|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Finolex Academy of Management and Technology, Ratnagiri | | | |
| **Department of Information Technology** | | | |
| **Subject:** | **Python Lab. (ITL404)** | | | |
| **Class:** | **SE IT / Semester – IV (Rev-2019 ‘C’) / Academic year: 2020-21** | | | |
| **Name of Student:** | **Borkar Afiya Ayub** | | | |
| **Roll No:** | **06** | | **Date of performance (DOP) :** |  |
| **Assignment/Experiment No:** | | **01** | **Date of checking (DOC) :** |  |
| **Title: Program to demonstrate basic data types and arithmetic operators** | | | | |
| **Marks:** | |  | **Teacher’s Signature:** |  |

**1. Aim**: To understand basics of Python programming.

**2. Prerequisites**:

1. Basics of programming disciplines.

**3. Hardware Requirements**:

1. PC with minimum 2GB RAM

**4. Software Requirements:**

1. Windows / Linux OS.
2. Python 3.6 or higher

**5. Learning Objectives:**

1. To understand Python as a software development platform.
2. To understand elementary building blocks of Python such as- expressions, variables, keywords, basic data types, and operators

**6. Learning Objectives Applicable: LO 1**

**7. Program Outcomes Applicable: PO 1**

**8. Program Education Objectives Applicable: PEO 1**

**9. Theory:**

**Data types:**

Data types are the classification or categorization of data items.

It represents the kind of value that tells what operations can be performed on a particular data.

Following are the standard or built-in data type of Python:

**Integers** – This value is represented by int class. It contains positive or negative whole numbers (without fraction or decimal). In Python there is no limit to how long an integer value can be.

E.g. type(19)

O/P : <class 'int'>

**Float** – This value is represented by float class. It is a real number with floating point representation. It is specified by a decimal point. Optionally, the character e or E followed by a positive or negative integer may be appended to specify scientific notation.

E.g. type(4.05)

O/P : <class 'float'>

**Complex Numbers** – Complex number is represented by complex class. It is specified as (real part) + (imaginary part)j. For example – 6+7j

E.g. type(6+9j)

O/p : <class 'complex'>

**String** - In Python, Strings are arrays of bytes representing Unicode characters. A string is a collection of one or more characters put in a single quote, double-quote or triple quote. In python there is no character data type, a character is a string of length one. It is represented by str class.

E.g. type("FAMT")

O/P : <class 'str'>

# Boolean - Data type with one of the two built-in values, True or False. Boolean objects that are equal to True are truthy (true), and those equal to False are falsy (false). But non-Boolean objects can be evaluated in Boolean context as well and determined to be true or false. It is denoted by the class bool.

E.g. type(True)

O/P : <class 'bool'>

**Arithmetic Operators:**

An arithmetic operator is a mathematical function that takes two operands and performs a calculation on them. They are used in common arithmetic and most computer languages contain a set of such operators that can be used within equations to perform a number of types of sequential calculation. Arithmetic operators are used to perform mathematical operations like addition, subtraction, multiplication, etc.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.  No. | Operator  Symbol | Operator  Name | Description | Syntax |
| 1 | + | Addition | adds two operands. | x + y |
| 2 | - | Subtraction | subtracts two operands. | x - y |
| 3 | \* | Multiplication | multiplies two operands. | x \* y |
| 4 | / | Division (float) | divides the first operand by the second. | x / y |
| 5 | // | Division (floor) | divides the first operand by the second. | x // y |
| 6 | % | Modulus | returns the remainder when first operand is divided by second. | x % y |
| 7 | \*\* | Power | Returns first raised to power second. | x \*\* y |

**Bitwise Operators:**

In Python, bitwise operators are used to perform bitwise calculations on integers. The integers are first converted into binary and then operations are performed on bit by bit, hence the name bitwise operators. Then the result is returned in decimal format.

**Note:** Python bitwise operators work only on integers.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.  No. | Operator  Symbol | Operator  Name | Description | Syntax |
| 1 | & | Binary AND | Operator copies a bit to the result if it exists in both operands | a & b |
| 2 | | | Binary OR | It copies a bit if it exists in either operand. | a | b |
| 3 | ^ | Binary XOR | It copies the bit if it is set in one operand but not both. | a ^ b |
| 4 | ~ | Binary Ones Complement | It is unary and has the effect of 'flipping' bits. | ~a |
| 5 | << | Binary Left Shift | The left operands value is moved left by the number of bits specified by the right operand. | a << 2 |
| 6 | >> | Binary Right Shift | The left operands value is moved right by the number of bits specified by the right operand. | a >> 2 |

**Program:**

x = int(input("Enter the value of x: "))

y = int(input("\nEnter the value of y: "))

print("\nx : ",x)

print("y : ",y)

print("\n\*\*\*Arithmetic Operators\*\*\*")

print("\nx+y = ",x+y)

print("x-y = ",x-y)

print("x\*y = ",x\*y)

print("x%y = ",x%y)

print("x/y = ",x/y)

print("x\*\*y = ",x\*\*y)

print("x//y = ",x//y)

print("\n\*\*\*Bitwise Operators\*\*\*")

print("\nx&y = ",x&y)

print("x|y = ",x|y)

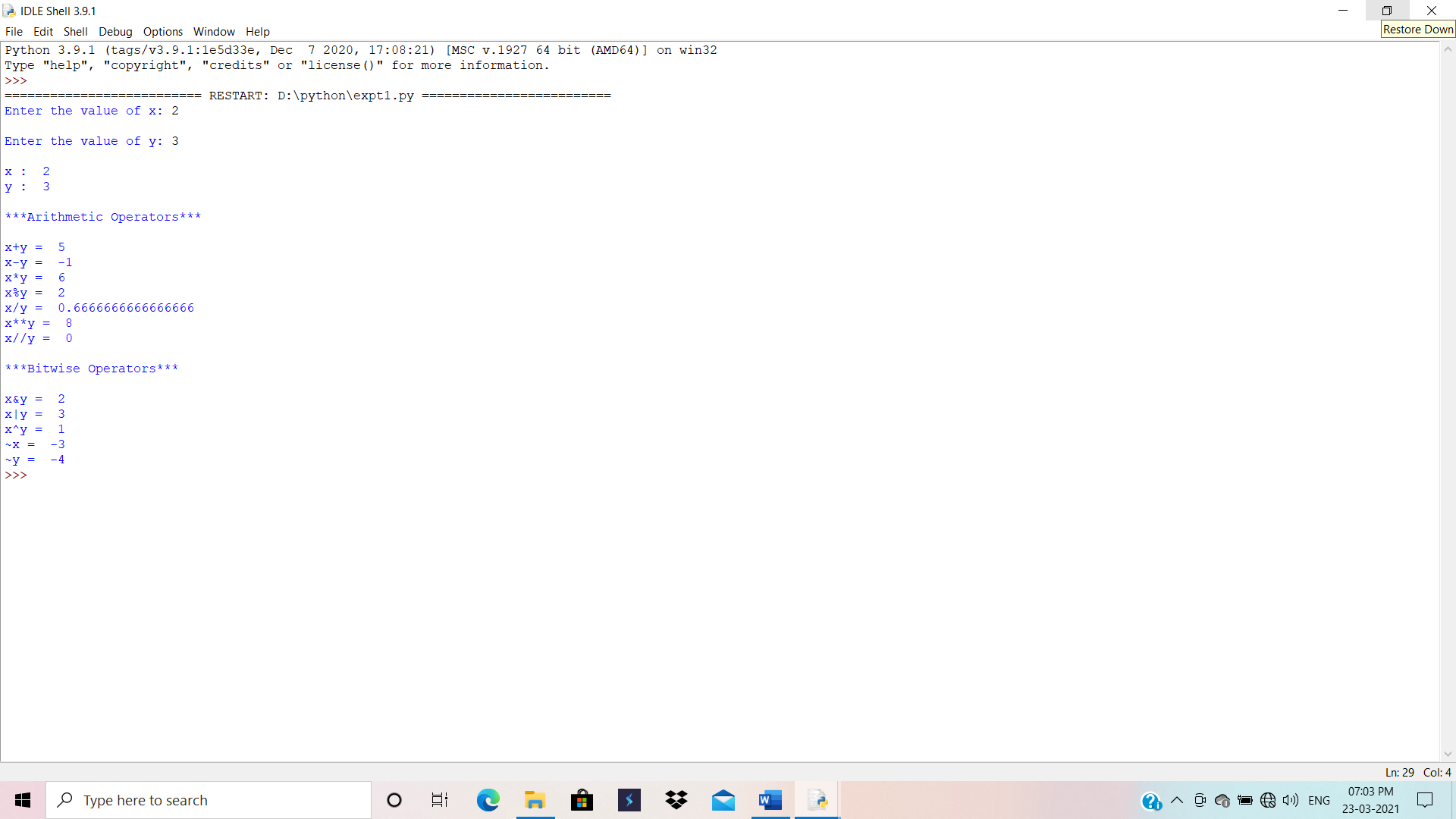
print("x^y = ",x^y)

print("~x = ",~x)

print("~y = ",~y)

**10. Results:**

**Output:**



**11. Learning Outcomes Achieved:**

1. Understood Python as a software development platform.

2. Understood elementary building blocks of Python such as- expressions, variables, keywords, basic data types, and operators.

**12. Conclusion:**

Thus, we have studied basics of Python Programming

**13. Experiment/Assignment Evaluation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Experiment/Assignment Evaluation:** | | | | | |
| **Sr. No.** | **Parameters** | | | **Marks obtained** | **Out of** |
| **1** | Technical Understanding (Assessment may be done based on Q & A **or** any other relevant method.) Teacher should mention the other method used - | | |  | 6 |
| **2** | Neatness/presentation | | |  | 2 |
| **3** | Punctuality | | |  | 2 |
| **Date of performance (DOP)** | |  | **Total marks obtained** |  | **10** |
| **Date of checking (DOC)** | |  | **Signature of teacher** | | |

**References**:

[1] James Payne, “Beginning Python using Python 2.6 and Python 3.1”, Wrox Publications.

[2] Dr. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press, Wiley Publications.

**Viva Questions**

1. What is Python?
2. How is Python different from C/C++?
3. How do you declare variables in Python?
4. What is range in Python?
5. What are different math operations and operators used for it in Python?