Name: Afrida Islam

Student ID: 21221030

EEE385IL (MACHINE LEARNING LABORATORY)

LAB 1: Introduction to Python

Different Types of Data in Python

Variables can store data of different types, and different types can do different things. Python has the following data types built-in by default:

- Text Type: str
- Numeric Types: int, float, complex
- Sequence Types: list, tuple, range
- Mapping Type: dict
- Boolean Type: bool
- Binary Types: bytes, bytearray, memoryview
- None Type: NoneType

Strings

A string is a series of characters, surrounded by single or double quotes

```
print("Hello world!")
```

```
Hello world!
```

```
msg = "Hello world!"
print(msg)
```

```
Hello world!
```

Concatenation (combining strings)

```
first_name = 'albert'
last_name = 'einstein'
full_name = first_name + ' ' + last_name
print(full_name)
```

```
albert einstein
```

Numeric Data

```
x = 20    #int
print(x)
y = 20.5    #float
print(y)
z = 1j    #complex
print(z)
```

```
20
20.5
1j
```

Getting the Data Type

```
x = 2 + 3j
print(type(x))
a = 5.0
```

```
<class 'complex'>
```

```
print(type(a))
```

```
<class 'float'>
```

```
a = 20
b = 30
print(a+b)
```

```
50
```

```
a = a + a
print(a)
```

```
200
```

```
a = 50
print(a+b)
```

```
80
```

Taking User Input

```
name = input("Please input your name: ")
print("Hello " + name + " !")
```

```
Please input your name: Afriiii
Hello Afriiii !
```

```
f_name = input("Please input your first name: ")
l_name = input("Please input your last name: ")
print("Hello " + f_name +" " +l_name + " !")
```

```
Please input your first name: Afrida
Please input your last name: Islam
Hello Afrida Islam !
```

```
text = "The value of the variable a is {} and x is {}".format(a, x)
print(text)
```

```
The value of the variable a is 200 and x is (2+3j)
```

```
print(f"The values are: {a} and {x}")
```

```
The values are: 200 and (2+3j)
```

```
a = int(input("First no: "))
b = int(input("Second no: "))
print (f"The summation is {a+b}")
```

```
First no: 10
Second no: 21
The summation is 31
```

TASK 1

Take input your ***Name*** and ***ID*** and then store them in two separate variables. Print your ***Name*** and ***ID*** after concatenation.

```
####################################
#######    Code starts      #########

Name = input("Enter Name:")
ID = input("Enter ID:")
print(Name+ " "+ ID)
#######    Code ends        #########
####################################
```

```
Enter Name:Afrida
Enter ID:21221030
Afrida 21221030
```

```
n1 = input("Enter the first number: ")
n2 = input("Enter the second number: ")
```

```
add = int(n1) + int(n2) # type casting
print(add)
```

```
Enter the first number: 10
Enter the second number: 20
30
```

```
print((type(n1)))
print((type(add)))
```

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

Conditional Statements

Conditional operators:

- equal --> ==
- NOT equal --> !=
- greater than or equal --> >=
- less than or equal --> <=

if-else statements

```
if condition:
    statements if condition is TRUE
elif condition:
    statements
else:
    statements
```

```
marks = int(input("Enter: "))
if marks > 90:
  print('Congrats! A!')
elif marks > 80:
  print("Alright!")
else:
  print('Need to do better!')
```

```
Enter: 95
Congrats! A!
```

Multiple Condition checking

`and` --> TRUE if both conditions are met

`or` --> TRUE if any one condition is met

TASK 2

if cg > 3.8 && credits > 30
scholarship 10%
if cg > 3.5 && credits > 30
scholarship 5%
if cg < 3.5 || credits < 30
no scholarship

```
####################################
#######    Code starts      #########
cgpa =float(input("Enter Cgpa:"))
cred = int(input("Enter Cred:"))
if cgpa > 3.8 and cred > 30:
  print("Scholarship 10%")
elif cgpa > 3.5 and cred > 30:
  print("Scholarship 5%")
```

```
else:
  print("No Scholarship")


#######    Code ends        #########
#####################################
```

```
Enter Cgpa:3.7
Enter Cred:111
Scholarship 5%
```

Loops

for loop

```
for i in range(N):
    statements
```

```
for a in range(0,10,2):
  print(a)  # indexing starts from 0 in python
```

```
0
2
4
6
8
```

while loop

```
while condition:
    statements
```

```
a = 5

while a <= 10:
  print(a)
  a = a + 1
```

```
5
6
7
8
9
10
```

TASK 3

Take input of the cgpa & credits of 3 students using loop and do the same thing as TASK 2 for each of the students

```
#####################################
#######    Code starts      #########
for a in range(1,4):
 cgpa =float(input(f"Enter Cgpa of student {a}:"))
 cred = int(input(f"Enter Cred of student {a}:"))

 if cgpa > 3.8 and cred > 30:
  print("Scholarship 10%")
 elif cgpa > 3.5 and cred > 30:
  print("Scholarship 5%")
 else:
  print("No Scholarship")

#######    Code ends        #########
#####################################
```

```
Enter Cgpa of student 1:3.8
Enter Cred of student 1:120
Scholarship 5%
Enter Cgpa of student 2:3.2
Enter Cred of student 2:66
No Scholarship
Enter Cgpa of student 3:3.0
Enter Cred of student 3:30
```

```
No Scholarship
```

List (like Array in C programming)

```
list_name = [val_1, val_2, val_3, .......]
```

```
marks = [88, 56, 94, 92, 76]

print(len(marks))
```

```
for mark in marks:
  print("The marks of Student is {}".format(mark)) #{} placeholder in python
```

```
5
The marks of Student is 88
The marks of Student is 56
The marks of Student is 94
The marks of Student is 92
The marks of Student is 76
```

TASK 4

Store the cgpa and completed credits of 5 students in two separate lists and then print their scholarship status.

```
####################################
#######    Code starts      #########
cgpa = [3.6, 2.3, 2.8, 3.9, 4.0]
credits = [120, 65, 130, 25, 100]

min_cgpa = 3.5
min_credits = 30

for i in range(len(cgpa)):
    if cgpa[i] >= min_cgpa and credits[i] >= min_credits:
        print(f"Student {i+1} is eligible for scholarship!")
    else:
        print(f"Student {i+1} is not eligible for scholarship")

#######    Code ends        #########
####################################
```

```
Student 1 is eligible for scholarship!
Student 2 is not eligible for scholarship
Student 3 is not eligible for scholarship
Student 4 is not eligible for scholarship
Student 5 is eligible for scholarship!
```

Common List operations

```
marks = [88, 56, 94, 82, 94]
print("Before changing any element                --> ", marks)

marks.append(99)
print("After appending one element                --> ", marks)

marks.remove(94)
print("After removing one element                 --> ", marks)

marks.insert(1, 71)
print("After inserting '71' at index '1'          --> ", marks)

marks.pop(2)
print("After popping out the element at index '2' --> ", marks)
```

```
Before changing any element               -->  [88, 56, 94, 82, 94]
After appending one element               -->  [88, 56, 94, 82, 94, 99]
After removing one element                -->  [88, 56, 82, 94, 99]
After inserting '71' at index '1'         -->  [88, 71, 56, 82, 94, 99]
After popping out the element at index '2' -->  [88, 71, 82, 94, 99]
```

Downloading file from Colab as PDF

```
# Install the package for Tex and then convert to PDF directly as LaTex
!sudo apt-get install texlive-xetex texlive-fonts-recommended texlive-plain-generic

# Provide the file path of the notebook file
!jupyter nbconvert --to pdf "/content/drive/MyDrive/Colab Notebooks/Afrida_Task01.ipynb"
```

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 libapache-pom-java libcommons-logging-java
  libcommons-parent-java libfontbox-java libfontenc1 libgs9 libgs9-common
  libidn12 libijs-0.35 libjbig2dec0 libkpathsea6 libpdfbox-java libptexenc1
  libruby3.0 libsynctex2 libteckit0 libtexlua53 libtexluajit2 libwoff1
  libzzip-0-13 lmodern poppler-data preview-latex-style rake ruby
  ruby-net-telnet ruby-rubygems ruby-webrick ruby-xmlrpc ruby3.0
  rubygems-integration t1utils teckit tex-common tex-gyre texlive-base
  texlive-binaries texlive-latex-base texlive-latex-extra
  texlive-latex-recommended texlive-pictures tipa xfonts-encodings
  xfonts-utils
Suggested packages:
  fonts-noto fonts-freefont-otf | fonts-freefont-ttf libavalon-framework-java
  libcommons-logging-java-doc libexcalibur-logkit-java liblog4j1.2-java
  poppler-utils ghostscript fonts-japanese-mincho | fonts-ipafont-mincho
  fonts-japanese-gothic | fonts-ipafont-gothic fonts-arphic-ukai
  fonts-arphic-uming fonts-nanum ri ruby-dev bundler debhelper gv
  | postscript-viewer perl-tk xpdf | pdf-viewer xzdec
  texlive-fonts-recommended-doc texlive-latex-base-doc python3-pygments
  icc-profiles libfile-which-perl libspreadsheet-parseexcel-perl
  texlive-latex-extra-doc texlive-latex-recommended-doc texlive-luatex
  texlive-pstricks dot2tex prerex texlive-pictures-doc vprerex
  default-jre-headless tipa-doc
The following NEW packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 libapache-pom-java libcommons-logging-java
  libcommons-parent-java libfontbox-java libfontenc1 libgs9 libgs9-common
  libidn12 libijs-0.35 libjbig2dec0 libkpathsea6 libpdfbox-java libptexenc1
  libruby3.0 libsynctex2 libteckit0 libtexlua53 libtexluajit2 libwoff1
  libzzip-0-13 lmodern poppler-data preview-latex-style rake ruby
  ruby-net-telnet ruby-rubygems ruby-webrick ruby-xmlrpc ruby3.0
  rubygems-integration t1utils teckit tex-common tex-gyre texlive-base
  texlive-binaries texlive-fonts-recommended texlive-latex-base
  texlive-latex-extra texlive-latex-recommended texlive-pictures
  texlive-plain-generic texlive-xetex tipa xfonts-encodings xfonts-utils
0 upgraded, 54 newly installed, 0 to remove and 49 not upgraded.
Need to get 182 MB of archives.
After this operation, 571 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-droid-fallback all 1:6.0.1r16-1.1build1 [1,805 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-lato all 2.0-2.1 [2,696 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/main amd64 poppler-data all 0.4.11-1 [2,171 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 tex-common all 6.17 [33.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-urw-base35 all 20200910-1 [6,367 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgs9-common all 9.55.0~dfsg1-0ubuntu5.9 [752 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libidn12 amd64 1.38-4ubuntu1 [60.0 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy/main amd64 libijs-0.35 amd64 0.35-15build2 [16.5 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/main amd64 libjbig2dec0 amd64 0.19-3build2 [64.7 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgs9 amd64 9.55.0~dfsg1-0ubuntu5.9 [5,033 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libkpathsea6 amd64 2021.20210626.59705-1ubuntu0.2 [60.4 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/main amd64 libwoff1 amd64 1.0.2-1build4 [45.2 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy/universe amd64 dvisvgm amd64 2.13.1-1 [1,221 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy/universe amd64 fonts-lmodern all 2.004.5-6.1 [4,532 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-noto-mono all 20201225-1build1 [397 kB]
```