19/8/21

Assignment-2

1. What the datatypes in python? Explain

A:- In python there are 5 different types

* numbers
* String
* List
* Tuple
* Dictionary

1. numbers:-

→ python numbers variable are created by the standard python method

```
var=382
```

→ python will automatically convert a number from one type to another if it needs

→ python will do variable conversion automatically

Eg:- message = " Good morning"
num = 85
pi = 3.14159

```
print (type (message)) # This will return string
print (type (n)) # This will return an integer
print (type (pi)) # This will return a float
```

2. String:-

→ Create string variables by Enclosing characters in quotes

→ python uses single quotes ' double quotes " and triple quotes """ to dorate literal strings

→ string can be Accessed as a whole string or substring of the compete variable using brackets [].

Eg:- var1 = ' Hello World!'
var2 = ' Rhino python'

```
print var1[0] # This will print the first character in the string an 'H'
print var2[1:5] # This will print the substring 'hinop'
```

3. List:-

-> List are a very useful variable type in python.
-> A List can contain a series of values.
-> List variables are declared by using brackets [] following the variable name

Eg:-

```
A=[ ] # This is a blank list variable
B=[1, 23, 45, 67] # This list creates an initial list of 4 numbers
C=[2,4, 'John'] # This lists can contain different variable types.
```

```
[2] my list:- [0,1,2,3]
my list[0]:= Rhino'
my list [1] = 'Grasshopper'
my list [2] = 'Flamingo'
my list [3] = 'Bongo'
Print mylist [1].
```

4. Tuple:-

-> Tuples are group of values like a list and are manipulated in similar ways.
-> In python the fixed size is considered immutable as compared to a list that is dynamic and mutable.
-> Tuples are defined by parenthesis().

Eg:- myGroup= ('Rhino' 'Grasshopper', 'Flamingo', 'Bongo').

5. Dictionary:-

-> Dictionaries in python are lists of key value pairs
-> The main operation of a dictionary is to extract a value based on the key name.
-> Dictionaries are using can also be used to sort, iterate and compare data.

Eg:-

```
room-num= {'John': 425, 'tom': 212}
room-num ['John'] = 645
Print (room-num ['tom'])
room-num ['issaac']=845
Print (room-num.keys())
print ('issac' in room-num)
```

2. Briefly. Explain history of python?

A:- → In the late 1980's history was about to be written
→ It was that time working on python started.
→ Soon after that Guido van Rossum began doing his application based work in December of 1989 by at Centrum Wiskunde & Informatica which is situated in Netherland.

→ The programming language in y which python is said to have succeeded is ABC programming language which had the interfacing with the Amoeba operating System and had the feature of Exception handling.

→ The inspiration for the name came from BBC's TV show 'Monty pythons Flying circus' as he was a big fan of TV show and also he wanted a short unique and slightly Mysterious name for his invention and hence named it python

→ when it was released, it used a lot fewer codes to Express the concepts, when we compare it with Java C++ etc. It design philosophy was quite good too. It's main objective is to provide code redability and advanced developer productivity.

3) Explain all the operators in python?

A:- python operators:-

1) Arthmatic operators
2) Relational operators
3) Logical operators
4) Bitwise operators
5) Assignment operators
6) Special operators.
7) Membership operator

1) Arthmetic operators:-
Arthmetic operators are used to perform mathamatical operations like addition, subtraction, multiplication & division

operator:- +, -, *, /, //, %, **

2) Relational operator:-
Relational operator compares the values. It either returns True or false according to the condition

operator:- >, ==, !=, >=

Eg:- a=13
b=33

a>b is false
print (a>b)

a<b is True
print (a<b)

a==b is false
print (a==b)

a!=b is True
print (a!=b)

a>=b is false
print (a>=b)

a<=b is True
print (a<=b)

O/p:-
False
True
False
True
False
True

3) Logical operator:-
Logical operators perform Logical AND, Logical OR, and Logical not operations.

operator:- and, or, not

Eg:- a= True
b= False

print a & b is false
print (a and b)

print a or b is True
print (a or b)

print not a is false.
print (not a)

o/p:-
False
True
False.

4) Bitwise operator:-
Bitwise operator acts on bits and perform bit by bit operation

operator:- &, |, ~, ^, >>

o/p:- a=10
b=4
# print bitwise AND operation
print (a&b)
print bitwise OR operation
print (a|b)
print bitwise NOT operation
print (~a)
print bitwise XOR operation
print (a^b)
print bitwise right shift operation
print (a>>2)
print bitwise left shift operation
print (a<<2)

o/p:- 0
14
-11
14
2
40

5) Assignment operators:- Assignment operators are used to assign values to the variables

operator:- =, +=, -=, *=, /=, %=, //=, **=, &=, |=, ^=, >>=,

6) Special operators:- There are some special type operators like Identity operators is and is not are the identity operators both are used to check if two values are located on the same part of the memory. Two variables that are Equal does not imply that they are identical.

is - True if the operands are identical
is not - True if the operands are not identical

Eg:-

```
a1=3
b1=3
a2='Greek for Greeks'
b2='Greek for Greeks'
a3=[1,2,3]
b3=[1,2,3]
print(a1 is not b1)
print(a2 is b2)
# output is false, since lists a
print(a3 is b3)
```

O/p:-

```
false
True
false
```

7)

Membership operators:-

Member ship operator is and not in are the member-ship operators; used to test whether a value or-variable is in a sequence

in - True if value is found
not in - True if value is not found in the sequence

Eg:-

```
x='Geeks for Geeks'
y={3:'a', 4:'b'}
print('G' in x)
print('geeks' not in x)
print('Geeks' not in x)
print(3 in y)
print('b' in y)
```

O/p:-

```
True
True
False
True
False
```

(9) Explain the features of python.

A:- features of python:-

1) Easy to learn and use python is Easy to learn and use
2) Expressive Language
3) Intepreted Language
4) Cross-platform Language
5) Free and open Source
6) Object-oriented Language
7) Extensible
8) Large Standard Library
9) GUI programming Support
10) Integrated

1. Easy to learn and use:-

python is Easy to learn and use. It is developer friendly and high level programming language

2. Expressive Language:-

python language is more Expressive means that it is more understandable and readable

3. Intepreted Language:-

python is Interpreted language interpreter Executes the code line by line at a time. This makes debugging Easy thus Suitable for beginners.

4. Cross-platform Language:-

python can run Equally on different platforms Such as windows, Linux, unix and macintosh etc

5. Free and open Source:-

python language is freely available at official web address. The Source-code is also available Therefore it is open Source.

6. Object-oriented Language:-

python Supports object oriented language and concepts of classes and objects come into Existence

7) Extensible:-

It implies that other languages such as c/c++ can be used to compile the code and thus it can be used further in our python code.

8) Large Standard Library:-

Python has a large and broad library and provides rich set of module and functions for rapid application development

9) GUI programming support:-

Graphical user interfaces can be developed by using python

10) Integrated:-

It can be Easily integrated with languages like c, c++, Java. etc

Justify why python is interactive interpreted language, unlike c/c++ etc, python is an interpreted object oriented programming language -- unlike c language, which is a compiled programming language. The compiler translates the whole code in one-go rather than line-by-line. This is the Errors are listed during compilation only