**Python:**

Python is a **high-level**, **interpreted**, and **general-purpose** programming language created by **Guido van Rossum** in **1991**. It emphasizes code readability with its clean syntax and indentation.

### ****Why Use Python?****

* Easy to learn and use.
* Extensive libraries for various applications.
* Supports multiple programming paradigms (object-oriented, functional, etc.).

### ****Where is Python Used?****

* Web development (e.g., Django, Flask)
* Data science and machine learning
* Automation and scripting
* Game development
* Scientific computing
* Artificial intelligence(AI) and Deep learning
* Internet of Things (IoT)
* Web scraping
* Cyber security

### ****Disadvantages of Python:****

* Slower compared to compiled languages like C or C++.
* High memory consumption.
* Not ideal for mobile development.

**Companies using Python:**

Google,Netlfix,instagram,spotify,dropbox,Nasa,uber,reddit,ibm and **10kcoders dashboard.**

**Arithmetic Operators in Python**

+,-,\*,/,//,%,\*\*

Print(2+4) - addition operater

Print(2-4) - subtraction operater

Print(2\*4) - multiplication operater

Print(2/4) - divison operater

Print(2%4) - module operater

Print(2//4) - floor divison

Print(2\*\*4) - exponentiation operater

**Without variables:**

Without variables we can write like this..

**Result=(12345+6789)**

Print(result) //prints results

**With variables:**

With variables we can write like this..

**a=123456**

**b=858968**

Print(a+b)

**Strings:(“ ”,’ ’)**

**Str1=”hello”**

**Str2=”hello world”**

Print(str1)

Print(str2)

Print(str1+str2) // string addition

**How to find length of a string: use len()**

**Str1=”hello”**

**Str2=”hello world”**

Print(len(str1))

Print(len(str2)) //prints the length of the string

**String indexing:**

**Str1=”hello”**

**Str2=”hello world”**

Print(str1[1]) //indexing starts from 0

Print(str2[5]) //space is also index

Print(str2[6])

**String negative indexing:**

**Str1=”hello”** //negative starts from -1

Print(str1[-1]) //it prints last letter of an string

Print(str1[-5]) // it prints first letter of an string

**Slicing:**

**Str1=”hello”**

**Str2=”hello world”**

Print(str1[1:3]) //prints “el”

Print(str2[5:7]) //space and “w” space also indexing in slicing

**Slicing with high (stop) value index:**

**Str1=”hello”**

**Str2=”hello world”**

Print(str2[4:100])

//prints from 4 index to complete string because it will accepts high indexing value also.the string beyond the availabe length doesn’t cause an error python simply return the string from index 4 to the end of the string.

**Slicing with space:**

**Str1=”hello”**

**Str2=”hello world”**

Print(str2[ 4:space] / /prints “o world”

Prints charcter from the start index4 upto the end of the string.

Print(str2[space:4]) //prints “hell”

Prints charcter from the start index 0 upto but not including index 4.

**Str1=”hello”**

**Str2=”hello world”**

Multiplication of two strings ---- shows an error

Addition of two strings ------ shows output

Multiplication of two strings ---- shows an error

Multiplication strings with integer ---- shows output

Subtraction of two strings ---- shows an error

String + integer ---- shows only concatenate strn(not an integer)to str

**Skip in python:**

**Str1=”hello”**

**Str2=”hello world”**

Print(str2[1:8]) //prints ello wor

Print(str2[1:8:2]) //prints elwr ----picks every second character.

**Skip** typically refer to the step parameter in slicing which controls how many elements to skip between selections.

**Numeric data type in python:**

**Num1=2.3**

**Num2=4.3**

Print(num1+num2)  **// addition operater**

Output:6.6

Print(num1-num2) **// subtraction operater**

Output:-2.0

Print(num1/num2) **// division operater**

Output:0.534883729……

Print(num1%num2) **// module operater**

Output:2.3

Print(num1\*num2) **// multiplication operater**

Output:9.889999…..

Print(num1//num2) **// floor division operater**

Output:0.0

Print(num1\*\*num2) **// exponentaiation operater**

Output:35.9277230..

**Integers data type in python:**

**Num1=2**

**Num2=4**

Integers supports all common arithematic operaters

+,-,\*,/,//,%,\*\*

**Complex numbers:**

Complex numbers is a number that has two parts they are..

**One is real part**

**Second is imaginary part**

**cmp1=5+6j (5 is a real part) (6j is a imaginery part)**

**cmp2 =7+oj (7 is a real part) (oj is a imaginery part)**

**Complex numbers with arthematic operaters**

Print(cmp1+cmp2)  **// addition operater with complex numbers**

Output: 12+6j

Print(cmp1-cmp2) **// subtraction operaterwith complex numbers**

Output: -2+6j

Print(cmp1\*cmp2) **// multiplication operater with complex numbers**

Output: 35+42j

Print(cmp1/cmp2) **// division operater with complex numbers**

Output: 0.7142857142857143 + 0.8571428571428571j

Print(cmp1%cmp2) **// module operater with complex numbers**

Output: not supports show an error

Print(cmp1//cmp2) **// floor division operater with complex numbers**

Output: not supports show an error

Print(cmp1\*\*cmp2) **// exponentaiation operater with complex numbers**

Output: -11+60j

**Boolean in python:**

Boolean is a data type that can have only two possible values

They are capital **(True / Flase)**

True and False are treated as special constants.

T and F are case-sensitive

Here **True=1 and false=0**

**Bool 1= True**

**Bool 2= False**

**Type() in python:**

The type() function is used to determine the **data type** of a given object or value.

**Name=”hello”**

Print(type(name)) // prints class ‘str’

**Name1=[1,2,3,4]**

Print(type(name1)) // prints class ‘list’

**Name2=True**

Print(type(name2)) // prints class ‘bool’

**Number1=123**

Print(type(number1)) // prints class ‘int’

**Numbe2r=2.5**

Print(type(number2)) // prints class ‘float’

**Name3=(1,3.4,5,6)**

Print(type(name3)) // printd class ‘tuple’

**cmp=3+4j**

Print(type(cmp)) // prints class‘complex’