LAB MANUAL

CS-340-L
Artificial Intelligence

LAB 02

Introduction to Python

Python is an object-oriented programming language created by Guido Rossum in 1989. It is ideally designed for rapid prototyping of complex applications. It has interfaces to many OS system calls and libraries and is extensible to C or C++. Many large companies use the Python programming language, including NASA, Google, YouTube, BitTorrent, etc.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

```
# This is my first Python program.
# This will print 'Hello, World!' as the output

print('hello world')

print("hello world")

print("hello world");
```

Indentation

```
if 5 > 2:

  print("Five is greater than two!")

  print("Five is greater than two!")
```

Variables

Varaibles

```
In [4]: t=88
        t="hello" #you can write 'hello' as well
        t
Out[4]: 'hello'
```

x=9

t=int(5)

print(t)

```
print(type(t))
```

```
<class 'str'>
```

This is a comment

print ("Hello, World!") #This is a comment

""" You can also multicomment in between these
"""

Variable Names

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume). Rules for Python variables: A variable name must start with a letter or the underscore character A variable name cannot start with a number A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _) Variable names are case-sensitive (age, Age and AGE are three different variables)

```
In [7]: var='Ali'
        var1='Ali'
        _var='Ali'
        var_1='Ali'
```

In [8]:
```
x, y, z = "Orange", "Banana", "Cherry"
print(x)
print(y)
print(z)
```

```
Orange
Banana
Cherry
```

In [9]:
```
x = "Python is "
y = "awesome"
z =  x + y
print(z)
```

```
Python is awesome
```

Python Datatype

Example Data Type

x = "Hello World" str
x = 20 int
x = 20.5 float
x = 1j complex
x = ["apple", "banana", "cherry"] list
x = ("apple", "banana", "cherry") tuple
x = {"name" : "John", "age" : 36} dict
x = {"apple", "banana", "cherry"} set
x = frozenset({"apple", "banana", "cherry"}) frozenset
x = True bool

List vs Tuple

```
In [2]: m=['apple',1,'capsicum']

In [3]: m[1]= 'Orange'

In [4]: m
Out[4]: ['apple', 'Orange', 'capsicum']

In [7]: x=('apple','orange','mangoes')

In [8]: x[2]='berry'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_11560/618453853.py in <module>
----> 1 x[2]='berry'

TypeError: 'tuple' object does not support item assignment
```

```
In [9]: import sys
        print(sys.getsizeof(m))
        print(sys.getsizeof(x))
```

```
120
64
```

To append and extend the list

```
In [26]: list.append("orange")
         print(list)
```

```
['apple', 'blackcurrant', 'cherry', 'orange', 'orange', 'orange']
```

```
In [27]: list.insert(1, "appricot")

In [29]: print(list)
```

```
['apple', 'appricot', 'blackcurrant', 'cherry', 'orange', 'orange', 'orange']
```

```
In [30]: list2 = ["tea", "mangoes", "halwa"]

In [31]: list.extend(list2)

In [32]: print(list)
```

```
['apple', 'appricot', 'blackcurrant', 'cherry', 'orange', 'orange', 'orange', 'tea', 'mangoes', 'halwa']
```

Dictionary

In [10]:
```
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
print(thisdict)
```

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

In [11]:
```
print(thisdict['brand'])
```

```
Ford
```

In [12]:
```
thisdict.update({'color':'Red'})
```

In [13]:
```
print(thisdict)
```

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

In [14]:
```
thisdict.pop("model")
```

Out[14]:
```
'Mustang'
```

In [15]:
```
thisdict
```

Out[15]:
```
{'brand': 'Ford', 'year': 1964, 'color': 'Red'}
```

Loops

In [17]:
```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

```
apple
banana
cherry
```

In [48]:
```
for x in fruits:
  print(x)
  if x == 'banana':
    break
```

```
apple
banana
```

In [19]:
```
numbers = [6, 5, 3, 8, 4, 2, 5, 4, 11]
sum = 2
for val in numbers:
    sum = sum+val

print("The sum is", sum)
```

```
The sum is 50
```

IF Else

In [38]:
```
a = 200
b = 33
if b > a:
  print("b is greater than a")
elif a == b:
  print("a and b are equal")
else:
  print("a is greater than b")
```

```
a is greater than b
```

In [39]:
```
x = 41

if x > 10:
  print("Above ten,")
  if x > 20:
    print("and also above 20!")
  else:
    print("but not above 20.")
```

```
Above ten,
and also above 20!
```

In [40]:
```
i = 1
while i < 6:
  print(i)
  if i == 3:
    break
  i += 1
```

```
1
2
3
```

In [42]:
```
i = 0
while i < 6:
  i += 1
  if i == 3:
    continue
  print(i)
```

```
1
2
4
5
6
```