1. What is software?

Software is a set of instructions for the hardware.

2. What is Python and what are its applications?

Python is an object-oriented programming language that is easy to learn and simple to implement.

Applications of Python

Python is a versatile language that has applications in almost every field

- Artificial intelligence (AI)
- Machine Learning (ML)
- Big Data
- Smart Devices/Internet of Things (IoT)
- Cyber Security
- Game Development
- Backend Development, etc.

3. What are the features of Python?

Features of Python:

- Easy to learn & code
- Open Source Programming Language
- Object-Oriented Language
- Dynamic Typed Language
- Large Standard Library

4. Is Python case-sensitive?

Yes, Python is case-sensitive. The **username**, **UserName**, and **userName** are three different variables, and using these names interchangeably causes an error.

Code

PYTHON

```python
username = "Rahul"
print(username)
print(userName)
```

Output

```
Rahul


NameError: name 'userName' is not defined
```

5. Is Python a dynamically typed programming language?

Yes, Python is a dynamically typed language. This means that in Python the type checking of a variable is done only as code runs, and the type of a variable is allowed to change over its lifetime. There is no need to declare the type of the variable

While programming languages like C, Java, C++, etc are statically typed languages where we cannot change the data type of a variable during the execution of the program.

Code

PYTHON

```python
x = 6
print(type(x))
x = 'Rahul'
print(type(x))
```

Output

```
<class 'int'>
<class 'str'>
```

6. What are the advantages of Python over Java?

Basis of Comparison	Python	Java
Learning curve	Easy to learn	Compared to Python, it's difficult to learn
Typing	Dynamically-typed	Statically-typed
Syntax	Easy to read and remember	Difficult to read and remember
Applications	Artificial Intelligence, Data Science and Machine Learning applications	Enterprise, Embedded and Cross-platform applications
Code Length	Fewer lines of code compared to Java	More lines of code compared to Python

Example Program	print("Hello World")	public class Simple {
		public static void main(String args[]){
		System.out.println("Hello World"); }}

7. How to perform the arithmetic operations using Python?

Addition

The addition is denoted by

`+` sign. It gives the sum of two numbers.

Code

PYTHON

```python
print(2 + 5)
print(1 + 1.5)
```

Output

```
7
2.5
```

Subtraction

The subtraction is denoted by

`-` sign. It gives the difference between the two numbers.

Code

PYTHON

```python
print(5 - 2)
```

Output

```
3
```

Multiplication

The multiplication is denoted by

`*` sign.

Code

PYTHON

```python
print(2 * 5)
print(5 * 0.5)
```

Output

```
10
2.5
```

Division

The division is denoted by

`/` sign.

Code

PYTHON

```python
print(5 / 2)
print(4 / 2)
```

Output

```
2.5
2.0
```

Modulus

To find the remainder between two numbers, we use the Modulus operator

`%`

Code

PYTHON

```python
print(6 % 3)
```

Output

```
0
```

Exponent

To calculate **a power b**, we use Exponent Operator

`**`

Code

PYTHON

```python
print(2 ** 3)
```

Output

```
8
```

8. What is floor division?

To find integral part of quotient we use Floor Division Operator

`//` .

- `a // b`

  Code

PYTHON

```python
print(3 // 2)
```

Output

```
1
```

9. What is Operator Precedence in Python?

The operator precedence determines which operator is executed first if there is more than one operator in an expression.

The operator precedence in Python is listed in the following table. It is in descending order (the upper group has higher precedence than the lower ones).

Operators	Meaning
`()`	Parentheses
`**`	Exponent
`+x` , `-x` , `~x`	Unary plus, Unary minus, Bitwise NOT
`*,` `/,` `//,` `%`	Multiplication, Division, Floor division, Modulus
`+` , `-`	Addition, Subtraction
`<<,` `>>`	Bitwise shift operators
`&`	Bitwise AND
`^`	Bitwise XOR
`|`	Bitwise OR
`==,` `!=,` `>,` `>=,` `<,` `<=,` `is` , `is not` , `in` , `not in`	Comparisons, Identity, Membership operators
`not`	Logical NOT

Operators	Meaning
and	Logical AND
or	Logical OR

BODMAS

The standard order of evaluating an expression

- ***Brackets*** (B)
- ***Orders*** (O)
- ***Division*** (D)
- ***Multiplication*** (M)
- ***Addition*** (A)
- ***Subtraction*** (S)

Expression:

`(5 * 2) + (3 * 4 + 4 / 2)`

Step by Step Explanation

```
(5 * 2) + (3 * 4 + 4 / 2)
(10) + (3 * 4 + 2)
(10) + (12 + 2)
(10) + (14)
24
```

Code

PYTHON

```python
print((5 * 2) + (3 * 4 + 4 / 2))
```

Output

```
24
```

10. What is a Variable?

Variables are like containers for storing values.

Assigning Value to Variable

The following is the syntax for assigning an integer value

`10` to a variable `age`

PYTHON

```python
age = 10
```

Here the equals to

`=` sign is called an **Assignment Operator** as it is used to assign values to variables.

11. What are Data Types?

In programming languages, every value or data has an associated type to it known as data type.
Some commonly used data types

- String
- Integer
- Float
- Boolean

This data type determines how the value or data can be used in the program. For example, mathematical operations can be done on Integer and Float types of data.

12. What are the numeric data types in Python?

The Numeric Data Types in Python are:

- Integers
- Float
- Complex Numbers

Code

PYTHON

```python
a = 10
print("Type of a: ", type(a))

b = 10.0
print("Type of b: ", type(b))

c = 10 + 20j
print("Type of c: ", type(c))
```

Output

```
Type of a:  <class 'int'>
Type of b:  <class 'float'>
Type of c:  <class 'complex'>
```

13. What is meant by mutability? Name some mutable data types?

Mutable means capable of being changed. In Python, objects whose value can be changed are said to be mutable.

Some of the mutable data types in Python are list, dictionary, set and user-defined classes.

14. What is meant by immutability? Name some immutable data types?

Immutable means capable of not being changed. In Python, objects whose value cannot be changed are said to be immutable.

Some of the immutable data types in Python are tuple, integer, boolean, string, etc.

15. What is type conversion or type casting?

Converting the value of one data type to another data type is called **Type Conversion** or **Type Casting**.
We can convert

- String to Integer
- Integer to Float
- Float to String and so on.

String to Integer

`int()` converts valid data of any type into integer

Code

PYTHON

```python
a = "5"
a = int(a)
print(type(a))
print(a)
```

Output

```
<class 'int'>
5
```

Integer to String

`str()` converts data of any type into a string.

Code

```python
a = input()
a = int(a)
b = input()
b = int(b)
result = a + b
print("Sum: " + str(result))
print("Sum: " + str(result))
```

Input

```
2
3
```

Output

```
Sum: 5
```

Similarly,

- `float()` -> Converts to a float data type
- `bool()` -> Converts to a boolean data type

16. What is a String?

A String is a stream of characters enclosed within quotes.
Stream of Characters

- Capital Letters (A – Z)
- Small Letters (a – z)
- Digits (0 – 9)
- Special Characters (~ ! @ # $ % ^ . ?,)
- Space

Some examples:

- `"Hello, World!"`
- `"some@example.com"`
- `"1234"`

17. What is String Slicing?

Obtaining a part of a string is called String Slicing.

Syntax:

`variable_name[start_index:end_index]`

- `end_index` is not included in the slice.

Code

PYTHON

```python
message = "Hi Ravi"
part = message[3:7]
print(part)
```

Output

```
Ravi
```

Slicing to End

If the end index is not specified, slicing stops at the end of the string.

Code

PYTHON

```python
message = "Hi Ravi"
part = message[3:]
print(part)
```

Output

```
Ravi
```

Slicing from Start

If the start index is not specified, slicing starts from the index 0.

Code

PYTHON

```python
message = "Hi Ravi"
part = message[:2]
print(part)
```

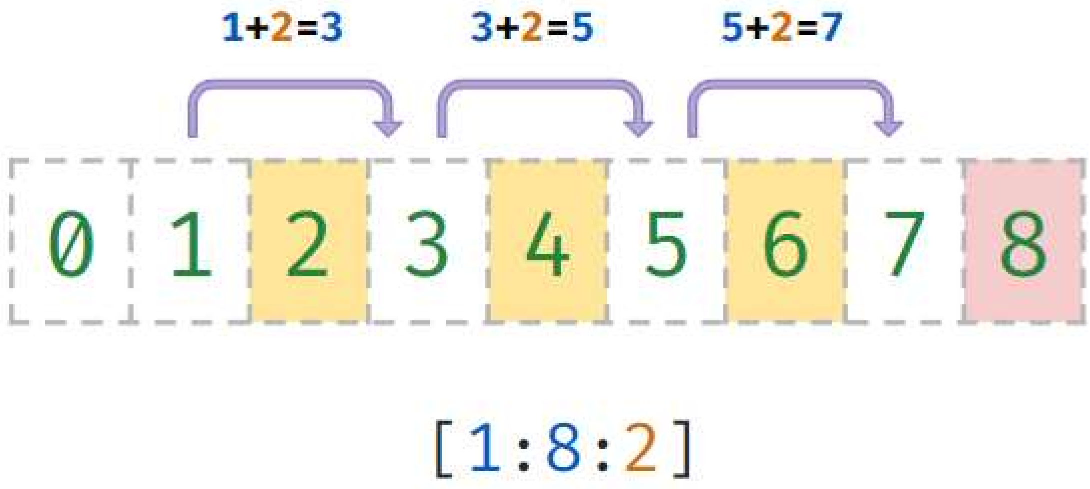
Output

```
Hi
```

Extended Slicing

Syntax:

`variable[start_index:end_index:step]`



Step determines the increment between each index for slicing.

Code

PYTHON

```python
a = "Waterfall"
part = a[1:8:2]
print(part)
```

Output

```
aefl
```

18. How to reverse a string?

A string can be reversed using extended slicing.

Syntax:

`variable[start:end:negative_step]`

`-1` for step will reverse the order of the characters.

Code

PYTHON

```python
string_1 = "Program"
string_2 = string_1[::-1]
print(string_2)
```

Output

```
margorP
```

19. What is string `capitalize()` in Python?

The

`capitalize()` method converts the first character of a string to an uppercase letter and all other alphabets to lowercase.

Code

PYTHON

```python
sentence = "proGraMmiNg"
capitalized_string = sentence.capitalize()
print(capitalized_string)
```

Output

```
Programming
```

20. What is string `replace()` in Python?

The

`replace()` returns a new string after replacing all the occurrences of the old substring with the new substring.

Syntax:

`str_var.replace(old, new)`

Code

PYTHON

```python
sentence = "teh cat and teh dog"
sentence = sentence.replace("teh", "the")
print(sentence)
```

Output

```
the cat and the dog
```

21. What is `round()` function?

Rounds the float value to the given number of decimal digits.

Syntax:

`round(number, digits(optional))`

`digits` -> defines the number of decimal digits to be considered for rounding.

- When `digits` not specified, the default value is **0**.

Code

PYTHON

```python
a = round(3.14159, 2)
print(a)
a = round(5.6777)
print(a)
```

Output

```
3.14
6
```

22. How to write comments in Python?

A comment starts with a hash

`#`

It can be written in its own line next to a statement of code.

Code

PYTHON

```python
n = 5
# Finding if Even
even = (n % 2 == 0)
print(even) # prints boolean value
```

Output

```
False
```

○ MARK AS COMPLETED

Submit Feedback

Notes Discussions