Getting Started with Python

Very Short Answer type Questions [1 mark each]

Question 1:
What are the two modes in Python ?
Answer:
Interactive Mode Programming and Script Mode Programming.

Question 2:
Write any two Standard Data Types in Python.
Answer:
List and String.

Question 3:
Is List a standard data type ?
Answer:
Yes, List is a standard data type.

Question 4:
Write the kind of Python's dictionary ?
Answer:
Python's dictionaries are kind of hash table type.

Question 5:
What is the extension of Python language?
Answer:
All Python files have extension ".py".

Question 6:
Which mode of Python invoking the interpreter without passing a script file as a parameter ?
Answer:
Interactive Mode Programming.

Question 7:
Which mode of Python invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished ?
Answer:
Script Mode Programming.

Question 8:

In which mode of Python, the interpreter is no longer active ?

Answer:

Script Mode Programming.

Question 9:

Do Python variables have to be explicitly declared to reserve memory space ?

Answer:

Yes.

Question 10:

Does Python allows you to assign a single value to several variables simultaneously ?

Answer:

Yes, Python allows you to assign a single value to several variables simultaneously.

Question 11:

Give a example of immutable data type.

Answer:

Tuple.

Question 12:

Which type of values can be store in Number data types.

Answer:

numeric values.

Question 13:

Does Python allow for only double quotes ?

Answer:

No, Python does allow for either pairs of single or double quotes.

Question 14:

Write the name of most versatile Python's compound data types.

Answer:

Lists.

Question 15:

Which data type consists of a number of values separated by commas ?

Answer:

A tuple

Question 16:
What is IDLE?
Answer:
IDLE (pronounced as Idli) is the most popular Python development environment IDLE is an acronym of Integrated Develop Environment

Question 17:
Name some commands of Python.
Answer:
Copyright, help, credits.

Short Answer type Questions [2 mark each]

Question 1:
What is the difference between a keyword and an identifier ?
Answer:
Keyword is a special word that has a special meaning and purpose. Keywords are reserved and are few. For example : if, else, elif etc.
Identifier is the user-defined name given to a part of a program like variable, object, functions etc. Identifiers are not reserverd. These are defined by the user but they can have letters, digits and a symbols underscore. They must begin with either a letter or underscore. For example : chess, _ch, etc.

Question 2:
What are literals in Python ? How many types of literals are allowed in Python ?
Answer:
Literals mean constants i.e. the data items that never change value during a program run. Python allow five types of literals :

1. String literals
2. Numeric literals
3. Boolean literals
4. Special literal (None)
5. Literal collections like tuples, lists etc.

Question 3:
How many ways are there in Python to represent an integer literal ?
Answer:
Python allows three types of integer literals :

1. Decimal (base 10) integer literals.
2. Octal (base 8) integer literals.

3. Hexadecimal (base 16) integer literals.

   For example, decimal 12 will be written as 14 as octal integer and as OXC as hexa decimal integer. $(12)_{10} = (14)_8 = (OXC)_{16}$. (as hexa decimal)

Question 4:

How many types of strings are supported in Python ?

Answer:

Python allows two string types :

1. Single line strings : Strings that are terminated in single line. For example :
   str = 'Oswal Books'
2. Multiple strings : Strings storing multiple lines of text. For example :
   str = 'Owal \
   Books'
   or str = " " " Oswal
   Books
   " " "

Question 5:

What is "None" literal in Python ?

Answer:

Python has one special literal called 'None'. The 'None' literal is used to indicate something that has not yet been created. It is also used to indicate the end of lists in Python.

Question 6:

What factors guide the choice of identifiers in Programs ?

Answer:

1. An identifier must start with a letter or underscore followed by any number of digits or/ and letters.
2. No special character (other than under-score) should be included in the identifier.
3. No reserved word or standard identifier should be used.
4. Upper and lower case letters are different. All characters are significant.

Question 7:

What will be the size of the following constants : "\a". "\a", "Manoj\'s", '\", "XY\ YZ"

Answer:

'\a' – size is 1 as there is one character and it is a string literal enclosed in single quotes.
"\a" – size is 1 as there is one character enclosed in double quotes.
"Manoj\'s" – size is 7 because it is a string having 7 characters enclosed in double quotes.
'\" – size is 1. It is a character constant and is containing just one character \".
"XY\ – size is 4. It is a multiline string create YZ" with \ in the basic string.

Question 8:
What is the difference between a tuple and a list ?
Answer:
A tuple is immutable i.e. cannot be changed. It can be operated on only. But a list is mutable. Changes can
be done internally to it.
tuple initialization: a = (2,4,5) list initialization: a = [2,4,5]

Question 9:
Write various python modules convert the list to generate the output "one, two, three" ? a = ['one', 'two', 'three']
Answer:
> > > a = ['one','two','three']
>>> ','.join(a)
'one,two,three'

Question 10:
What would the following code yield ? word = 'abcdefghij'
Answer:abcdefghij
print word[:3] + word[3:]

Question 11:
Is there a tool to help find bugs or perform static analysis?
Answer:
Yes. PyChecker is a static analysis tool that finds bugs in Python source code and warns about code complexity and style. Pylint is another tool that checks if a module satisfies a coding standard, and also makes it possible to write plug-ins to add a custom feature.

Question 12:
What is a tuple ?
Answer:
A tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.

Question 13:
What is a list?
Answer:
Lists are the most versatile of Python's compound data types. A list contains items separated by commas and enclosed within square brackets ([ ]). To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different data type.

Question 14:

Explain String data type.

Answer:

Strings in Python are identified as a contiguous set of characters in between quotation marks. Python allows for either pairs of single or double quotes. Subsets of strings can be taken using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.

Question 15:

What is a Number data types ?

Answer:

Number data types store numeric values. They are immutable data types, which means that changing the value of a number data type results in a newly allocated object.

Question 16:

Write the names of all Standard Data Types.

Answer:

(a) Numbers
(b) String
(c) List
(d) Tuple
(e) Dictionary

Question 17:

Write a list comprehension that builds a list of the even numbers from 1 to 10 (inclusive).

Answer:

foo = [x for x in range(l, 11) if (x % 2) = = 0] print foo [2,4,6,8,10]

Question 18:

When do you use list vs. tuple vs. dictionary vs. set?

Answer:

'List' is like an array, individual element of list data can be accessed using indexing and can be manipulated. "Tuples" are similar to list, but there data can be changed once created through the execution of program. 'Set' stores unordered values and have no index. And unlike Tuples and lists, sets can have no duplicate data. "Dictionary" is similar to what their name is. It consist of pairs of keys and thier corresponding values.

Question 19:

Explain the dictionary in Python.

Answer:

- Python's built-in data type is dictionary, which defines one-to-one relationships between keys and values.
- Dictionary is indexed by keys.
- Dictionary is similar to association array or hash table of other languages.
- Dictionary consist of pairs'-of keys and their corresponding values.

Question 20:
What is PEP 8?
Answer:
PEP 8 is a coding convention(a set of recommendations) to write your Python code in order to make it more readable and useful for those after you.

Question 21:
Explain how Python is interpreted.
Answer:
Python program runs directly from the source code. Each type Python programs executed code is required. Python converts source code written by the programmer into intermediate language which is again translated into the native language/ machine language that is executed. So Python is an interpreted language.

Long Answer type Questions [4 mark each]

Question 1:
How do we share global variables across modules in Python?
Answer:
We can create a config file & store the entire global variable to be shared across modules or script in it. By simply importing config, the entire global variable defined. It will be available for use in other modules.
For example we want a, b & c to share between modules.
config.py :
a = 0
b = 0
c = 0
moduleI.py:
import config
config.a = 1
config.b = 2
config.c = 3
• print " a, b & c are : " , config.a, config.b, config.c

Question 2:

What are the rules for local and global variables in Python?

Answer:

In Python, variables that are only referenced inside a function are implicitly global. If a variable is assigned a new value anywhere within the function's body, it's assumed to be a local. If a variable is ever assigned a new value inside the function, the variable is implicitly local, and you need to explicitly declare it as 'global'. Though a bit surprising at first, a moment's consideration explains this. On one hand, requiring global for assigned variables provides a bar against unintended side-effects. On the other hand, if global was required for all global references, you will be using global all the time. You have to declare as global every reference to a builtin function or to a component of an imported module. This clutter would defeat the usefulness of the global declaration for identifying side-effect.

Question 3:

What does 'immutable' mean ? Which data type in Phython are immutable.

Answer:

An immutable variable change of value will not happen in place. Modifying an immutable variable will rebuild the same variable. For example,

>>> x= 5

will create a value 5 referenced by

x x —> 5

>>> y = x

This statement will make y refer to 5 of x.

x

5

y

>>> x = x + y

As x being, immutable type, has been rebuild. In the statement expression of RHS will result into value 10 and when this is assigned to LHS (x), x will rebuild to 10. An Integer data type in python is immutable.

Question 4:

Does Python support data type conversion ?

Answer:

Yes, Python support data type conversion. To convert between built-in types, programmer simply use the type name as a function. There are several built-in functions to perform conversion from one data type to another. These functions return a new object represting the converted value.

Function	Description
int (x [, base])	Converts x to an integer, base specifies the base if x is a string.
long [x [, base])	Converts x to a long integer, base specifies the base if x is a string.
float (x)	Converts x to a floatingpoint number.

complex (real, [ing])	Creates a complex number.
str (x)	Converts object x to a string representation.
repr (x)	Converts object x to an expression string.
tuple (x)	Converts x to a tuple.
list (x)	Converts x to a list.
chr (x)	Convets an integer to a character.
unichr (x)	Converts an integer to a Unicode character.
diet (x)	Creates a disetionary, x must be a sequence of tuples.
set (x)	Converts x to a set

Question 5:

How do you make an array in Python?

Answer:

Use a list: ["this", 1, "is", "an", "array"] Lists are equivalent to C or Pascal arrays in their time complexity. The primary difference is that a Python list can contain objects of many different types. The array module also provides methods for creating arrays of fixed types with compact representations, but they are slower to index than lists. Also note that the numeric extensions and others define array-like structures with various characteristics as well.

Question 6:

How do you make conversion between tuples and lists ?

Answer:

The function tuple (seq) converts any sequence (actually, any.iterable) into a tuple with the same items in the same order.

For example, tuple([1, 2, 3]) yields (1, 2, 3) and tuple ('abc') yields ('a', 'b', 'c'). If the argument if a tuple, it does not make a copy but returns the same object, so it is cheap to call tuple( ) when you aren't sure that an object is already a tuple.

The function list(seq) converts any sequence or iterable into a list with the same items in the same order. For example, list((1, 2, 3)) yields [1, 2, 3] and list ('abc') yields ['a', 'b', 'c'j. If the argument is a list, if makes a copy just like se[:j would.

Question 7:

What is the difference between list and tuple ? Give an example.

Answer:

Lists are Python's general purpose container, often used for collections of similar objects. Lists are mutable objects that can contain any Python data. Example :

>my list = [ ] # make an Empty list

>my list = [1,1.0+3j, "aperitivo", true] # make a list containing four entities.

A Tuple is an immutable value (can't be changed) that can contain any Python data. They are generally used for small collectioin of data.

Example :

>mytuple = [1, 2, 3, 4) # create a four element tuple.

>mytuple[2] = 4 # Error – tuple can't be modified. > print mytuple[2], len(mytuple)

Question 8:
What is used to represent Strings in Python ?
Answer:
Using Single Quotes (')
You can specify strings using single quotes such as 'Quote me on this'. All white space i.e. spaces and tabs are preserved as it is.
Using Double Quotes (")
Strings in double quotes work exactly the same way as strings in single quotes. An example is "What's your name?"
Using Triple Quotes o("' or " " ")
You can specify multi-line strings using triple
quotes. You can use single quotes and double
quotes freely within the triple quotes. An example
is
"'This is a multi-line string. This is the first line. This is the second line.
"What's your name?," I asked.
He said "syed saif naqvi."

Question 9:
Is there a tool help to find bugs or perform static analysis ?
Answer:
Yes, pychecker is a static analysis tool that finds bugs in Python source case and warns about code complexity and style.
Pylint is another tool that checks if a module satisfies a coding standard, and also make it possible to write plug-ins to add a custom feature.

Question 10:
Which of the following variable names are invalid ? Justify.
(a) try
(b) 123 Hello
(c) sum
(d) abc@123
Answer:
(a) try : is a keyword can't be used as an identifier.
(b) 123 Hello : Variable names can't start with a digit.
(c) abc@123 : Special characters aren't allowed in variable names.

Question 11:
Name four Python's basic data types. Why are they called so ?

Answer:
The four basic data types are :

1. Number
2. Sequences
3. Sets
4. Maps.

These are called so because :

1. Number data type stores numerical values and is immutable i.e., value of its object cannot be changed.
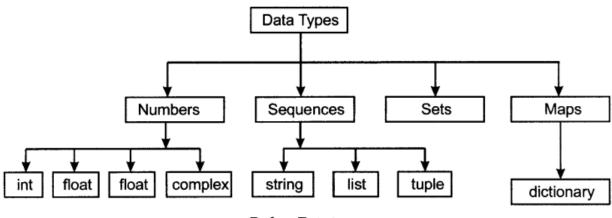   These are of 3 types :
   1. Integer and Long
   2. Float/Floatingpoint
   3. Complex
2. Sequence is an ordered collection of items, indexed by positive integers, It is a combination of mutable and non-mutable data types. Three types of sequence data type available in Phyton are strings, lists and tuples.
3. Sets in an unordered collection of values of any type, with no duplicate entry. Sets are immutable. Example S = Set ([1,2,3,4])
4. Mapping data types are unordered and mutable. Dictonaries falls under mapping.
   Example
   d = {1 : 'a', 2 : Tb', 3 : 'c'

Question 12:
Classify the Python data types.
Answer:

Python Data types