**1.What is EOL?**

\* EOL stands for “end of life,” which occurs to hardware and software. It is the stage of a product in which it becomes outdated or unsupported by the manufacturer.

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

\*To insert characters that are illegal in a string, use an escape character. An escape character is **a backslash \ followed by the character you want to insert**.

|  |  |
| --- | --- |
| **Escape sequence** | **Meaning** |
| \’ | This represents a single quote |
| \n | This represents a newline |
| \r | This represents a carriage return |
| \t | This represents a tab |
| \b | This represents a backspace |
| \f | This represents a formfeed |
| \ooo | This represents an octal value |
| \xhh | This represents a hex value |
| \\ | This represents a backslash |
| \uxxxx | This represents 16 bits hex value |
| \uxxxxxxxx | This represents 32 bits hex value |
| **3.What is the maximum line length in a python program?**  \*The Python standard library is conservative and requires limiting lines to **79 characters** (and docstrings/comments to 72). The preferred way of wrapping long lines is by using Python's implied line continuation inside parentheses, brackets and braces. |  |

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

\*There are 33 keywords in Python 3.7.

and a few of the keywords generally used in the program coding are break,continue, true, false, and, or, not, for, while, def, class, if, else, elif, import, from, except, exec, print, return, yield, lambda, global, etc.

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

\*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 is the name of a memory cell. It is "variable" because the value in the cell can change. Each memory cell has an address. Python and other high-level languages use a symbol table to map a variable name to the address it represents.

*Program*  
x = 2  
y = x + 6

*Main Memory*

|  |  |
| --- | --- |
| **Address** | **Value** |
| 0 | 2 |
| 4 | 8 |
| 8 |  |

*Symbol Table*

|  |  |
| --- | --- |
| x | 0 |
| y | 4 |
|  |  |

The program puts data in *contiguous* memory. So the variable 'x' is at address 0, 'y' at address 4.

The symbol table maps variable names to addresses. So x is 0 to the computer, and y is the 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**

* Single lowercase
  + Often used for local variables in functions, such as x or i
* Single uppercase
  + Often used to name matrices
* Lowercase word
  + The most frequent naming convention for the widest variety of objects
* Lowercase word with underscores
  + Same uses as Lowercase word, but for more complicated names
* Uppercase word
  + Used for single-word static variables
* Uppercase word with underscores
  + Used for multi-word static variables
* Capitalized word
  + This is where each word is capitalized, and there are no spaces or underscores between them
  + Used for naming classes (even if the name is just a single, capitalized word)
* Mixed Case
  + This is where you start with a lowercase word followed by every other word capitalized
  + This convention is predominantly used in Java and less in Python

Each of these options has different use cases. They are not only for aesthetics, but each option also helps you create meaningful objects in your code.

**Names to Avoid**

Just as there are things you should do, there are things you shouldn’t. These are conventions you will want to avoid when writing Python code.

* Lowercase letter “el” – instead, use a capital letter “L”
* Uppercase letter “oh” – “O”
* Uppercase letter “eye” – “I”

We avoid those names because in some fonts, they are indistinguishable from the numbers “1” and “0”, which makes it difficult for the user to understand what’s going on.

You should also avoid using Python keywords and built-in class/function names for your variable names. For example, words like “max,” “sum,” “class,” and “list” are words that already exist in any Python environment. Therefore, to use them for something other than their function could make things messy and confusing.

**Modules and Packages**

A module is a collection of pre-built functions and other objects used to perform certain tasks. Modules should have short, lowercase names. Underscores can be used in the module name if it improves readability (like for names with multiple words).

Packages are like directories that contain modules and other objects. Python packages should follow the same convention, although the use of underscores in package names is discouraged.

**Variables and Functions**

Do you know the difference between a function and a method?To understand the difference, think of the square-rectangle relationship: a method is a function, but not all functions are methods. A method is a special type of function that belongs to a class. Only objects of that class type can use these special functions. This is part of the object-oriented paradigm of Python and can help improve the organization and readability of our code. In Python, the names of variables and functions should be lowercase. Individual words can be separated by underscores when needed. This will improve readability within your code. Method names should follow the same conventions as function names.

Here are a few best practices to follow when naming your variables and functions:

* Constants should be represented by all capital letters and separated by underscores when needed
* Use names that are representative of the meaning of the object rather than meaningless, single-character names
* Names i, j, and k should be reserved for representing index values

**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.

Example

>>> 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.

**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. 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")

**PYTHON FUNDAMENTALS**

**One Mark Questions**

1. **What is character set?**

\*A character set refers to the composite number of different characters that are being used and supported by a computer software and hardware. It consists of codes, bit pattern or natural numbers used in defining some particular character.

A character set may also be referred to as character map, charset or character code.

1. **What is token?**

a token is **an object that represents something else, such as another object (either physical or virtual), or an abstract concept** as, for example, a gift is sometimes referred to as a token of the giver's esteem for the recipient. In computers, there are a number of types of tokens.

1. **List the types of tokens**

In the C language, the following 6 types of tokens are available:

* Identifiers.
* Keywords.
* Constants.
* Operators.
* Special Characters.
* Strings.

1. **What is keyword?**

\* keywords have a special meaning and they are used for special purposes in Python programming language. For example – Python keyword “while” is used for while loop thus you can't name a variable with the name “while” else it may cause compilation error. There are total 33 keywords in Python 3.6.

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

An identifier is a name that identifies (that is, labels the identity of) either a unique object or a unique class of objects, where the "object" or class may be an idea, physical countable object (or class thereof), or physical noncountable substance (or class thereof).

Example: int amount; double totalbalance; In the above example, amount and totalbalance are identifiers and int, and double are keywords.

1. **What is a literal?**

\*Literals in Python is defined as the raw data assigned to variables or constantswhile programming. We mainly have five types of literals which includes string literals, numeric literals, boolean literals, literal collections and a special literal None.

1. **What is string?**

\*A string literal is where you specify the contents of a string in a program. >>> a = 'A string' Here 'A string' is a string literal. The variable a is a string variable, or, better put in Python, a variable that points to a string. String literals can use single or double quote delimiters.

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

\*Single line comments are those comments which are written without giving a line break or newline in python. A [python comment](https://www.pythonforbeginners.com/comments/comments-in-python)is written by initializing the text of comment with a # and terminates when the end of line is encountered. The following example shows a single line comment in a program where a function is defined to add a number and its square to a [python dictionary](https://www.pythonforbeginners.com/dictionary/how-to-use-dictionaries-in-python/) as key value pair.

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

\*As the name specifies, a multi line comment expands up to multiple lines. But python does not have syntax for multi line comments. We can implement multi line comments in python using single line comments or triple quoted python strings.

**10.What is EOL?**

\*EOL stands for “end of life,” which occurs to hardware and software. It is the stage of a product in which it becomes outdated or unsupported by the manufacturer.

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

\*Escape sequences are typically used to specify actions such as carriage returns and tab movements on terminals and printers. They are also used to provide literal representations of nonprinting characters and characters that usually have special meanings, such as the double quotation mark (").

**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. Any non-zero value is True, and the value zero is False

**13. What is none?**

\*The None keyword 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 logicalcomputation. The value that the operator operates on is called the operand. For example: >>> 2+3 5. 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. For example, if a number is positive, it becomes negative when the number is preceded by the unary operator.

**16. What is Binary Operator?**

\*In Python, bitwise operators are **used to performing 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.

**17. List the shift operators**

\*The bitwise shift operators are the right-shift operator ( >> ), which moves the bits of an integer or enumeration type expression to the right, and the left-shift operator ( << ), which moves the bits to the left.

List the Bitwise operators

|  |  |
| --- | --- |
| Operators | Meaning of operators |
| & | [Bitwise AND](https://www.programiz.com/c-programming/bitwise-operators#and) |
| | | [Bitwise OR](https://www.programiz.com/c-programming/bitwise-operators#or) |
| ^ | [Bitwise XOR](https://www.programiz.com/c-programming/bitwise-operators#xor) |
| ~ | [Bitwise complement](https://www.programiz.com/c-programming/bitwise-operators#complement) |
| << | [Shift left](https://www.programiz.com/c-programming/bitwise-operators#left-shift) |
| >> | [Shift right](https://www.programiz.com/c-programming/bitwise-operators#right-shift) |

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

\*An assignment statement **sets the current value of a variable, field, parameter, or element**. The statement consists of an assignment target followed by the assignment operator and an expression. When the statement is executed, the expression is evaluated and the resulting value is stored in the target.

**19. What is Punctuators?**

\*A punctuator is a token that has syntactic and semantic meaning to thecompiler, but the exact significance depends on the context. A punctuator can also be a token that is used in the syntax of the preprocessor. C99 and C++ define the following tokens as punctuators, operators, or preprocessing tokens

**20. 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. There are three types of comments in Python – Single line Comments.

**21. What is whitespace?**

\*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.

**22. 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. For example, a = 10 is an assignment statement.

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

\*Python uses the ; as a separator, not a terminator. You can also use them at the end of a line, which makes them look like a statement terminator, but this is legal only because blank statements are legal in Python -- a line that contains a semicolon at the end is two statements, the second one blank.

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

\*PEP 8 suggests lines should be limited to 79 characters. 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.

**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. Statement: Instructions that a Python interpreter can execute are called statements.

**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?**

\*The language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters. The while keyword, for example, must be typed “while”, not “While” or “WHILE”.

**30. What is variable?**

\*A Python variable is a symbolic name that is a reference or pointer to an object. Once an object is assigned to a variable, you can refer to the object by that name. But the data itself is still contained within the object. For example: >>> >>> n = 300.

**31. What is Lvalue?**

\*The Lvalue (pronounced: L value) concept refers to the requirement that the operand on the left side of the assignment operator is modifiable, usually a variable.

**32. What is Rvalue?**

\*Rvalue references is a small technical extension to the C++ language. Rvalue references allow programmers to avoid logically unnecessary copying and to provide perfect forwarding functions. They are primarily meant to aid in the design of higer performance and more robust libraries

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

\*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 means that the type of the variable is determined only duringruntime. Due to strong typing, types need to be compatible with respect to the operand when performing operations. For example Python allows one to add an integer and a floating point number, but adding an integer to a string produces error.

` **PYTHON FUNDAMENTALS**

**Two Marks Questions**

1. **Explain the character set of python**

\*A character set is a set of valid characters acceptable by a programming language in scripting. In this case, we are talking about the Python programming language. So, the Python character set is a valid set of characters recognized by the Python language.

1. **What are the types of tokens supported in python language?**
2. **Write any four keywords of python language**

\* In Python, there are approximately around thirty-three (33) keywords, and a few of the keywords generally used in the program coding are **break, continue, true, false, and, or, not, for, while, def, class, if, else, elif, import, from, except, exec, print, return, yield, lambda, global**, etc.

1. **What are the types of literals?**

Python has different types of literals.

* String literals.
* Numeric literals.
* Boolean literals.
* Literal Collections.
* Special literals.

1. **Explain Boolean literals**

\*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. Any non-zero value is True, and the value zero is False.

1. **What are relational operators?**

\*Relational operators are used for comparing the values. It either returns True or False according to the condition. These operators are also known as Comparison Operators.

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

\*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.

An assignment statement gives a value to a variable. For example, ... the variable may be a simple name, or an indexed location in an array, or a field (instance variable) of an object, or a static field of a class; and. the expression must result in a value that is compatible with the type of the variable .

1. **What is General Structure or General format or Syntax?**

\*The syntax of the Python programming language is the set of rules that defines how a Python program will be written and interpreted (by both the runtime system and by human readers).

1. **What are the types of comments? Explain with suitable examples**

There are three types of comments: **single-line, multi-line, and docstring comments**. The syntax of comments varies depending on the type. This tutorial will explore every kind of comment individually, along with examples.

### Example 1: Writing Single-Line Comments

### # printing a stringprint

### ('Hello world')

### '''

### I am a

### multiline comment!

### '''

### print("Hello World")

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

\*A variable is the name of a memory cell. It is "variable" because the value in the cell can change. Each memory cell has an address. Python and other high-level languages use a symbol table to map a variable name to the address it represents.

There are two parts of memory:

* stack memory
* heap memory

The methods/method calls and the references are stored in **stack memory** and all the values objects are stored in a **private heap.**

#### **Work of Stack Memory**

The allocation happens on contiguous blocks of memory. We call it stack memory allocation because the allocation happens in the function call stack. The size of memory to be allocated is known to the compiler and whenever a function is called, its variables get memory allocated on the stack.

It is the memory that is only needed inside a particular function or method call. When a function is called, it is added onto the program’s call stack. Any local memory assignments such as variable initializations inside the particular functions are stored temporarily on the function call stack, where it is deleted once the function returns, and the call stack moves on to the next task. This allocation onto a contiguous block of memory is handled by the compiler using predefined routines, and developers do not need to worry about it.

**Example:**

|  |
| --- |
| def func():   # All these variables get memory      # allocated on stack      a = 20      b = []      c = "" |

#### **Work of Heap Memory**

The memory is allocated during execution of instructions written by programmers. Note that the name heap has nothing to do with heap data structure. It is called heap because it is a pile of memory space available to programmers to allocated and de-allocate. The variables are needed outside of method or function calls or are shared within multiple functions globally are stored in Heap memory.

**Example:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # This memory for 10 integers  # is allocated on heap.  a = [0]\*10   1. **While dealing with dynamic typing what caution must be taken care of? Explain with suitable example**.   \*Dynamic Typing: the property of a language where type checks are performed mostly at run time. Traditionally (and mayhap not wisely) opposed to StaticTyping. Another term that is sometimes used is "value typed" opposed to "variable or reference typed".  For example Python allows one to add an integer and a floating point number, but adding an integer to a string produces error.  **PYTHON FUNDAMENTALS**  **Three Marks Questions**   1. **What are the python naming conventions?**   \*In Python programming, naming conventions are set of rules for choosing the valid name to be used for variable, package, module, class and function in a python program. Naming Conventions rules for Variables, Packages, Modules and Functions (Methods) are:  * It should begin with an alphabet. * It should be written in lowercase. * There may be more than one alphabet, but without any spaces between them. * Digits may be used but only after alphabet. * No special symbol can be used except the underscore (\_) symbol. When multiple words are needed, an underscore should separate them. * No keywords or command can be used. * All statements in Python language are case sensitive. Thus a name A (in uppercase) is considered different from a name a (in lowercase).  1. **Explain the representation of string in python language**.   \*Like many other popular programming languages, strings in Python are **arrays of bytes representing unicode characters**. However, Python does not have a character data type, a single character is simply a string with a length of 1. Square brackets can be used to access elements of the string.   1. **Explain the types of strings supported by python language.**   \*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.  There are 2 types of strings supported by python. Strings stored as characters and stored as bytes.  Strings stored as characters are represented as unicode in python 2 or str in python 3. Specifying a unicode string can be done by adding ‘u' before string. Eg u'hello' will be a unicode string.  Strings stored as bytes are represented as str in python 2 or bytes in python 3. Specifying a byte string can be done by adding ‘b' before string. Eg b'hello' will be a byte string.  In case of text formats such as UTF-8, python supports all formats.   1. **Explain escape sequences.**   \*Suppose you have the string containing a double backslash (\\\\), and you want to print the string with a double backslash, but it is impossible to print it directly. So here comes the role of escape sequences.  Firstly, understand sequences. 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.  An escape sequence is a sequence of characters that, when used inside a character or string, does not represent itself but is converted into another character or series of characters. So escape sequences are formed using two things: the first is a backslash (\\), and the second is the set of one or more characters following that backslash (\\). **List of Escape Sequence Available in Python**  | **Escape Sequence** | **Meaning** | | --- | --- | | \’ | Single quote | | \\’ | Double quote | | \\ | Backslash | | \n | Newline | | \r | Carriage Return | | \t | Horizontal Tab | | \b | Backspace | | \f | Formfeed | | \v | Vertical Tab | | \0 | Null Character | | \N{Name} | Unicode character Database named lookup | | \uxxxxxxxx | Unicode character with a 16-bit hex value | | \Uxxxxxxxx | Unicode character with a 32-bit hex value | | \000 | Character with octal value ooo | | \xhh | Character with hex value hh |  1. **Explain numerical literals supported in python language**.   \*Numeric literals are used to represent numbers in a python program.In python we have different types of numeric literals such as **integers, floating point numbers and complex numbers**. Integers in python are numbers with no fractional component. An integer representing a decimal number can be defined as follows   1. **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   + [Binary floating-point literals](https://www.ibm.com/docs/en/zos/2.4.0?topic=literals-floating-point#lit_fltpt__fp_decimal)   + [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) * [Complex literals](https://www.ibm.com/docs/en/zos/2.4.0?topic=literals-floating-point#lit_fltpt__complex_lit)   Floating-point literals are numbers that have a decimal point or an exponential part. They can be represented as: Real literals.   1. **Explain the General structure of python program and give example.**   \*The basic Python data structures in Python include list, set, tuples, anddictionary. Each of the data structures is unique in its own way. Data structures are “containers” that organize and group data according to type.  Dijkstra developed structural programming functionality as a widely used method, in which a program is divided into multiple sections with multiple exits and one access point. Modular programming is another example of structural programming, where a program is divided into interactive modules   1. **What is whitespace how its useful in python programming**   \*whitespace is a pre-initialized string used as string constant. In Python, string. whitespace will give the characters space, tab, linefeed, return, formfeed, and vertical tab. Syntax : string.whitespace. Parameters : Doesn't take any parameter, since it's not a function  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.   1. **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.  input function always takes the input data as String. So if you are asking the user to enter age, and he enters age as 25, this will be considered a string. Additionally, you need to convert it to int before you take any action on it.  Write the following in IDE and run it.  # taking input from the user  name = input()  print(name)  We have already seen the [***print function in Python,***](https://www.toolsqa.com/python/python-print-function/) which sends data to the console. ***Python Input function*** does the reverse of it. In other words, it takes user data from the console so that the program can take action based on that input.  The ***input*** function indicates the user to enter the data on the console. Moreover, the program executes further only when the user has entered the data. This data can then be stored in a variable and can use it in the program wherever we need it.   1. **What is print ( ) function? Write down the general format of print ( ) function and explain with proper example.**   \*If something is not right in the code, then we can use the print function to print what is happening in the code. Many times, we expect a certain value of a variable to be one thing, but we cannot see what our program sees.  If we use the print function to print out the value of a variable, then we will see what we thought was not present in our program. Python Print() Function Syntax/Format print(\*object, sep= “ ”, end = “\n”, file= sys.stdout, flush= False )   * **\*object:** One or more objects to be printed. * **sep:** Separator between objects. Default value = single space   **Example:**   |  | | --- | | ```  a = ‘Welcome’  b = ‘Python’  print(a, b, sep = ‘ , ‘)   ``` | |
|  |