

PYTHON ASSIGNMENT-2

K. Saketh
221210201027
EI

(1) What are the data types in python? Explain.

(A) Every value in python has data type. Since everything is an object in python programming, data types are actually classes and variables are instances of these classes.

There are various data types in python. Five of the important data types are listed below:

(a) Numeric:-

A numeric value is any representation of data which has a numeric value. Python identifies 3 types of numbers.

i) Integer: It includes positive (or) negative whole numbers.

ii) Float: It includes any real number with a floating point representation in which a fractional component.

iii) Complex Number:- It includes the combination of real numbers & imaginary component represented as $x+yj$, where x, y are real numbers & value of j is $\sqrt{-1}$.

(b) Boolean:-

Data with one of two built-in values True (or) False. In this 'T' and 'F' are of upper case. true and False are not valid in python will throw an error.

(c) Sequence Type:-

A sequence is an ordered collection of similar (or) different data types. python has the following built-in sequence data types.

- i) String: A string value is a collection of one or more characters put in single, double (or) triple quotes.
- ii) List: A list object is an ordered collection of one (or) more data items, not necessarily of the same type, put in square brackets.
- iii) Tuple: A Tuple object is an ordered collection of one or more data items, not necessarily of the same type, put in parentheses.

(d) Dictionary:-

A dictionary object is an unordered collection of data in a key:value pair form. A collection of such pairs is enclosed in curly brackets.

Ex:- {1:"Steve", 2:"Bill", 3:"Ram", 4:"Farah"}.

(Q) Briefly explain history of python.

(A) Python is an interpreted, high-level, general-purpose programming language. Created by "Guido Van Rossum" and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant white space.

Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large scale projects.

Python is dynamically typed and garbage-collected. It supports multiple-programming paradigms, including structured, object-oriented and functional programming.

Python is often described as "batteries included" language due to its comprehensive standard library.

Python was conceived in the late 1980's as a successor

to the ABC language.

Python 2.0, was released in 2000, introduced features like list comprehension and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language which is not completely backward-compatible and many of the python 2 codes does not run unmodified on python 3.

The python 2 language was officially discontinued in 2020 and python 2.7.18 is the last python 2 released and therefore, the last python 2 released.

No more security patches or other improvements will be released for it. With python 2's end-of-life, only python 3.5.x and later are supported.

Python interpreters are available for many operating systems. A global community of programmers develops and maintains CPython, an open source reference implementation.

A non-profit organization, the Python Software Foundation, manages and directs the resources for Python and CPython developments.

(3) Explain all the operators in python.

(A) Operators are special symbols in python that carry out arithmetic or logical computations. The value that operator operates on is called operand.

(a) Arithmetic Operators:-

Arithmetic Operators are used to perform mathematical operations like +, -, *, /, etc. ,

Operator	Meaning	Example
+	add 2 operands (or) unary plus	$x+y$
-	Subtract right operand from the left (or) unary minus	$x-y$
*	multiplier 2 operands	$x*y$
/	divides left operand by the right one	x/y
%.y	gives the remainder of the division of left operand by right	$x\%.y$
//	Division that results into whole number adjusted to the left in number line.	$x//y$
**	Left operand raised to the power of right operand	$x**y$

(b) Comparison Operators:-

Comparison Operators are used to compare values. It returns either True (or) False according to the condition.

Operator	Meaning	Example
>	Greater than - True if left operand is greater than the right	$x > y$
<	Less than - True if left operand is less than the right	$x < y$
==	True if both operands are equal	$x == y$

$!$	True if operands are not equal	$x != y$
$>$	True if left operand is greater than or equal to the right	$x > y$
$<=$	True if left operand is less than or equal to the right	$x <= y$

(c) Logical Operators:-

Logical Operators are the and, or, not operators.

Operator	Meaning	Example
And	True if both the operands are true	$x \text{ and } y$
Or	True if either of the operand is true	$x \text{ or } y$
not	True if operand is false	not x

(d) Bit-wise Operators:-

Bitwise operators act on the operands as if they were strings of binary digits. They operate bit by bit, hence the name. For example, 2 is 10 in binary & 7 is 111.

Operators	Meaning	Example
$\&$	Bitwise AND	$x \& y = 0$
$ $	Bitwise OR	$x y = 11$
$-$	Bitwise NOT	$-x = -1$
\wedge	Bitwise XOR	$x \wedge y = 11$
$>>$	Bitwise right shift	$x >> 2 = 2$
$<<$	Bitwise left shift	$x << 2 = 40$

(e) Assignment Operators:-

Assignment operators are used in python to assign value to variable. $a=5$ is a simple assignment operator that assign value 5 on the right to the variable a on the left. There are various compound operators in python like $a+=5$ that add to the variable and later assigns the same. It is equivalent to $a=a+5$.

Operator	Example	Equivalent
$=$	$x=5$	$x=5$
$+=$	$x+=5$	$x=x+5$
$-=$	$x-=5$	$x=x-5$
$*=$	$x*=5$	$x=x*5$
$/=$	$x/=5$	$x=x/5$
$\div=$	$x\div=5$	$x=x\div 5$
$//=$	$x//=5$	$x=x//5$
$**=$	$x**=5$	$x=x**5$
$\&=$	$x\&=5$	$x=x\&5$
$ =$	$x =5$	$x=x 5$
$\wedge=$	$x\wedge=5$	$x=x\wedge 5$
$>>=$	$x>>=5$	$x=x>>5$
$<<=$	$x<<=5$	$x=x<<5$

(F) Special Operators:-

Python language offers some special type of operators like the identity operators they are described below with examples.

(g) Identity Operators:-

`is` and `is not` are the identity operators in python. They are located on the same part of the memory. Two variables that are equal does not imply that they are identical.

Operator	Meaning	Example
<code>is</code>	True if the operands are identical	<code>20 is true</code>
<code>is not</code>	True if the operands are not identical	<code>20 is not 10</code>

(h) Membership Operators:-

`'in'` and `'not in'` are the membership operators in python. They are used to test whether a value or variable is found in a sequence.

Operator	Meaning	Example.
<code>in</code>	True if value/variable is found in the sequence	<code>5 in x</code>
<code>not in</code>	True if value/variable is not found in the sequence.	<code>5 not in x</code>

(4) Explain the features of python.

(A) Python provides lots of features that are listed below.

(1) Easy to learn and use.

Python is high level programming language. It is easy to learn and use. It is developer-friendly and high level programming language.

(2) Expressive language.

Python language is more expressive that means it is more understandable and readable.

(3) Interpreted language.

Python is an interpreted language i.e. interpreted executes the code line by line at a time. This makes debugging easy and thus suitable for beginners.

(4) Cross-platform language.

Python can run equally on different platforms such as windows, linux, Unix and Macintosh etc. So we can say that python is a portable language.

(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, objects and encapsulation etc. come into existence.

(7) Extensible

Python is an Extensible language. 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 using python.

(10) Integrated.

It can be easily integrated with the languages like C, C++ & JAVA etc.,

(6) Justify why python is interactive interpreted language.

(A) Unlike C/C++ etc., python is an interpreted object-oriented 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 byte code.

An interpreter is a translator in computer's language which translates the given code line-by-line into machine-readable byte codes. 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 compiler translates the whole code in one-go rather than line-by-line. This is the reason why in C language, all the errors are listed during compilation only.

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 used in a similar way as the UNIX command line or the terminal. Interactive python is very much helpful for the debugging purpose.

It simply returns the ">>>" prompt on the corresponding output of the statement if appropriate and returns error for incorrect statements.

In this way if you have any doubts like: whether the syntax is correct or the module you are importing exists or anything like that, you can be sure within seconds using python interactive mode.