Date: 19-06-2020

Page No.

Python-01 Assignment

1. What the datatypes in Python? Explain.

Every value in Python has a datatype. Since everything is an object in Python Programming, datatypes are actually classes and variables are instance of these classes.

There are various datatypes in Python. Some of the important types are listed below.

→ Python Numbers:

Integers, floating point numbers and Complex numbers fall under Python numbers category. They are defined as $\boxed{\text{int}}$, $\boxed{\text{float}}$ and $\boxed{\text{Complex}}$ classes in Python.

we can use the $\boxed{\text{type()}}$ function to know which class a variable or a value belong to. Similarly, the $\boxed{\text{instance()}}$ function is used to check if an object belongs to a particular class

eg:

```
a=5
Print(a, "is a type", type(a))
a=2.0
Print(a, "is a type", type(a))
a=1+2j
Print(a, "is Complex number?", isinstance(1+2j, Complex))
```

Scanned with CamScanner

Date:

Page No.

→ Python List.

List is an ordered collection sequence of items. It is one of the most used datatype in Python and very flexible. All the items in a list do not need to be of the same type. Declaring a list is pretty straight forward. Items seperated by commas are enclosed within brackets '[ ]'. The index starts from 0 in Python.

eg
```
a = [5,10,15,20,25,30,35,40]
Print ("a[2] =", a[2])
Print ("a[0:3] =", a[0:3])
Print ("a [5:] =", a [5:])
```

Lists are mutable, meaning, the value of elements of a list can be altered.

→ Python tuple

Tuple is an ordered sequence of items same as a list. The only difference is that tuples are immutable, Tuples once created cannot be modified. Tuples are used to write-Protect data and are usually faster than lists as they cannot change dynamically.

It is defined within Parenthesis ( ) where items are seperated by commas.

we can use the slicing operator [ ] to extract items but we cannot change its value

eg : t = (5, 'program', 1+3j)
```
    Print ("t[1] =", t[1])
    Print ("t[0:3] =", t[0:3])
    t[0]=10.
```

→ Python Strings

String is sequence of Unicode characters. We can use single quotes or double quotes to represent strings. Multi-line strings can be denoted using triple quotes, $\boxed{' ' '}$ or $\boxed{" " "}$

eg: S= "This is a string"
```
    Print(s).
    S='''A multiline string'''
    Print (s)
```

→ Python set

Set is an unordered collection of unique items. Set is defined by values seperated by comma inside braces { }. Items in a set are not ordered.

eg:
```
a={5,2,3,1,4}
Print ("a=",a)
Print (type (a))
```

We can Perform set operations like Union, intersection on two sets. Sets have Unique values. They eliminate duplicates.

→Python Dictionary

Dictionary is an Unordered collection of Key-Value Pairs It is generally used when we have a huge amount of data. Dictionaries are optimized for retrieving data. Dictionaries are defined within braces {} with each items being a pair in the form $\boxed{\text{Key:value}}$. Key and value Can be of any type.

eg:
```
d={1:'value','Key':2}
Print (type(d))
Print ("d[1]=",d[1])
Print ("d['Key']=",d[Key]);
Print ("d[2]=",d[2])
```

DATE: / /
PAGE:

Briefly explain history of Python.

In the late 1980s, history was about to written. It was that time when working on Python started. Soon after that, Guido Van Rossum began doing its application based work in December of 1989 by at Centrum Wiskunde & information (CWI) which is Suituated in Netherland. It was started firstly as a hobby Project because he was looking for an intersting Project to keep him occupied during chirstmas.

Python is a general-Purpose interpreted, interactive, object-oriented, and high level programming language.
Python is named after a TV show called 'Monty Python's Flying circus' and not after Python - the snake.

3 Explain all the Operators in Python
Operators are special symbols that represent computations like addition and multiplication. The values the operator is applied to are called operands.
The operator +, -, *, /, and ** perform addition, subtraction, multiplication, division and exponentiation, as in the following examples:
20+32
hour-1
hour*60+minute.

```
minute/60
5**2
(5+9)*(15-7)
```

There has been a change in the division operator between Python 2.x and Python 3.x. In Python 3.x, the result of this division is a floating point result:

```
>>> minute=59
>>> minute/60
0.9833333333333333
```

The division operator in Python 2.0 would divide two integers and truncate the result to an Integer:

```
>>> Minute=59
>>> minute/60
0
```

To obtain the same answer in Python 3.0 is used floored (//integer) division.

```
>>> minute=59
>>> minute//60
0
```

In Python 3.0 integer division function much more as you would expect if you entered the expressions on a calculator.

→ Arithematic operators

Arithematic operators are used with numeric values to Perform Common mathematical operations

operator	Name	Example.
+	Addition	x + y
-	subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
%	Modulus	x % y
**	exponential	x ** y
//	Floor Division	x // y

→ Assignment operators

Assignment operators are used to assign value to variables

operator	~~Name~~	Example	Same as
=		x = 5	x = 5
+=		x += 5	x = x + 5
-=		x -= 5	x = x - 5
*=		x *= 5	x = x * 5
/=		x /= 5	x = x / 5
%=		x %= 5	x = x % 5
//=		x //= 5	x = x // 5
**=		x**x = 5	x = x**x 5

operator	Example	Same as
&=	x&=5	x=x&3
|=	x|=5	x=x|5
^=	x^=5	x=x^5
>>=	x>>=5	x=x>>5
<<=	x<<=5	x=x<<5

→ Comparision operators

Comparision operators are used to Compare two values

Operator	Name	Example
==	Equal	x==y
!=	not equal	x!=y
>	Greaterthan	x>y
<	Lesser than	x<y
>=	Greater than equalto	x>=y
<=	lesser than equalto	x<=y

→ logical operators

logical operators are used to combine the conditional statements.

Page No.

operator	Description	Example
and	Returns True if both statements are true	x<5 and x<10
or	Returns True if one of the statement is true	x<5 or x<4
not	Reverse the result, returns False if the result is true	not(x<5 and x<10)

→Identity operators

Identity operators are used to Compare the objects.

operator	Description	Example
is	Returns True if both variables are the same object	x is Y
is not	Returns True if both variables are not the same object	x is not Y

→Bitwise operators

Bitwise operators are used to Compare (binary) numbers.

Operator	Name	Description
&	AND	Sets each bit to 1 if both bits are 1
|	OR	sets each bit to 1 if one of the bit is 1
^	XOR	sets each bit to 1 if only one of two bits is 1
~	NOT	Inverts all bits

Date:

Page No.

operator	Name	Description
<<	zero fill left shift	shift left by Pushing zero in from the right and let the leftmost bits fall off
>>	Signed right shift	shift right by Pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off.

4. Explain the features of Python

Python is a dynamic, high-level, free-open source and interpreted Programming language.

Features in Python:

* Easy to code: It is a high-level Programming language. Python is very easy to learn language as Compared to other language like C, C#, java, javascript etc.

* Free and open Source: It is freely available at official website. Since, it is a open Source code is also available to the Public

DATE: / /
PAGE:

* object - Oriented language : Python supports object Oriented language and concept of classes, objects encapsulation etc.

* GUI Programming support: GUI is the most Popular option for Creating graphical apps with Python.

* High-level language: when we write Programs in Python, we do not remember the system architecture, nor do we need to manage the memory

* Extensible feature: we can write and Compile python code in C or C++ language

* Python is Portable language.

* Python is integrated language: we can easily integrate Python with other languages like C/C++

* Interpreted language: Python Code executes line by line at a time. The Source Code of Python is Converted into an immediate form called bytecode.

* Large-standard library: Python has large library which Provides rich set of module and functions.

5. Justify why Python is interactive interpreted language?

Python is interactive. When a Python statement is entered, and is followed by the Return key, if appropriate the result will be printed on the screen, immediately in the next line. This is Particularly advantageous in the debugging process. In interactive mode of operation.

Python is an interpreted object-Oriented Programing language. By interpreted it is meant that each time a Program is run the interpreter checks through the code for errors and then interprets the instructions into machine-readable bytecode.

An interpreter is a translator in Computer's language which translates the given code line-by line in machine readable bytecodes. And if any error is encountered it stops the translation Until the error is fixed. Unlike C language, which is a Compiled Programming language. The Computer translates the whole code in one-go rather than line-by-line.