Q.1. What are the key features of Python?

If it makes for an introductory language to programming, Python must mean something. These are its qualities:

Interpreted

Dynamically-typed

Object-oriented

Concise and simple

Free

Has a large community

Q.2 We know Python is all the rage these days. But to be truly accepting of a great technology, you must know its pitfalls as well. Would you like to talk about this?

Of course. To be truly yourself, you must be accepting of your flaws. Only then can you move forward to work on them. Python has its flaws too:

Python Interview Question - limitations of Python

Python’s interpreted nature imposes a speed penalty on it.

While Python is great for a lot of things, it is weak in mobile computing, and in browsers.

Being dynamically-typed, Python uses duck-typing (If it looks like a duck, it must be a duck). This can raise runtime errors.

Python has underdeveloped database access layers. This renders it a less-than-perfect choice for huge database applications.

And then, well, of course. Being easy makes it addictive. Once a Python-coder, always a Python coder.

Q.3. How do you insert an object at a given index in Python?

Let’s build a list first.

>>> a=[1,2,4]

Now, we use the method insert. The first argument is the index at which to insert, the second is the value to insert.

>>> a.insert(2,3)

>>> a

[1, 2, 3, 4]

Q.4. And how do you reverse a list?

Using the reverse() method.

>>> a.reverse()

>>> a

[4, 3, 2, 1]

You can also do it via slicing from right to left:

>>> a[::-1]

>>> a

[1, 2, 3, 4]

This gives us the original list because we already reversed it once. However, this does not modify the original list to reverse it.

Q.5. What is the Python interpreter prompt?

This is the following sign for Python Interpreter:

>>>

If you have worked with the IDLE, you will see this prompt.

Q.6. Why do we need break and continue in Python?

Both break and continue are statements that control flow in Python loops. break stops the current loop from executing further and transfers the control to the next block. continue jumps to the next iteration of the loop without exhausting it.

Q.7. Will the do-while loop work if you don’t end it with a semicolon?

Trick question! Python does not support an intrinsic do-while loop. Secondly, to terminate do-while loops is a necessity for languages like C++.

Q.8. What will the following code output?

>>> word=’abcdefghij’

>>> word[:3]+word[3:]

The output is ‘abcdefghij’. The first slice gives us ‘abc’, the next gives us ‘defghij’.

Q.9. What are membership operators?

With the operators ‘in’ and ‘not in’, we can confirm if a value is a member in another.

>>> 'me' in 'disappointment'

True

>>> 'us' not in 'disappointment'

True

Q.10. Explain identity operators in Python.

The operators ‘is’ and ‘is not’ tell us if two values have the same identity.

>>> 10 is '10'

False

>>> True is not False

True

Q.11. What is a docstring?

A docstring is a documentation string that we use to explain what a construct does. We place it as the first thing under a function, class, or a method, to describe what it does. We declare a docstring using three sets of single or double-quotes.

>>> def sayhi():

"""

The function prints Hi

"""

print("Hi")

>>> sayhi()

Hi

To get a function’s docstring, we use its \_\_doc\_\_ attribute.

>>> sayhi.\_\_doc\_\_

‘\n\tThis function prints Hi\n\t’

A docstring, unlike a comment, is retained at runtime.

Q.12. What is recursion?

When a function makes a call to itself, it is termed recursion. But then, in order for it to avoid forming an infinite loop, we must have a base condition.

Let’s take an example.

>>> def facto(n):

if n==1: return 1

return n\*facto(n-1)

>>> facto(4)

Q.13. What does the function zip() do?

One of the less common functions with beginners, zip() returns an iterator of tuples.

>>> list(zip(['a','b','c'],[1,2,3]))

[(‘a’, 1), (‘b’, 2), (‘c’, 3)]

Here, it pairs items from the two lists and creates tuples with those. But it doesn’t have to be lists.

>>> list(zip(('a','b','c'),(1,2,3)))

[(‘a’, 1), (‘b’, 2), (‘c’, 3)]

Q.14. How do you calculate the length of a string?

This is simple. We call the function len() on the string we want to calculate the length of.

>>> len('Ayushi Sharma')

13

Q.15. Explain Python List Comprehension.

The list comprehension in python is a way to declare a list in one line of code. Let’s take a look at one such example.

>>> [i for i in range(1,11,2)]

[1, 3, 5, 7, 9]

>>> [i\*2 for i in range(1,11,2)]

[2, 6, 10, 14, 18]

Q.16. How do you get all values from a Python dictionary?

We saw previously, to get all keys from a dictionary, we make a call to the keys() method. Similarly, for values, we use the method values().

>>> 'd' in {'a':1,'b':2,'c':3,'d':4}.values()

False

>>> 4 in {'a':1,'b':2,'c':3,'d':4}.values()

True

Q.17. What if you want to toggle case for a Python string?

We have the swapcase() method from the str class to do just that.

>>> 'AyuShi'.swapcase()

‘aYUsHI’

Let’s apply some concepts now, shall we? Questions 50 through 52 assume the string ‘I love Python’. You need to do the needful.

Q.18. Write code to print only upto the letter t.

>>> i=0

>>> while s[i]!='t':

print(s[i],end=’’)

i+=1

I love Py

Q.19. Write code to print everything in the string except the spaces.

>>> for i in s:

if i==' ': continue

print(i,end='')

IlovePython

Q.20. Now, print this string five times in a row.

>>> for i in range(6):

print(s)

I love Python

I love Python

I love Python

I love Python

I love Python

I love Python

Okay, moving on to more domains to conquer.