BST281
Lab Session 3

Announcement:

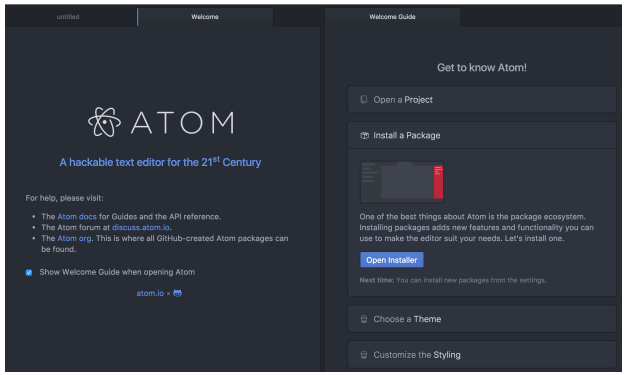
- Don't worry, Problems2 has not been posted yet. Deadline will be two weeks from the date it is posted.
- Homework solution will not be shared with students later. You're welcome to ask me in lab or email me.
- Please feel free to email me if you are confused about certain topics of python in class or lab. We welcome any feedback for the lab and we're eager to modify lab content based on students' demand.

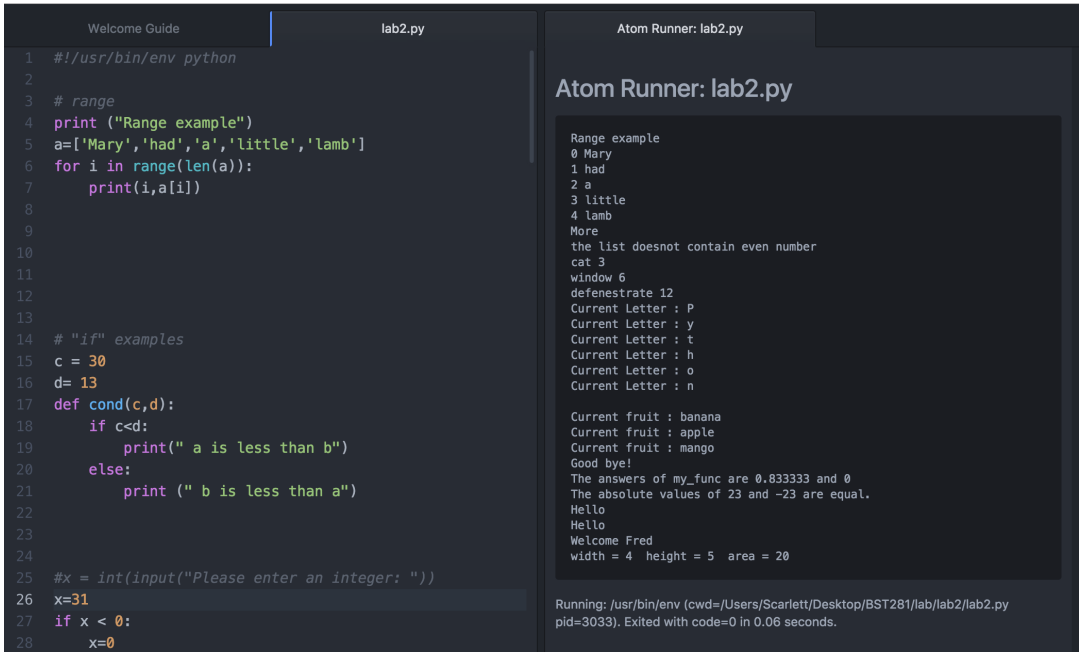
TA: Xue Zou Email: xuz943@mail.harvard.edu

Run python script in text editor

ATOM
A hackable text editor
for the 21st Century

- Install Atom (text editor)
- Install runner package for Atom
- Ctl+R (mac) or Alt+R (windows)





TA: Xue Zou xuz943@mail.harvard.edu

How to ensure your homework will run correctly

The first part of making sure your homework will run correctly is to make sure your docstring is formatted correctly. The top of your script should have the following form:

```
#!/usr/bin/env python
"""
<first name> <last name>
<script name>
<course number> <date>
<number of hours the assignment took you to complete>

<overall goals of the assignment>

<thoughts about the assignment>
"""
```

Strip Function
- Some Useful String Functions

We will learn some functions that can be used to deal with strings

- Start by opening the interpreter (open the command line and type python). Or create a py file in Atom

.strip()

Enter the following into the interpreter:

```
strString = "\tSubj\tSequence1\tSequence2\n"
print (strString)
strString.strip()
print (strString)
```

You should see:

```
	Subj	Sequence1	Sequence2

Subj	Sequence1	Sequence2
```

1. What does `.strip()` do?

2. Is `.strip()` the kind of function that modifies the value in place, or that creates a copy and returns it?

Split Function
- Some Useful String Functions

We will learn some functions that can be used to deal with strings

- Start by opening the interpreter (open the command line and type python).

.split()

Now enter the following into the interpreter:

```
string1="Hello!World"
string1.split("\t")
string1.split("!")
string1.split("e")

string2=",Subj,Sequence1,Sequnece2\n"
string2.split('\t')
string2.split('\n')
string2.split(',')
string2.split('Subj')

string3="Sequence1\tSequence2"
string3.split('\t')
```

```
>>> string1="Hello!World"
>>> string1.split("\t")
['Hello!World']
>>> string1.split("!")
['Hello', 'World']
>>> string1.split("e")
['H', 'llo!World']
>>> string2=",Subj,Sequence1,Sequnece2\n"
>>> string2.split('\t')
[',Subj,Sequence1,Sequnece2\n']
>>> string2.split('\n')
[',Subj,Sequence1,Sequnece2', '']
>>> string2.split(',')
['', 'Subj', 'Sequence1', 'Sequnece2\n']
>>> string2.split('Subj')
[',', ',Sequence1,Sequnece2\n']
>>> string3="Sequence1\tSequence2"
>>> string3.split('\t')
['Sequence1', 'Sequence2']
```

1. What does `.split(<string>)` do?

2. Is `.split()` a function that modifies the value in place, or that creates a copy and returns it?

Join Function
- Some Useful String Functions

We will learn some functions that can be used to deal with strings

- Start by opening the interpreter (open the command line and type python).

.join()

Now enter the following in the interpreter:

```
astrList = ["Subj","Sequence1","Sequence2"]
"\t".join(astrList)
" ".join(astrList)
"\t".join(["Subj"])
"\t".join("Subj")
" a string ".join(astrList)
"".join(["a",1])
```

1. What does `<string>.join(<list>)` do?

2. Is `.join()` a function that modifies the value in place, or that creates a copy and returns it?

3. What causes the error when you try `"".join(["a",1])`?

4. How could you fix the error in question 3 so that the output is `"a1"`?

You can quit the interpreter by using `quit()`

- https://docs.python.org/2/howto/regex.html

Learn about Regular Expression

Exercise: `re.search`

Syntax

Type the following into the interpreter

```
import re
re.search(r'xy')
print re.search(r'xy','xyz')
print re.search(r'a','xyz')
```

You should see the following output:

```
>>> import re
>>> re.search(r'xy')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: search() takes at least 2 arguments (1 given)
>>> print re.search(r'xy','xyz')
<_sre.SRE_Match object at 0x100439c60>
>>> print re.search(r'a','xyz')
None
>>>
```

Please try this after the Wednesday Class!
Answers to the questions are shown below:

1. How many and what kind of arguments does `re.search` take?
   It takes two arguments: the first is a pattern to find, and the second is a string in which to search for the pattern.

2. What does `re.search` return?
   It returns a Match-Object if the pattern is found in the string, and None if it is not.

Rosalind http://rosalind.info/problems/locations/ – resource of python practice

ROSALIND About Problems Statistics Glossary search Log in Register

Locations

Rosalind is a platform for learning bioinformatics and programming through problem solving. Take a tour to get the hang of how Rosalind works.

If you don't know anything about programming, you can start at the Python Village. For a collection of exercises to accompany Bioinformatics Algorithms book, go to the Textbook Track. Otherwise you can try to storm the Bioinformatics Stronghold right now.

Python Village

If you are completely new to programming, try these initial problems to learn a few basics about the Python programming language. You'll get familiar with the operations needed to start solving bioinformatics challenges in the Stronghold.

Bioinformatics Stronghold

Discover the algorithms underlying a variety of bioinformatics topics: computational mass spectrometry, alignment, dynamic programming, genome assembly, genome rearrangements, phylogeny, probability, string algorithms and others.

Bioinformatics Armory

Ready-to-use software tools abound for bioinformatics analysis. Whereas in the Stronghold you implement algorithms on your own, in the Armory you solve similar problems by using existing tools.

Problems

Bioinformatics Armory List Tree

Rosalind is a platform for learning bioinformatics and programming through problem solving. Take a tour to get the hang of how Rosalind works.

Last win: stempellek vs. "Rabbits and Recurrence Relations", 12 minutes ago

Problems: 284 (total), users: 43682, attempts: 738119, correct: 417665

ID	Title	Solved By	Correct Ratio
INI	Introduction to the Bioinformatics Armory	3204	
DBPR	Introduction to Protein Databases	1839	

Exercise: Codecademy 6-9

- If you are behind the Codeacademy exercise, feel free to continue from 6.
- If you have finished the previous exercise up to 7, please start on the Codecademy 8 "Practice Makes Perfect" section, continue through 9.