Now, you can begin reading the questions/answers given in the below section. All of these are related to the application of Python and would test your scripting skills of the language.

* [**What is the function to randomize the items of a list in-place**](https://www.techbeamers.com/python-interview-questions-experienced/#q1)**?**
* [**What is the best way to split a string in Python?**](https://www.techbeamers.com/python-interview-questions-experienced/#q2)
* [**What is the right way to transform a Python string into a list?**](https://www.techbeamers.com/python-interview-questions-experienced/#q3)
* [**How does exception handling in Python differ from Java? Also, list the optional clauses for a “try-except” block in Python?**](https://www.techbeamers.com/python-interview-questions-experienced/#q4)
* [**What do you know about the “list” and “dict” comprehensions? Explain with an example.**](https://www.techbeamers.com/python-interview-questions-experienced/#q5)
* [**What are the methods you know to copy an object in Python?**](https://www.techbeamers.com/python-interview-questions-experienced/#q6)
* [**Can you write code to determine the name of an object in Python?**](https://www.techbeamers.com/python-interview-questions-experienced/#q7)
* [**Can you write code to check whether the given object belongs to a class or its subclass?**](https://www.techbeamers.com/python-interview-questions-experienced/#q8)
* [**What is the result of the following Python program?**](https://www.techbeamers.com/python-interview-questions-experienced/#q9)
* [**What is the result of the below lines of code?**](https://www.techbeamers.com/python-interview-questions-experienced/#q10)
* [**What is the result of the below Python code?**](https://www.techbeamers.com/python-interview-questions-experienced/#q11)
* [**How would you produce a list with unique elements from a list with duplicate elements?**](https://www.techbeamers.com/python-interview-questions-experienced/#q12)
* [**Can you iterate over a list of words and use a dictionary to keep track of the frequency(count) of each word? Consider the below example.**](https://www.techbeamers.com/python-interview-questions-experienced/#q13)
* [**What is the result of the following Python code?**](https://www.techbeamers.com/python-interview-questions-experienced/#q14)
* [**Can you describe what’s wrong with the below code?**](https://www.techbeamers.com/python-interview-questions-experienced/#q15)

Top 15 Python Questions And Answers For Experienced.

**Python Questions and Answers for Experienced**

💡 **Fact – Python assumes a TAB equal to 8 Spaces.**

Q-1. What Is The Function To Randomize The Items Of A List In-Place?

**Ans.** Python has a built-in module called as <random>. It exports a public method <shuffle(<list>)> which can randomize any input sequence.

import random

list = [2, 18, 8, 4]

print "Prior Shuffling - 0", list

random.shuffle(list)

print "After Shuffling - 1", list

random.shuffle(list)

print "After Shuffling - 2", list

Q-2. What Is The Best Way To Split A String In Python?

**Ans.** We can use Python <split()> function to break a string into substrings based on the defined separator. It returns the list of all words present in the input string.

test = "I am learning Python."

print test.split(" ")

Program Output.

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

['I', 'am', 'learning', 'Python.']

Q-3. What Is The Right Way To Transform A Python String Into A List?

**Ans.** In Python, strings are just like lists. And it is easy to convert a string into the list. Simply by passing the string as an argument to the list would result in a string-to-list conversion.

list("I am learning Python.")

Program Output.

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

=> ['I', ' ', 'a', 'm', ' ', 'l', 'e', 'a', 'r', 'n', 'i', 'n', 'g', ' ', 'P', 'y', 't', 'h', 'o', 'n', '.']

Q-4. How Does Exception Handling In Python Differ From Java? Also, List The Optional Clauses For A <Try-Except> Block In Python?

**Ans.** Unlike Java, Python implements exception handling in a bit different way. It provides an option of using a <try-except> block where the programmer can see the error details without terminating the program. Sometimes, along with the problem, this <try-except> statement offers a solution to deal with the error.

There are following clauses available in Python language.

**1.** try-except-finally  
**2.** try-except-else

💡 **Must Read –**[**30 Most Important Python Interview Questions and Answers**](https://www.techbeamers.com/python-interview-questions-programmers/)**.**

Q-5. What Do You Know About The <List> And <Dict> Comprehensions? Explain With An Example.

**Ans.** The <List/Dict> comprehensions provide an easier way to create the corresponding object using the existing iterable. As per official Python documents, the list comprehensions are usually faster than the standard loops. But it’s something that may change between releases.

The <List/Dict> Comprehensions Examples.

#Simple Iteration

item = []

for n in range(10):

item.append(n\*2)

print item

#List Comprehension

item = [n\*2 for n in range(10)]

print item

Both the above example would yield the same output.

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

[0, 2, 4, 6, 8, 10, 12, 14, 16, 18]

#Dict Comprehension

item = {n: n\*2 for n in range(10)}

print item

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

{0: 0, 1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 12, 7: 14, 8: 16, 9: 18}

💡 **Fact – In interactive mode, the last printed expression is assigned to the variable \_ (underscore).**

Q-6. What Are The Methods You Know To Copy An Object In Python?

**Ans.** Commonly, we use <copy.copy()> or <copy.deepcopy()> to perform copy operation on objects. Though not all objects support these methods but most do.

But some objects are easier to copy. Like the dictionary objects provide a <copy()> method.

Example.

item = {n: n\*2 for n in range(10)}

newdict = item.copy()

print newdict

Q-7. Can You Write Code To Determine The Name Of An Object In Python?

**Ans.** No objects in Python have any associated names. So there is no way of getting the one for an object. The assignment is only the means of binding a name to the value. The name then can only refer to access the value. The most we can do is to find the reference name of the object.

Example.

class Test:

def \_\_init\_\_(self, name):

self.cards = []

self.name = name

def \_\_str\_\_(self):

return '{} holds ...'.format(self.name)

obj1 = Test('obj1')

print obj1

obj2 = Test('obj2')

print obj2

Q-8. Can You Write Code To Check Whether The Given Object Belongs To A Class Or Its Subclass?

**Ans.** Python has a built-in method to list the instances of an object that may consist of many classes. It returns in the form of a table containing tuples instead of the individual classes. Its syntax is as follows.

<isinstance(obj, (class1, class2, ...))>

The above method checks the presence of an object in one of the classes. The built-in types can also have many formats of the same function like <isinstance(obj, str)> or <isinstance(obj, (int, long, float, complex))>.

Also, it’s not recommended to use the built-in classes. Create an user-defined class instead.

We can take the following example to determine the object of a particular class.

Example.

def lookUp(obj):

if isinstance(obj, Mailbox):

print "Look for a mailbox"

elif isinstance(obj, Document):

print "Look for a document"

else:

print "Unidentified object"

Q-9. What Is The Result Of The Following Python Program?

**Ans.** The example code is as follows.

def multiplexers ():

return [lambda n: index \* n for index in range (4)]

print [m (2) for m in multiplexers ()]

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

[6, 6, 6, 6]

The output of the above code is <[6, 6, 6, 6]>. It’s because of the late binding as the value of the variable <index> gets looked up after a call to any of multiplexers functions.

💡 **Also Read – [20 Python Programming Interview Questions for Practice](https://www.techbeamers.com/python-programming-interview-questions-with-answers/" \t "_blank).**

Q-10. What Is The Result Of The Below Lines Of Code?

Here is the example code.

def fast (items= []):

items.append (1)

return items

print fast ()

print fast ()

**Ans.** The above code will give the following result.

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

[1]

[1, 1]

The function <fast> evaluates its arguments only once after the function gets defined. However, since <items> is a list, so it’ll get modified by appending a <1> to it.

💡 **Fact – You can inspect objects in Python by using dir().**

Q-11. What Is The Result Of The Below Python Code?

keyword = 'aeioubcdfg'

print keyword [:3] + keyword [3:]

**Ans.** The above code will produce the following result.

<'aeioubcdfg'>

In Python, while performing string slicing, whenever the indices of both the slices collide, a <+> operator get applied to concatenates them.

Q-12. How Would You Produce A List With Unique Elements From A List With Duplicate Elements?

**Ans.** Iterating the list is not a desirable solution. The right answer should look like this.

duplicates = ['a','b','c','d','d','d','e','a','b','f','g','g','h']

uniqueItems = list(set(duplicates))

print sorted(uniqueItems)

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h']

Q-13. Can You Iterate Over A List Of Words And Use A Dictionary To Keep Track Of The Frequency(Count) Of Each Word? Consider The Below Example.

{'Number':Frequency, '2':2, '3':2}

**Ans.** Please find out the below code.

def dic(words):

wordList = {}

for index in words:

try:

wordList[index] += 1

except KeyError:

wordList[index] = 1

return wordList

wordList='1,3,2,4,5,3,2,1,4,3,2'.split(',')

print wordList

print dic(wordList)

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

['1', '3', '2', '4', '5', '3', '2', '1', '4', '3', '2']

{'1': 2, '3': 3, '2': 3, '5': 1, '4': 2}

Q-14. What Is The Result Of The Following Python Code?

class Test(object):

def \_\_init\_\_(self):

self.x = 1

t = Test()

print t.x

print t.x

print t.x

print t.x

**Ans.** All print statement will display <1>. It’s because the value of object’s attribute(x) is never changing.

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.8.2] on linux

1

1

1

1

Also, <x> becomes a part of the public members of class Test.

Hence, it can be accessed directly.

💡 **More Questions – [Top 10 Python Questions Every Developer Should Know](https://www.techbeamers.com/10-python-interview-questions/).**

Q-15. Can You Describe What’s Wrong With The Below Code?

testProc([1, 2, 3]) # Explicitly passing in a list

testProc() # Using a default empty list

def testProc(n = []):

# Do something with n

print n

**Ans.** The above code would throw a <NameError>.

The variable n is local to the function <testProc> and can’t be accessed outside.

So, printing it won’t be possible.