**PYTHON FUNDAMENTALS**

**2 mark Answesr**

**1.What is EOL?**

An EOL ( **End of Line** ) error indicates that the Python interpreter expected a particular character or set of characters to have occurred in a specific line of code, but that those characters were not found before the end of the line . This results in Python stopping the program execution and throwing a syntax error .

**2. What is an escape sequence?**

* An escape sequence is a special character used in the form of backslash(\) followed by a character that is required.
* These characters are used to represent whitespace.
* Whitespace gives characters like space, tab, formfeed, vertical tab.

**3.What is the maximum line length in a python program?**

A maximum line length of 79 characters is recommended by PEP 8, and I usually don't find it too hard to follow this recommendation.Also in accordance with PEP 8, I try to rely on the implied line continuation inside parentheses. This is because it allows you to have multiple files open next to one another, while also avoiding line wrapping. Of course, keeping statements to 79 characters or less is not always possible.

**4**.**Write any four keywords of python language?**

* Value Keywords: True, False, None.
* Operator Keywords: and, or, not, in, is.
* Control Flow Keywords: if, elif, else.
* Iteration Keywords: for, while, break, continue, else.
* Structure Keywords: def, class, with, as, pass, lambda.
* Returning Keywords: return, yield.
* Import Keywords: import, from, a.

**5.What are the types of Assignment statements? Explain**

Assignment statements enable the programmer to define or redefine a symbol by assigning it a value. This value may be a reference to another symbol, register name, or expression. The new value takes effect immediately and remains in effect until the symbol is redefined

There are two types of assignment statements:

* Symbol assignment statements, which define or redefine a symbol in the symbol name space.
* Register assignment statements, which define or redefine a register name in the symbol name space

**6. Explain with a diagram how a variable refers to a memory location?**

**A variable name of memory cell it is variable because the value in the cell can change .each memory cell has an address python and other higher language use a symbol table to map a variable name to the address it represent**

Programme

X=2

Y=x+6

Memory cell

Address Value

0 2

4 8

8

Symbol table

X 0

Y 4

The programme puts data in contigues memory so the variable x is at 0 adress Y at address 4

The symbol table maps variable name to address so x is 0 to the computer and y is address 4

**7. What is Dynamic typing?**

Python is a dynamically typed language. This means that the Python interpreter does type checking only as code runs, and the type of a variable is allowed to change over its lifetime.

**8. Write any four python naming conventions**

**1. General**

* Avoid using names that are too general or too wordy. Strike a good balance between the two.
* Bad: data\_structure, my\_list, info\_map, dictionary\_for\_the\_purpose\_of\_storing\_data\_representing\_word\_definitions
* Good: user\_profile, menu\_options, word\_definitions
* Don’t be a jackass and name things “O”, “l”, or “I”
* When using CamelCase names, capitalize all letters of an abbreviation (e.g. HTTPServer)

**2. Packages**

* Package names should be all lower case
* When multiple words are needed, an underscore should separate them
* It is usually preferable to stick to 1 word names

**3. Modules**

* Module names should be all lower case
* When multiple words are needed, an underscore should separate them
* It is usually preferable to stick to 1 word names

**4. Classes**

* Class names should follow the UpperCaseCamelCase convention
* Python’s built-in classes, however are typically lowercase words
* Exception classes should end in “Error”

**5. Global (module-level) Variables**

* Global variables should be all lowercase
* Words in a global variable name should be separated by an underscore

**9. What is input ( ) function? Write down the general format of input ( ) function and explain with proper example**.

Python input() function is used to get input from the user. It prompts for the user input and reads a line. After reading data, it converts it into a string and returns that. It throws an error EOFError if EOF is read.

Signature

1. input ([prompt])

Parameters

**prompt**: It is a string message which prompts for the user input.

Return

It returns user input after converting into a string.

Let's see some examples of input() function to understand it's functionality.

Python input() Function

Here, we are using this function get user input and display to the

1. # Python input() function example
2. # Calling function
3. val = input("Enter a value: ")
4. # Displaying result
5. print("You entered:",val)

**10. What is print ( ) function? Write down the general format of print ( ) function and explain with proper example.**

The print() function prints the specified message to the screen, or other standard output device. message can be a string, or any other object, the object will be converted into a string before written to the screen.

## Syntax

print*(object(s)*, sep=*separator*, end=*end*, file=*file*, flush=*flush*)

Example :

input ( ) with prompt string

>>> city = input (“Enter Your City: ”)

Enter Your City: Madurai

>>> print (“I am from “ , city)

I am from Madurai

**PYTHON FUNDAMENTALS**

**1 mark Answer**

**1.What is character set?**

A character set defines the valid characters that can be used in source programs or interpreted when a program is running. The source character set is the set of characters available for the source text. The execution character set is the set of characters available when executing a program. The source character set does not necessarily match the execution character set; for example, when the execution character set is not available on the devices used to produce the source code

**2. What is token?**

A token is the smallest individual unit in a python program. All statements and instructions in a program are built with tokens.

**3. List the types of**

1 Identifiers.

2 Keyword

3 Constants.

4 Operators.

5 Special Characters.

6 Strings.

**4. What is keyword?**

These keywords have a special meaning and they are used for special purposes in Python programming language

**5. What is an identifier? Give suitable example**.

A Python identifier is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore (\_) followed by zero or more letters, underscores and digits (0 to 9).

Example:

myClass , var\_1 and print\_this\_to\_screen

**6.What is a literal ?**

literals in Python is defined as the raw data assigned to variables or constants while programming

**7. What is string?**

A string is a sequence of characters.A character is simply a symbol

**8. What is single line string?**

**Whenever Python interpreter starts processing a string it looks for a quotation mark**, it could either be a single quote or double quote. The opening quote indicates the starting of the string and closing quote indicates the end of the string.

**9. What is multi line string?**

A multiline string in Python **begins and ends with either three single quotes or three double quotes**. Any quotes, tabs, or newlines in between the “triple quotes” are considered part of the string. Python's indentation rules for blocks do not apply to lines inside a multiline string.

**10. What is EOL?**

 EOL while scanning string literal "EOL" stands for "**end of line**". An EOL error means that Python hit the end. of a line while going through a string. This can be because you forgot ending quotes , or because you tried. to make a string extend past one line. Strings enclosed in single or double quotes.

**11. What is an escape sequence?**

A sequence is just a set of two or more characters and an escape where the sequence begins with a backslash (\\) and other characters in the set follow that backslash.

**12. What is Boolean literal?**

Python literal **Boolean has two values.** **One is True, and another one is False**. In any programming language, Booleans have only two values. In the same vein, Python language also exhibits the same properties.

**13. What is none?**

The None is **used to define a null value, or no value at all**. None is not the same as 0, False, or an empty string. None is a data type of its own (NoneType) and only None can be None

**14. What is an operator?**

Operators are **special symbols in Python that carry out arithmetic or logical computation**. The value that the operator operates on is called the operand. Here, + is the operator that performs addition. 2 and 3 are the operands and 5 is the output of the operation.

**15. What is Unary Operator?**

 A unary operator is **an operator which works on a single operand**. Python support unary minus operator(-). When an operand is preceded by a minus sign, then the unary operator negates its value.

**16. What is Binary Operator?**

Python bitwise operators work on integers only and the final output is returned in the decimal format. Python bitwise operators are also called binary operators. **The integers are converted into binary format and then operations are performed bit by bit**, hence the name bitwise operators.

**17. List the shift operators?**

1 Right shift operator

2 left shift operator

**18. List the Bitwise operators?**

AND, OR, XOR, NOT, SHIFT, and MASK

**19. What is an assignment statement?**

Assignment statement. A statement that assigns a value to a name (variable).

1. To the left of the assignment operator, =, is a name.

2.To the right of the assignment operator is an expression which is evaluated by the Python interpreter and then assigned to the name

**20 What is Punctuators?**

Punctuators are nothing but symbols used in programming languages to organize sentence structures, and indicate the rhythm and emphasis of expressions, statements and program structure.

**Examples of punctuators in python**: ‘, “, #, \, /, (, ), [, ], {, }, @, ;, :, . Etc.

**21 What is comment ?**

Comments in Python are the lines in the code that are ignored by the interpreter during the execution of the program. Comments enhance the readability of the code and help the programmers to understand the code very carefully.

**22. What is whitespace?**

Whitespace is simply a character that is used for spacing and has an “empty” representation

**23. What is statement?**

A statement is an instruction that a Python interpreter can execute. So, in simple words, we can say anything written in Python is a statement. Python statement ends with the token NEWLINE character. It means each line in a Python script is a statement.

**24. Weather python uses statement termination? Justify your answer**

python does not use any symbol to terminate the statement.

**25. What is the maximum line length in a python program?**

79 characters is maximum line length in a python

**26. What is Block?**

 A block is a piece of Python program text that is executed as a unit.

The following are blocks: a module, a function body, and a class definition. Each command typed interactively is a block.

**27. What is Code Block?**

A code block is a piece of Python program text that can be executed as a unit, such as a module, a class definition or a function body. Some code blocks (like modules) are normally executed only once, others (like function bodies) may be executed many times. Code blocks may textually contain other code blocks.

**28. What is Code?**

The code module provides facilities to implement read-eval-print loops in Python. Two classes and convenience functions are included which can be used to build applications which provide an interactive interpreter prompt.

**29. what do you mean by case sensitive language**

It's the differentiation between lower- and uppercase letters. It can be a feature not only of a programming language but of any computer program

**30. What is variable?**

Variables are used to store data, they take memory space based on the type of value we assigning to them. Creating variables in Python is simple, you just have write the variable name on the left side of = and the value on the right side

**31. What is Lvalue?**

An lvalue refers to an object that persists beyond a single expression.

**32. What is Rvalue?**

An rvalue is a persist temporary value that does not beyond the expression that uses it.

**33. What is an Assignment state?**

An assignment statement evaluates the expression list (remember that this can be a single expression or a comma-separated list, the latter yielding a tuple) and assigns the single resulting object to each of the target lists, from left to right.

**34. What is Dynamic typing?**

Dynamic typing in Python allows it to store the variable in separate memory locations without declaring or knowing the type of variable unless it reaches the run-time.

**PYTHON FUNDAMENTALS**

**3 mark Answes**

**1.What are the python naming conventions**

**Python naming conventions for variable** names are same as function names.naming convention is a convention (generally agreed scheme) for naming things. Conventions differ in their intents, which may include to: Allow useful information to be deduced from the names based on regularities.

**Naming Conventions rules for Variables, Packages, Modules and Functions (Methods) are:**

* **Rule-1**: You should start variable name with an alphabet or **underscore(\_)** character.
* **Rule-2:** A variable name can only contain **A-Z,a-z,0-9** and **underscore(\_)**.
* **Rule-3:** You cannot start the variable name with a **number**.
* **Rule-4:** You cannot use special characters with the variable name such as such as**$,%,#,&,@.-,^**etc.
* **Rule-5**: Variable names are **case sensitive**. For example str and Str are two different variables.
* **Rule-6:** Do not use reserve keyword as a variable name for example keywords like **class, for, def, del, is, else,** **try, from,**etc

**2.Explain the representation of string in python language?**

A string is a series of characters. In Python, anything inside quotes is a string. And you can use either single or double quotes.

For example: message = 'This is a string in Python' message = "This is also a string"

In Python, a string is a series of characters. Python supports two types of strings — Single-line strings and Multi-line strings. Single line strings are enclosed in single or double quotes and terminate in one line. Multi-line strings store multiple lines of text and are enclosed in triple quotes.

Message='This is a string in Python'

Message="It's a string"

* Also, Python strings are immutable.
* Use quotes, either single-quotes or double-quotes to create string literals.
* Use the backslash character \ to escape quotes in strings
* Use raw strings r'...' to escape the backslash character.
* Use f-strings to insert substitute variables in literal strings.
* Place literal strings next to each other to concatenate them. And use the + operator to concatenate string variables.
* Use the len() function to get the size of a string.
* Use the str[n] to access the character at the position n of the string str.
* Use slicing to extract a substring from a string.

**3.Explain the types of strings supported by python language.**

Python supports two types of strings.

They are:

Single Line Strings: Strings are terminated in a single line.

Multiple Strings: Strings that stores multiple lines of text.

Python supports two types of strings — Single-line strings and Multi-line strings.

Single line strings are enclosed in single or double quotes and terminate in one line.

Multi-line strings store multiple lines of text and are enclosed in triple quotes.

Message='This is a string in Python'

Message="It's a string"

**4.Explain escape sequences.**

***The sequence of characters after a backslash is known as an escape sequence***

The sequence of character which has indirect meaning when it placed within double quotes.

Example

#escape sequence example

print("Happy New Year")

# Useful Escape Sequences

| Escape sequence | Description | Example | Output |
| --- | --- | --- | --- |
| \n | New line | print("Hello\nWorld") | Hello World |
| \t | Horizontal tab | print("Hello\tWorld") | Hello    World |
| \' | Single quote | print("Hello \'World\' ") | Hello 'World' |
| \" | Double quote | print("Hello \"World\" ") | Hello "World" |
| \\ | Backslash | print("Hello \\World") | Hello \World |

**5.Explain numerical literals supported in python language.**

A numeric literal is a character string selected from the digits, the plus sign, the minus sign, and the decimal point.

The following rules govern the formation of numeric literals:

* A literal must contain at least one digit
* A literal must contain no more than one sign character and, if one is used, it must be the leftmost character of the string
* A literal must not contain more than one decimal point. The decimal point is treated as an assumed decimal point and may appear anywhere within the literal except as the rightmost characterIf a literal conforms to the rules for formation of a numeric literal, but is enclosed in quotation marks, it is a nonnumeric literal.

Numeric literals may also be specified using binary, octal, or hexadecimal notation. To specify a numeric literal in one of these forms, preface the number with one of the following

For example:

|  |  |
| --- | --- |
| Binary | "B#" |
| Octal | "O#" |
| Hexadecimal | “X#” or “H#” |

| **Number** | **Binary** | **Octal** | **Hexadecimal** |
| --- | --- | --- | --- |
| 3 | B#11 | O#3 | X#3 or H#3 |
| 8 | B#1000 | O#10 | X#8 or H#8 |
| 12 | B#1100 | O#14 | X#C or H#C |
| 128 | B#10000000 | O#200 | X#80 or H#80 |
| 255 | B#11111111 | O#377 | X#FF or H#FF |

**6.Explain the Floating point literals supported in python language**

*Floating-point literals* are numbers that have a decimal point or an exponential part. They can be represented as:

Real literals

1.[Binary floating-point literals](https://www.ibm.com/docs/en/zos/2.4.0?topic=literals-floating-point" \l "lit_fltpt__fp_decimal)

2.C only[Hexadecimal floating-point literals (C only)](https://www.ibm.com/docs/en/zos/2.4.0?topic=literals-floating-point#lit_fltpt__hex_float_constants)

3. [Complex literals](https://www.ibm.com/docs/en/zos/2.4.0?topic=literals-floating-point#lit_fltpt__complex_lit)

1 . Binary floating-point literals

A real binary floating-point constant consists of the following:

* An integral part
* A decimal point
* A fractional part
* An exponent part
* An optional suffix

Both the integral and fractional parts are made up of decimal digits. You can omit either the integral part or the fractional part, but not both. You can omit either the decimal point or the exponent part, but not both.

2. Hexadecimal floating-point literals (C only)

Real hexadecimal floating-point constants, which are a C99 feature, consist of the following parts.

* a hexadecimal prefix
* a significant part
* a binary exponent part
* an optional suffix

**The significant part represents a rational number and is composed of the following**:

* a sequence of hexadecimal digits (whole-number part)
* an optional fraction part
* The optional fraction part is a period followed by a sequence of hexadecimal digits.
* The exponent part indicates the power of 2 to which the significant part is raised, and is an optionally signed decimal integer. The type suffix is optional

## 3.Complex literals

Complex literals, which were introduced in the C99 standard, are constructed in two parts: the real part, and the imaginary part.

**7.Explain the General structure of python program and give example?**

The basic Python data structures in Python include list, set, tuples, and dictionary. Each of the data structures is unique in its own way. Data structures are “containers” that organize and group data according to typpe

* Lists, sets, and tuples are the basic data structures in the Python programming language.
* One of the differing points among the data structures is mutability, which is the ability to change an object after its creation.
* Lists and tuples are the most useful data types, and they can be found in virtually every Python program.

**8.What is whitespace how its useful in python programming?**

Whitespace is simply **a character that is used for spacing and has an “empty” representation**. It refers to tabs and spaces in the context of Python (it also includes exotic Unicode spaces). The Python String isspace() method is used to determine whether an argument has all whitespace characters such as: ' ' – Space.

Python uses white space for two things: newlines terminate logical lines, and changes in indentation delimit blocks. Both of these behaviors are somewhat contextual.

Python distinguishes physical lines from logical lines. A physical line is an actual line of text: a string of characters, terminated by a newline. A logical line is a syntactic entity: a statement or part of a definition. Multiple physical lines may be joined together into a logical line if the newlines separating them are escaped with backslashes or if the newlines occur within a nest of parentheses or bracket

**9.What is input ( ) function? Write down the general format of input ( ) function and explain with proper example**

The input() function **allows a user to insert a value into a program**. input() returns a string value. You can convert the contents of an input using any data type

Example: Take User Input

 Copy

>>> user\_input = input()

How are you?

>>> user\_input

'How are you?'

Above, the input() function takes the user's input in the next single line, so whatever user writes in a signle line would be assign to to a variable user\_input. So, the value of a user\_input would be whatever user has typed.

The following example demonstrets how to use the optional prompt parameter.

Example: Input with Prompt

 Copy

>>> name = input('Enter Your Name: ')

Enter Your Name: Steve

>>> name

'Steve'

In the above example, the input('Enter Your Name: ') function with prompt string 'Enter Your Name: '. So, in the next line it will display a prompt first, asking user what to do. User can then enter the name after the prompt string in the same line and that would be assign to the name variable.

The input() function converts all the user input to a string even if that's the number.

Example: Input with Prompt

 Copy

>>> age = input('Enter Your Age: ')

Enter Your Name: 25

>>> age

'25'

>>> type(age)

<class 'str'>

User can enter Unicode characters as well, as shown below.

Example: input() with Unicode Chars

 Copy

>>> uc = input("Enter Unicode Char: ")

Enter Unicode Char: åê

>>> uc

åê

**10.What is output()function?write down the general format of output() function and explain with proper example**

**The print() function prints the specified message**

**the screen, or other standard output device**. The message can be a string, or any other object, the object will be converted into a string before written to the screen

**print (“hello world”)**