



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No.1
To implement the basic data types and control structures in python.
Date of Performance:
Date of Submission:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No 1:

Aim: To implement the basic data types and control structures in python.

Theory:

Python has the following data types built-in by default, in these categories

Text Type: Str

Numeric Types: int, float, complex

Sequence Types: list, tuple, range

Mapping Type: Dict

Set Types: set, frozenset

Boolean Type: Bool

Binary Types: bytes, bytearray, memoryview

Getting the Data Type

You can get the data type of any object by using the type() function:

Print the data type of the variable x:

```
x = 5
```

```
print(type(x))
```

Casting

There can be two types of Type Casting in Python –

- Implicit Type Casting
- Explicit Type Casting

Implicit Type Conversion



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

In this, methods, Python converts data type into another data type automatically. In this process, users don't have to involve in this process.

```
# Python program to demonstrate
```

```
# implicit type Casting
```

```
# Python automatically converts
```

```
# a to int
```

```
a = 7
```

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

```
# Python automatically converts
```

```
# b to float
```

```
b = 3.0
```

```
print(type(b))
```

```
# Python automatically converts
```

```
# c to float as it is a float addition
```

```
c = a + b
```

```
print(c)
```

```
print(type(c))
```

```
# Python automatically converts
```

```
# d to float as it is a float multiplication
```

```
d = a * b
```

```
print(d)
```

```
print(type(d))
```

Output:

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



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

```
10.0
```

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

```
21.0
```

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

Explicit Type Casting

In this method, Python need user involvement to convert the variable data type into certain data type in order to the operation required.

Mainly in type casting can be done with these data type function:

- **Int()** : Int() function take float or string as an argument and return int type object.
- **float()** : float() function take int or string as an argument and return float type object.
- **str()** : str() function take float or int as an argument and return string type

object. **Let's see some example of type casting:**

Type Casting int to float:

Here, we are casting integer object to float object with **float()** function.

```
# Python program to demonstrate
```

```
# type Casting
```

```
# int variable
```

```
a = 5
```

```
# typecast to float
```

```
n = float(a)
```



```
print(n)
```

```
print(type(n))
```

Output:

```
5.0
```

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

Sequence data types

Python has 4 built in data types used to store collections of data, the List, Tuple, Set, and Dictionary, all with different qualities and usage.

1.List: Lists are used to store multiple items in a single variable.

```
thislist = ["apple", "banana", "cherry"]
```

```
print(thislist)
```

2.Tuple: A tuple is a collection which is ordered and **unchangeable**.

Tuples are written with round brackets.

```
thistuple = ("apple", "banana", "cherry")
```

```
print(thistuple)
```

3.Set: A set is a collection which is *unordered*, *unchangeable**, and *unindexed*. * **Note:** Set *items* are unchangeable, but you can remove items and add new items. Sets are written with curly brackets.

```
thisset = {"apple", "banana", "cherry"}
```

```
print(thisset)
```

4.Dictionary: A dictionary is a collection which is ordered*, changeable and do not allow duplicates. Dictionaries are written with curly brackets, and have keys and values:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
thisdict = {  
  
    "brand": "Ford",  
    "model": "Mustang",  
    "year": 1964  
}  
print(thisdict)  
  
# Python3 program for  
explaining # use of list,  
tuple, set and  
  
# dictionary  
  
# Lists  
  
l = []  
  
# Adding Element into list  
  
l.append(5)  
  
l.append(10)  
  
print("Adding 5 and 10 in  
list", l) # Popping  
Elements from list  
  
l.pop()  
  
print("Popped one element from  
list", l) print()  
  
# Set  
  
s = set()  
  
# Adding element into set  
  
s.add(5)  
  
s.add(10)
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
print("Adding 5 and 10 in  
set", s) # Removing  
element from set
```

```
s.remove(5)
```

```
print("Removing 5 from  
set", s) print()
```

```
# Tuple
```

```
t = tuple(1)
```

```
# Tuples are immutable
```

```
print("Tuple", t)
```

```
print()
```

```
# Dictionary
```

```
d = {}
```

```
# Adding the key value pair
```

```
d[5] = "Five"
```

```
d[10] = "Ten"
```

```
print("Dictionary", d)
```

```
# Removing key-value pair
```

```
del d[10]
```

```
print("Dictionary", d)
```

Control Structures in Python

Python programming language provides following types of loops to handle looping requirements.

1.While Loop

Syntax :



while expression:

statement(s)

2.For in Loop

Syntax:

for iterator_var in sequence:

statements(s)

3.Nested Loops

Syntax:

for iterator_var in sequence:

for iterator_var in sequence:

statements(s)

statements(s)

The syntax for a nested while loop statement in Python programming language is as follows:

while expression:

while expression:

statement(s)

statement(s)

Control Statements

1.Continue Statement

It returns the control to the beginning of the loop.

for i in range(0,10):

if (i==5):



break

print (i)

2. Break Statement

It brings control out of the loop

```
for i in range(0,10):
```

```
    if (i==5):
```

```
        break
```

```
    print (i)
```

2.Pass Statement

We use pass statement to write empty loops. Pass is also used for empty control statement, function and classes.

PROGRAM:

```
print("-----Program for Student Information-----")
```

```
D = dict()
```

```
n = int(input('How many student record you want to store?? '))
```

```
# Add student information
```

```
# to the dictionary
```

```
for i in range(0,n):
```

```
    x, y = input("Enter the complete name (First and last name) of student: ")
```

```
    z = input("Enter contact number: ")
```

```
    m = input('Enter Marks: ')
```

```
    D[x, y] = (z, m)
```

```
# define a function for shorting
```

```
# names based on first name
```

```
def sort():
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
ls = list()

# fetch key and value using

# items() method

for sname,details in D.items():

# store key parts as an tuple

tup = (sname[0],sname[1])

# add tuple to the list

ls.append(tup)

# sort the final list of tuples
```

```
ls = sorted(ls)

for i in ls:

# print first name and second
name print(i[0],i[1])

return
```

```
# define a function for

# finding the minimum marks

# in stored data

def minmarks():

ls = list()

# fetch key and value using

# items() methods

for sname,details in D.items():
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
# add details second element

# (marks) to the list
ls.append(details[1])


# sort the list elemnts
ls = sorted(ls)

print("Minimum marks: ", min(ls))

return


# define a function for searching
# student contact number
def searchdetail(fname):

    ls = list()

    for sname,details in D.items():

        tup=(sname,details)

        ls.append(tup)

    for i in ls:

        if i[0][0] == fname:

            print(i[1][0])

    return


# define a function for
# asking the options
def option():

    choice = int(input('Enter the operation
    detail: \n \ 1: Sorting using first name \n \
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

2: Finding Minimum marks \n \

3: Search contact number using first
name: \n \ 4: Exit\n \

Option: '))

```
if choice == 1:
```

```
# function call
```

```
sort()
```

```
print('Want to perform some other operation??? Y or N: ')
```

```
inp = input()
```

```
if inp == 'Y':
```

```
option()
```

```
# exit function call
```

```
exit()
```

```
elif choice == 2:
```

```
minmarks()
```

```
print('Want to perform some other operation??? Y or N: ')
```

```
inp = input()
```

```
if inp == 'Y':
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
option()
```

```
exit()
```

```
elif choice == 3:
```

```
first = input('Enter first name of student: ')
```

```
searchdetail(first)
```

```
print('Want to perform some other operation??? Y or N: ')
```

```
inp = input()
```

```
if inp == 'Y':
```

```
option()
```

```
exit()
```

```
else:
```

```
print('Thanks for executing me!!!!')
```

```
exit()
```

```
option()
```

Output

```
====RESTART:
```

```
C:/Users/admin/AppData/Local/Programs/Python/Python310/exp1.py === -----
```

```
Program for Student Information-----
```

```
How many student record you want to store?? 3
```

```
Enter the complete name (First and last name) of student: Ram Sharma
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Enter contact number: 9850883323

Enter Marks: 87

Enter the complete name (First and last name) of student: Geeta
Varma Enter contact number: 9920771234

Enter Marks: 67

Enter the complete name (First and last name) of student: Seeta
Shukla Enter contact number: 8856646725

Enter Marks: 56

Enter the operation detail:

- 1: Sorting using first name
- 2: Finding Minimum marks
- 3: Search contact number using first name:
- 4: Exit

Option: 1

Geeta Varma

Ram Sharma

Seeta Shukla

Want to perform some other operation??? Y or N:

Y

Enter the operation detail:

- 1: Sorting using first name
- 2: Finding Minimum marks
- 3: Search contact number using first name:
- 4: Exit



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Option: 2

Minimum marks: 56

Want to perform some other operation??? Y or N:

Y

Enter the operation detail:

1: Sorting using first name

2: Finding Minimum marks

3: Search contact number using first name:

4: Exit

Option: 3

Enter first name of student: Seeta

8856646725

Want to perform some other operation??? Y or N:N

Conclusion: the experiment effectively showcased the integration of essential data types and control structures within Python. By engaging in practical exercises, participants acquired proficiency in manipulating variables, employing loops, leveraging conditionals, and utilizing fundamental data structures.