

Artificial Intelligence Lab

CSL 411

Lab Journal 2



Name:
Enrollment:
BSCS 6A

Department of Computer Science
BAHRIA UNIVERSITY, ISLAMABAD

Lab # 2: Lists, Tuples, Set, Dictionary, Numpy & Pandas

Objectives:

To learn about different data structures in python and how to use them.

Tools Used:

IDLE (Python 3.4 GUI Python)

Submission Date:

Evaluation:

Signatures of Lab Engineer:

Task # 1:

Create list of Fibonacci numbers after calculating Fibonacci series up to the number n which you will pass to a function as an argument. The number n must be input by the user.

Fibonacci numbers are calculated using the following formula: The first two numbers of the series are always equal to 1, and each consecutive number returned is the sum of the last two numbers.

Hint: Can you use only two variables in the generator function?

The code below will simultaneously switch the values of a and b.

```
a = 1
b = 2
a, b = b, a
```

The first number in the series should be 1. (The output will start like 1,1,2,3,5,8,...)

Procedure/Program:

```
def FibonacciSeries(n):
    num1,num2=0,1
    count =0
    while count<n:
        print(num1)
        term=num1+num2
        num1=num2
        num2=term
        count+=1
n=int(input("Enter number of terms: "))
print("Fibonacci Series: ")
FibonacciSeries(n)
```

Result/Output:

```
Enter number of terms: 10
Fibonacci Series:
0
1
1
2
3
5
8
13
21
34
```

Analysis/Conclusion:

In above task we learn how to make fibseries in python

Task # 2:

Write a program that lets the user enter in some English text, then converts the text to Pig-Latin. To review, Pig-Latin takes the first letter of a word, puts it at the end, and appends “ay”. The only exception is if the first letter is a vowel, in which case we keep it as it is and append “hay” to the end. For example: “hello” -> “ellohay”, and “image” -> “imagehay”

It will be useful to define a list or tuple at the top called VOWELS. This way, you can check if a letter *x* is a vowel with the expression *x* in VOWELS.

It’s tricky for us to deal with punctuation and numbers with what we know so far, so instead, ask the user to enter only words and spaces. You can convert their input from a string to a list of strings by calling split on the string:

“My name is John Smith”.split(“ ”) -> [“My”, “name”, “is”, “John”, “Smith”]

Procedure/Program:

```
vowels=('A','a','E','e','I','i','O','o','U','u')
sentence=input("Enter a sentence: ")
sentence=sentence.split()
for x in range(len(sentence)):
    if sentence[x][0] in vowels:
        sentence[x]+='hay'
    else:
        sentence[x]=sentence[x][1:]+sentence[x][0]
        sentence[x]+='ay'
sentence=' '.join(sentence)
print(sentence)
```

Result/Output:

```
In [22]: runfile('C:/Users/Student/Desktop/untitled4.py', wdir='C:/Users/Student/Desktop')

Enter a sentence: hello
ellohay

In [23]: runfile('C:/Users/Student/Desktop/untitled4.py', wdir='C:/Users/Student/Desktop')

Enter a sentence: image
imagehay

In [24]: runfile('C:/Users/Student/Desktop/untitled4.py', wdir='C:/Users/Student/Desktop')

Enter a sentence: My name is John Smith
yMay amenay ishay ohnJay mithSay
```

Analysis/Conclusion:

In above task we learn how to use built in function of split and join in python

Task # 3:

Write a Pandas/Numpy program to find the index of the first occurrence of the smallest and largest value of a given series

Procedure/Program:

```
import pandas as pd
numbers=pd.Series([11, 8, 7, 5, 6, 5, 3, 4, 7, 1])
print("Original Series: ")
print(numbers)
print("Index of the smallest and largest value of the above series is: ")
print(numbers.idxmin())
print(numbers.idxmax())
```

Result/Output:

```
In [32]: runfile('C:/Users/Student/Desktop/untitled4.py', wdir='C:/Users/
Student/Desktop')
Original Series:
0    11
1     8
2     7
3     5
4     6
5     5
6     3
7     4
8     7
9     1
dtype: int64
Index of the smallest and largest value of the above series is:
9
0
```

Analysis/Conclusion:

In above task we learn how to use panda module and its built in functions in python

Task # 4:

Write a Pandas program to compute the Euclidean distance between two given series.

Euclidean distance

From Wikipedia, In mathematics, the Euclidean distance or Euclidean metric is the "ordinary" straight-line distance between two points in Euclidean space. With this distance, Euclidean space becomes a metric space.

Series-1: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Series-2: [11, 8, 7, 5, 6, 5, 3, 4, 7, 1]

Procedure/Program:

```
import pandas as pd
import numpy as np
```

```

x=pd.Series([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
y=pd.Series([11, 8, 7, 5, 6, 5, 3, 4, 7, 1])
print("Original Series: ")
print(x)
print(y)
print("Euclidean distance between two series: ")
print(np.linalg.norm(x-y))

```

Result/Output:

```

In [34]: runfile('C:/Users/Student/Desktop/untitled4.py', wdir='C:/Users/
Student/Desktop')
Original Series:
0      1
1      2
2      3
3      4
4      5
5      6
6      7
7      8
8      9
9     10
dtype: int64
0     11
1      8
2      7
3      5
4      6
5      5
6      3
7      4
8      7
9      1
dtype: int64
Euclidean distance between two series:
16.4924225025

```

Analysis/Conclusion:

In above task we learn how to use pandas and numpy module and its built-in functions in python

Task # 5:

Visualize the following data in python. Please provide the reason for the choice of graph.

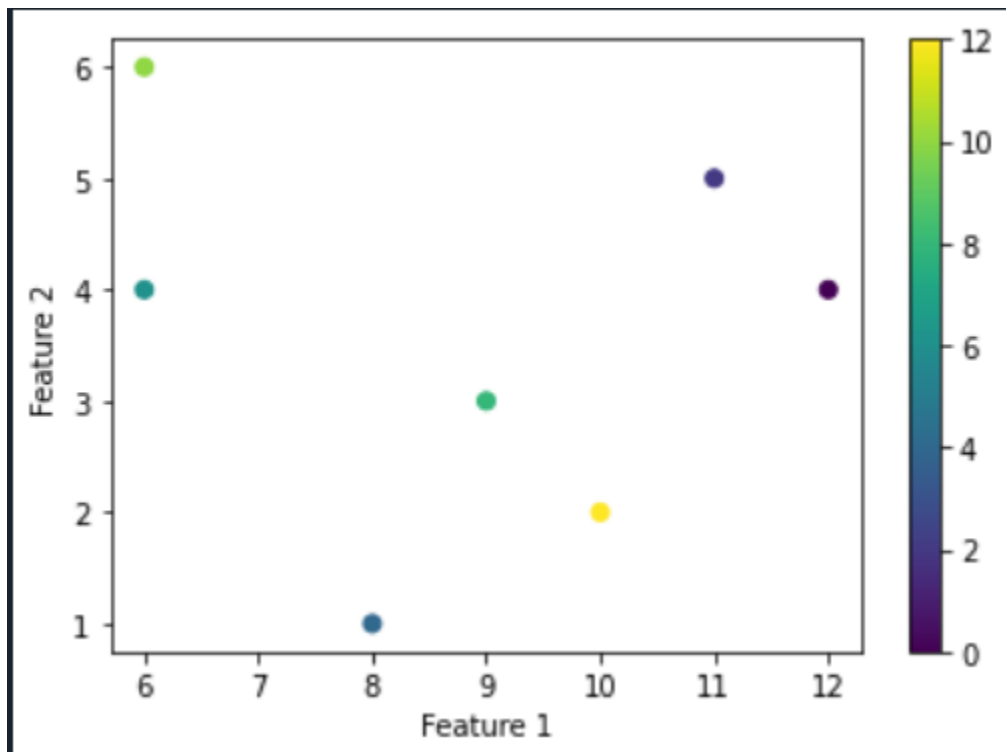
Feature 1	Feature 2	Class
12	4	a
11	5	a
8	1	a

6	4	b
9	3	b
6	6	a
10	2	b

Procedure/Program:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.array([12,11,8,6,9,6,10])
y = np.array([4,5,1,4,3,6,2])
colors = np.array([0, 2, 4, 6, 8, 10, 12])
plt.scatter(x, y, c=colors, cmap='viridis')
plt.xlabel("Feature 1")
plt.ylabel("Feature 2")
plt.colorbar()
plt.show()
```

Result/Output:



Analysis/Conclusion

In above task we learn how to plot graphs in python