**Python\_Lesson3: Sets and Dictionaries**

Please don't forget to submit your feedback after the class. This helps a lot in increasing effectiveness of the course. Use the following link to submit your feedback:

**Lesson Overview:**

In this lesson, we will learn about sets and dictionaries in Python. We also exercise some useful application of them.

a. Sets: Usage, Features, Operations on sets like Union, Intersection and Difference

b. Dictionaries: Usage, Features, Operations on dictionaries as remove, update and insert, using dictionaries in loops

c. Some simple functions and features of extracting data from web pages like Wikipedia

**Programming elements:**

Sets, Dictionaries and Web scraping

**In class programming:**

1. Dictionaries and sets

a. Write a Python program to perform the symmetric difference between two sets.

*Input:* s1= {1,2,4,5} s2={1,2}

*Output:* {4,5}

b. Write a Python program for the following DNA application using dictionaries

* Firstly, take a codon sequence input from the user as “AAAGGGTTTTTT”
* Create a list by splitting the input sequence into 3 character long sub strings where each substring is a codon specifying an amino acid in the codon.tsv file

[‘AAA’, ‘GGG’, ‘TTT’, ‘TTT’]

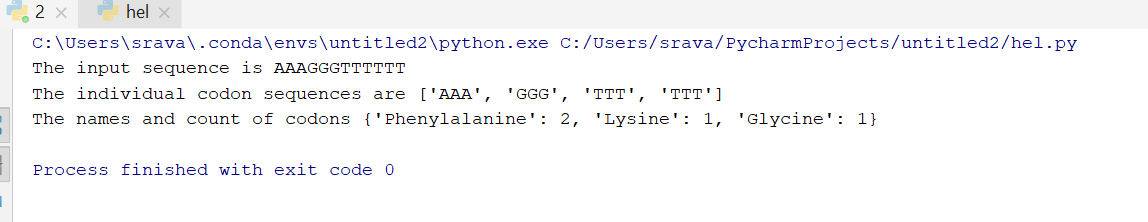
* Then read the codon.tsv file from here <https://umkc.box.com/s/azf67d5bmxfmdtmj90u0m3oqr24f9wty>
* Finally by using the codon file print the names of each codon sequence along with their count in the form of a dictionary like

{'Phenylalanine': 1, 'Lysine': 2, 'Glycine': 1} where {AAA: ‘Lysine’, TTT: ‘Phenylalanine’, ‘GGG’: ‘Glycine’}

**Reference**:

1. How to read a tsv file

<https://medium.com/@adds68/parsing-tsv-file-with-csv-in-python-662d6347b0cd>



2. Web scraping

Write a simple program that parse a Wiki page mentioned below and follow the instructions:

<https://en.wikipedia.org/wiki/Deep_learning>

1. Import these libraries

import urllib.request

from bs4 import BeautifulSoup

import os

2. Define a variable and put the link on that

3. Get the url

ex: urllib.request.urlopen(‘step 2 variable’)

4. Parse the source code using the Beautiful Soup library and save the parsed code in a variable

5. Print out the title of the page

6. Find all the links in the page (‘a’ tag)

7. Iterate over each tag(above) then return the link using attribute "href" using get

**Note:** *Cheating, plagiarism, disruptive behavior and other forms of unacceptable conduct are subject to strong sanctions in accordance with university policy. See detailed description of university policy at the following URL:* [*https://catalog.umkc.edu/special-notices/academic-honesty/*](https://catalog.umkc.edu/special-notices/academic-honesty/)