 'Smith'],
 ['Abroms', 'Curtson', 'Williams', 'Smith'],
 ['Wills', 'Wrigley', 'Olson', 'Lee'],
 ['Black', 'Wrigley']]
```
Part 4 — peopleTakingOnlyOneCourse

bit.ly/101fall14-1009-07

• Given a list of strings that have the name of a course (one word), followed by last names of people in the course:
  – Find number of people taking just one course

[['Smith', 'Ye', 'Li', 'Lin', 'Abroms', 'Black'],
['Green', 'Wei', 'Lin', 'Yavatkar', 'Delong',
'Noell', 'Ye', 'Smith'], ['Abroms', 'Curtson',
'Williams', 'Smith'], ['Wills', 'Wrigley',
'Olson', 'Lee'], ['Black', 'Wrigley']]

11 people taking only one course

['Curtson', 'Delong', 'Green', 'Lee', 'Li', 'Noell',
'Olson', 'Wei', 'Williams', 'Wills', 'Yavatkar']
People taking
Only one course

ECON101

COMPSCI101

MATH101

FRENCH1

HISTORY230

- Curtson
- Williams
- Abroms
- Li
- Smith
- Ye
- Lin
- Black
- Wrigley
- Green
- Noell
- Wei
- Delong
- Yavatkar
- Wills
- Lee
- Olson
APT - UniqueZoo

- How do you solve this problem?
- How is it similar to the problem we just solved?
Example Data for UniqueZoo

["zebra bear fox elephant","bear crocodile fox",
"rhino elephant crocodile kangaroo",
"elephant bear"]
UniqueZoo – two zoos have unique animals

- zebra
- elephant
- fox
- bear
- crocodile
- rhino
- kangaroo