Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Computer Science and Engineering (Data Science)

Course: Minors in Data Science

Subject: Machine Learning (DJ19MN4C2)

AY: 2022-23

Experiment 1

Name: Ayush Jain
SAP ID: 60004200132
Branch: Computer Engineering

Aim: Fundamentals of python programming.

Theory:

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.

- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-oriented way or a functional way.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Lab Assignments to complete in this session

Use google colab and complete code for following questions.

Question: Write a comment in Python.

Code: `# Ayush Jain`

Question: Write a multiline comment/paragraph in Python.

```
'''60004200132
   B
  Comps'''
```

Primitive Data Types

Question: Write a program to print an integer, float, string, complex number, Boolean, and bytes in Python and display their data type.

Code:

```
a=1
b=1.0
c=1+2j
d=True
e=bytes(4)
print(type(a))
print(type(b))
print(type(c))
print(type(d))
print(type(e))
```

Data Structures

Lists

Question: Write a program to create a list. Collect heterogenous data in it.

Code:

```
list1 = ["Apple",10,5.5]
```

Question: Write a program to print a list.

Code:

```
list1 = ["Apple",10,5.5]
print(list1)
```

```
Output: ['Apple', 10, 5.5]
```

Question: Write a program to print a new list. Append an item in this list.

Code:

```
list2 = ["Banana","mango"]
list2.append("Cherry")
print(list2)
```

```
Output: ['Banana', 'mango', 'Cherry']
```

Question: Write a program to make a copy of the previous list.

Code:

```
list3 = list2.copy()
print(list3)
```

```
Output: ['Banana', 'mango', 'Cherry']
```

Question: Write a program to concatenate 2 lists and print the output.

Code:

```
list4 = list1 + list2
print(list4)
```

```
Output: ['Apple', 10, 5.5, 'Banana', 'mango', 'Cherry'
```

Question: Write a program to count the number of elements present in a list.

Code:

```
count = 0
for i in range(len(a_list)) :
    count+=i
print(count)
```

Question: Write a program to print the length of a list.

Code:

```
print(len(list4))
```

```
Output: 6
```

Question: Write a program to append more than 1 item in a list.

Code:

```
lista = [1,2,3]
listb = ["a","b","c"]
for x in listb:
  lista.append(x)
print(lista)
```

Question: Write a program to extend a list.

Code:

```
list5 = [1,2,3,4,5]
list5.extend([6,7])
print(list5)
```

```
Output: [1, 2, 3, 4, 5, 6, 7]
```

Question: Write a program to insert a value at a position in a list.

Code:

```
fruits = ['apple', 'banana', 'cherry']
fruits.insert(1, "orange")
print(fruits)
```

```
Output: ['apple', 'orange', 'banana', 'cherry']
```

Question: Write a program to delete a value at a given position in a list.

Code:

```
fruits.pop(1)
print(fruits)
```

```
Output: ['apple', 'banana', 'cherry']
```

Question: Write a program to remove a value from the list.

Code:

```
fruits.pop(1)
print(fruits)
```

```
Output: ['apple', 'banana', 'cherry']
```

Question: Write a program to slice the data in a list.

Code:

```
slice = [1,2,3,4,5]
print(slice[1:3])
```

```
Output: [2, 3]
```

Question: Write a program to slice data in a list using positions.

Code:

```
x=slice(2)
e_list=b_list.copy()
print(e_list[x])
```

Question: Write a program to print the last 8 elements.

Code:

```
n=8
b_list.extend([8,9,10,11,12,13,14,15])
print(b_list)
result=b_list[-n:]
print(result)
```

Question: Write a program to print the last value of a list.

Code:

```
l = [0,1,2,3,4,5,6,7,8,9]
print(l[-1])
```

```
Output: 9
```

Question: Write a program to print the central value of a list.

Code:

```
l2 = [1,2,3,4,5]
middle = int((len(l2) - 1)/2)
print(middle)
print(l2[middle])
```

Tuples

Question: Write a program to create a tuple. Collect heterogenous data in it.

Code:

```
tup = (7,"apple",1,2.5,True)
print(tup)
```

```
Output: (7, 'apple', 1, 2.5, True)
```

Question: Write a program to print the position of an item in the tuple.

Code:

```
print(tup.index("apple"))
```

```
Output: 1
```

Question: Print a new tuple. Write a program to concatenate two tuples.

Code:

```
tup2 = ("Banana","Cherry")
print(tup + tup2)
```

```
Output: (7, 'apple', 1, 2.5, True, 'Banana', 'Cherry')
```

Question: Write a program to print the value at position 2 in the concatenated tuple.

Code:

```
print(tup2[3])
```

Question: Write a program to change the element of a tuple.

Code:

```
tup4 = ("apple", "banana", "cherry")
print(tup4)
m = list(tup4)
m[1] = "kiwi"
tup4 = tuple(m)
print(tup4)
```

```
Output: ('apple', 'kiwi', 'cherry')
```

Dictionary

Question: Write a program to create and print a dictionary.

Code:

```
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
print(thisdict)
```

```
Output: {'brand': 'Ford', 'model': 'Mustang', 'year': 1964}
```

Question: Write a program to print values of a dictionary using keys.

Code:

```
print(thisdict["year"])
```

```
Output: 1964
```

Question: Write a program to create a multidimensional dictionary.

Code:

```
myfamily = {

  "child1" : {
    "name" : "Emil",
    "year" : 2004
  },
  "child2" : {
    "name" : "Tobias",
    "year" : 2007
  },
  "child3" : {
    "name" : "Linus",
    "year" : 2011
```

```
  }
}
```

Question: Write a program to print values from the multidimensional dictionary using keys.

Code:

```
print(myfamily["child2"]["name"])
```

```
Output: Tobias
```