

Programming for Data Science G (11521 Online & On-campus)

Week 5 Tutorial

List, Tuple and Assignment 1

Objectives

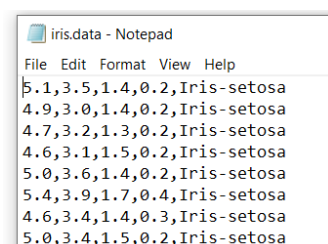
- To learn List and Tuple
- To apply List and Tuple to Assignment 1 to display data samples and cluster centres.

Create a new Python project

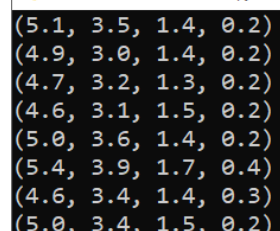
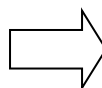
- Create **Week5Tutorial** project (Python)
- Add a module file and name it **io_data_module.py**
- There are 13 examples for List and 9 examples for Tuple in Week 4 Lecture
- Review those examples before you answer the questions below.

Questions: Write Python code in **Week5Tutorial.py** for each question below

- **Question 1:** Create a list of 100 elements like this `[0, 1, 2, 3, 4, ..., 99]`
- **Question 2:** Create a tuple of 100 elements like this `(0, 1, 2, 3, 4, ..., 99)`
- **Question 3:** Change values of `input_list` from string to number and output as `output_list`
 - `input_list = ['2.1', '3.5', '4.8', '1.1', '2.0']`
 - `output_list = [2.1, 3.5, 4.8, 1.1, 2.0]`
- **Question 4:** Change each element `x` in a list to `x / sum` where `sum` is total of all elements in that list. For example, `mylist = [0, 2, 1, 3, 1, 2, 0, 1]` and `sum = 0+2+1+3+1+2+0+1 = 10` and `mylist` becomes `[0.0, 0.2, 0.1, 0.2, 0.3, 0.2, 0.0, 0.1]`
- **Question 5:** Remove the first and last elements from a list. For example, `my_list = ['red', 0, 2, 1, 1, 2, 0, 1, 'blue']` becomes `my_list = [0, 2, 1, 1, 2, 0, 1]`
- **Question 6:** Change `0` to `10` in `[0, 1, 0, 2, 0, 1]` to have `[10, 1, 10, 2, 10, 1]`
- **Question 7:** Combine `list1` and `list2` to have `list3`, `list4` and `list5` as follows
 - `list1 = [2, 3, 1]`
 - `list2 = [4, 5, 2]`
 - `list3 = [2, 3, 1, 4, 5, 2]`
 - `list4 = [[2, 3, 1], [4, 5, 2]]`
 - `list5 = [(2, 3, 1), (4, 5, 2)]`
- **Question 8:**
 - Write a function (**`read_multi_dim_data`**) in **`io_data_module.py`** that reads data from **`iris.data`** file and outputs a list of tuples where each tuple is a data sample.
 - Write a program in **`Week5Tutorial.py`** that calls the **`read_multi_dim_data`** function to print out all 150 data samples. Below is example for the first 8 lines



```
iris.data - Notepad
File Edit Format View Help
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
```



```
(5.1, 3.5, 1.4, 0.2)
(4.9, 3.0, 1.4, 0.2)
(4.7, 3.2, 1.3, 0.2)
(4.6, 3.1, 1.5, 0.2)
(5.0, 3.6, 1.4, 0.2)
(5.4, 3.9, 1.7, 0.4)
(4.6, 3.4, 1.4, 0.3)
(5.0, 3.4, 1.5, 0.2)
```

For more information on the **iris** data set

- Each line contains 4 real numbers and a string. The 4 numbers form a 4-dimensional data sample and the string is class label (the term 'class' here is for data classification, it's not class in Python object-oriented programming).
 - There are 3 classes (Iris-setosa, Iris-versicolor, and Iris-virginica) in the Iris dataset.
 - There are 50 data samples in each class.
 - All 150 data samples are in 4-dimensional data space.
- **Question 9:** Write a Python program in **Week5Tutorial.py** that
 - reads all 150 data samples in **iris.data** and displays them on canvas using the function developed in Question 8 above, where **x** and **y** are the **first** and **second** values in each data sample, respectively, and
 - displays 3 centre samples **centre_1** = (5.1, 3.0, 1.1, 0.5), **centre_2** = (4.4, 3.2, 2.8, 0.2), and **centre_3** = (5.7, 3.9, 3.9, 0.8) on the same canvas with the **iris.data** data samples as seen below where red dots are iris data samples and black dots are centres.

