

The use of or operator returns: True
 The use of or operator returns: True
 The use of not operator on x returns: False
 The use of not operator on y returns: True

Explanation: The use of “and” operation on x and y returns “False” since both are not True. However, “x” and “x” returns True since both the conditions are True. Similarly “y” and “y” returns “False” since both the conditions are “False”. The use of “or” operator returns “True” since at least one of the conditions is “True”. The “not x” returns False (opposite of True) and “not y” returns True (opposite of False).

1.5.6 Operators Precedence

Operator precedence determines the grouping of terms in an expression and decides how an expression is evaluated. Certain operators have higher precedence than others. For example, $x = 7 + 3 * 2$; here, x is assigned 13, not 20. Since, the multiplication operator has a higher precedence than the addition operator, so 3 is first multiplied with 2 and then adds into 7. The following table shows operator precedence in descending order.

Operator	Details
()	Parenthesis
**	Exponential
/ * // %	Division, Multiplication, Integer Division, Modulus
+ -	Addition and Subtraction
>, >=, <, <=, ==, !=	Relational Operators
=, +=, -=, **=, /=, //-, %=	Assignment Operators
not or and	Logical and Boolean operators

The operators which have a higher precedence appear at the top of the table and those with the lowest appear at the bottom. This means that within an expression, higher precedence operators will be evaluated first. The following program shows the impact of operator precedence in Python.

```
#Program to show utility of operator precedence
x=20
y=10
z=5
print("Parenthesis has a highest precedence:",(x+z)*(x+y))
print("Multiplication has higher precedence than addition:",x+z*y)
print("Relational operators has a higher precedence than logical operators:",x>z or y<z)
-----
In [1]: runfile(...)
Parenthesis has a highest precedence: 750
Multiplication has higher precedence than addition: 70
Relational operators has a higher precedence than logical operators: True
```

Explanation: In the first example “x+z” and “x*y” are evaluated first since they are inside the parenthesis and are then multiplied to each other. Hence, the result is: $25*30= 750$. In the second example, since, multiplication has higher precedence than addition, hence multiplication of z and y is done first and then added to x. Hence, the result is $20+50 = 70$. Similarly, in the last case since relational operators have a higher precedence than logical, hence “x>z” and “y<z” are evaluated first and then “or” operator is used. Since x>z is True and y<z is False, hence “or” operator used on True and False returns True.

1.6 LIBRARIES IN PYTHON

Python has a rich collection of functions that comes along with Python software. Besides, Python has an extensive list of functions which are accessible only when the user imports the required library. Python library is a collection of functions and sub packages. Functions related to one domain are grouped inside a library and when the user wants, a particular library can be accessed using import statement in Python. There are thousands of libraries for Python, written by many authors. Some of these libraries implement specialized statistical methods, others give access to arrays, and others are designed to create visualization effects etc. The user can import the respective libraries in the program; depending on the requirement to access particular functions. Some of the common libraries include:

1. **numPy:** The most fundamental library around which the scientific computation stack is built is numPy (Numerical Python). It provides an abundance of useful features for large, multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays.
2. **pandas:** A Python library designed to work with labeled and relational data is pandas. It designed for quick and easy data manipulation, aggregation, and visualization. The pandas is a spreadsheets for Python and it is able to describe the data efficiently. It can do grouping and pivot tables on larger data than most spreadsheet programs.
3. **matplotlib:** A Python library that is tailored for creating simple and powerful visualizations with ease is matplotlib. Line plots, Scatter plots, Bar charts and histograms, Pie charts, Stem plots, Contour plots, Quiver plots, Spectrograms etc. Different formatting styles like title, labels, grids, legends etc. are also available in this library.
4. **seaborn:** The seaborn library is mostly focused on the visualization of statistical models; such visualizations include heat maps and those that summarize data but still depict overall distributions.
5. **scipy:** The scipy contains modules for linear algebra, optimization, integration, and statistics. It provides efficient numerical routines as numerical integration, optimization, and many others via its specific sub modules. It adds significant power to the interactive Python session by providing the user with high-level commands and classes for manipulating data.
6. **NLTK:** The name of this suite of libraries stands for Natural Language Toolkit and, as the name implies, it used for common tasks associated with symbolic and statistical natural language processing. The functionality of NLTK allows a lot of operations such as text tagging, classification, and tokenizing, name entities identification,

building corpus, stemming, semantic reasoning etc. All of these building blocks allow for building complex research systems for different tasks; for example, sentiment analytics and summarization.

7. statsmodels: The statsmodels library for Python enables its users to conduct data exploration via the use of various methods of estimation of statistical models and perform statistical assertions and analysis. Among its many useful features are descriptive and result statistics via the use of linear regression models and various estimators.

As discussed earlier, the user can import the respective library depending upon his requirement of necessary function. However, there are three different ways to import functions and library.

- (i) `from library import *`: This approach imports all the functions from the library in the program. This provides more opportunities for name collisions and hence makes code less maintainable. Through this approach, we can call function directly from a library.
- (ii) `import library`: This approach also imports all the functions from the library. The difference between the two approaches is that through this approach, we need to specify library name before calling a function and through the previous approach, the function can be called directly by specifying the function name.
- (iii) `from library import function1,function2,...`: This is considered to be the best approach, since it imports only the required functions from library, which makes the code more manageable.

SUMMARY

- Python is an interpreted programming language and software environment for statistical analysis and data visualization and reporting.
- Programming in an easy way can be done using Anaconda or PyCharm software. Anaconda software include Jupyter Notebook and Spyder IDE.
- A single line comment start with a hash sign (#) in the start of line and Multi-line comments can be created using triple double quotes (""") before and at the end of block.
- Python is a case-sensitive programming language
- Python does not allow punctuation characters such as @, \$, and % within identifiers.
- The reserved words may not be used as constants or any other identifier names.
- Input in a Python program is taken using the `input()` and output is displayed using `print()` function.
- The `input()` function accepts only string data. Python accepts numeric input from the user in the form of an integer or float number using the `int()` and `float()` function respectively along with `input()`.
- Python provides a distinct feature of taking input from the user using `eval()` function which considers the data type according to the nature of input provided.
- Unlike other programming language like C, C++, it is possible to give multiple inputs using one single input statement.

- Another keyword argument is “sep” which allows us to control how the print function visually separates the arguments it displays.
- Python language is rich in built-in operators and provides the following types of operators: arithmetic operators, relational operators, assignment operators, logical operators and Boolean operators.
- Python library is a collection of functions and sub packages. Some of the common libraries include numpy, scipy, pandas, statsmodel, matplotlib, seaborn, NLTK etc.

QUESTIONS

1. Discuss the use of different logical operators in Python with example.
2. Is it possible to give multiple inputs using single statement? If yes, explain with an example.
3. Differentiate between the int() , float() and the eval() functions used for taking input from user.
4. What is the significance of operator precedence in Python?
5. What are the common libraries available in Python? Explain their importance

MULTIPLE CHOICE QUESTIONS

1. This software does not help in Python programming.

(a) anaconda	(b) spyder
(c) jupyter	(d) visual
2. _____ is not a keyword.

(a) answer	(b) int
(c) eval	(d) print
3. A multi-line comment in Python is written using _____ at the start and end of block.

(a) //	(b) /*
(c) """	(d) #
4. The result of `print("print(\"An\", \"Example\", sep='--')` is:

(a) An Example	(b) An Example --
(c) -- An --Example	(d) An--Example
5. _____ is not a library in Python.

(a) pandas	(b) shapiro
(c) scipy	(d) numpy



1. (d) 2. (a) 3. (c) 4. (d) 5. (b)

ANSWERS