19/06/2020

Assignment - 2

Srinija. D
321910305002

1

1. What are the datatypes in Python? Explain.

A. There are various datatypes in python. They are

1. Integer - int
2. Float - float
3. Complex No. - complex
4. Boolean - bool
5. String - str

Integer datatype: This data type takes in positive or negative whole numbers.

Eg.

```
a = 10
print(a, type(a))
```

O/P: 10 <class 'int'>

Float data type: This datatype takes in floating point values.

Eg.

```
a = 5.6
print(a, type(a))
```

O/P: 5.6 <class 'float'>

Complex data type: This data type takes in complex numbers in the form of
a + jb (or) a + bj.
j is used to represent the complex part.

Eg. b = 5 + 8j
print(b, type(b))

O/p: (5+8j) <class 'complex'>

We can perform various operations such as addition, subtraction, multiplication, division on two complex numbers.
Note that we cannot convert a complex number into an integer or a floating point data type.

Eg. b1 = 5 + 2j
b2 = 7 + 8j
print (b1+b2, b1 - b2, b1 * b2, b1/b2)

O/p: (12+10j), (-2-6j) (19+54j) (0.45132-0.23008j)

Boolean data type: This data type returns two values. Either True or False.

Eg a = 23<7
print(a)

O/p: False

a = (34==34)
print (a, type(a))

O/p: True <class 'bool'>

String data type: It is a collection (or) combination of data types either enclosed in single quotes or double quotes (' ' or " ").

Eg.

```
a = "Hello World!"
print(a, type(a))
```

O/p: Hello World! <class 'str'>

```
a = "Srinija"
print(a)
```

O/p: Srinija

We can perform two operations on strings: Indexing and slicing.

To determine the length of a string, we can use the len() function.

```
Eg. str = "Hello World!"
    length = len(str)
    print(length)
```

O/p: 12

```
str[0:10:2]
```

O/p: 'HloWr'

```
str[0:4]
```

O/p: 'Hell'

```
str[6:]
```

O/p: World!

```
str[1]
```

O/p: e

```
str[-1::-1]
```

O/p: '!dlroW olleH'

4 19/06/2020

2. Briefly explain the history of Python.

Python was designed by Guido Van Rossum in 1991. It was named after a comedy show called 'Monty Python's Flying Circus'

Python is a widely used general-purpose, interpreted, high-level, portable, interactive, extensive and embeddable, also object oriented.

Python is well known for its simple, free and open source system and libraries. There are a lot of packages and libraries to choose from.

3. Explain all the operators in Python.

Python has a wide range of arithmetic operators. They are:

+	Addition	a+b
-	Subtraction	a-b
*	Multiplication	a*b
/	Division	a/b
//	Floor div.	a//b
%	Modulus	a%b
**	Exponent/Power	a**b

Examples:

x = 10,
y = 5

Example:

```
a, b = 20, 40
print(a+b, a-b, a*b, a/b, a//b)
```

O/P: 60 -20 800 0.5 20

~~a~~ print(a//b, a**b)

O/P: 0 1099511627776 X 10^{40}

Python also has various assignment operators:

=	Equal to	a = 5
+=	Add/Increment	a += 5
-=	decrement	a -= 5
*=	Multiply by itself	a *= 5
/=	Division by itself	a /= 5

Examples:

a = 5.

print(a, a+=2, a-=2, a*=2, a/=2)

a = 5

a += 3

print(a) ⇒ O/P: 8

a -= 3

print(a) ⇒ O/P: 2

a /= 2

print(a) ⇒ O/P: 1.666666667

a *= 2

print(a) ⇒ O/P: 15

Python has a set of comparison operators. They are:

==	Equal	x == y
!=	Not equal	x != y
>	grater than	x > y
<	less than	x < y
>=	grat-than or =	x >= y
<=	less than or =	x <= y

Examples:

```
x, y = 5, 10
print(x == y)   => o/p: False
print(x != y)   => o/p: True
print(x > y)    => o/p: False
print(x < y)    => o/p: True
print(x >= y)   => o/p: False
print(x <= y)   => o/p: True
```

There is a special operator called 'in'. It tells whether a particular object is present in the string or a bigger object or not.

Example: x in y

```
y = "Hello there! How are you?"
x = "there"
print(x in y)
```

o/p: True

19/06/2020

4. Explain the features of Python.

Python is a programming language. The features are:

1. Simple
2. Easy to learn
3. Free and open source
4. High-level language
5. Beginner's language
6. Portable
7. Interactive
8. Interpreted
9. Extensible
10. Object Oriented
11. Embeddable
12. Extensive libraries
13. Has many databases
14. GUI programming and
15. Scalable.

5. Justify why Python is an interactive and an interpreted language.

Interactive: Python is an interactive language because we can execute our programs line-by-line using the Python prompt or compiler. It is called interactive because interactive mode is a command line prompt which gives immediate output for each statement.

This happens while running previously fed statements in the active memory.

Interpreted:

1. Python is interpreted because a Python program directly runs from the source code.

2. Python converts source code written by us into machine language and it is executed. So, Python is an interpreted language.

3. Also, it is processed at run-time by the interpreter.

4. It can prompt and interact with the interpreter directly to write the programs.