Python Interview Questions

1. What is mutable and Immutable?

Everything in Python is Object. Mutable objects can be changed after it is created while the immutable object cannot change.

Objects of built-in types like (int, float, bool, str, tuple, unicode) are immutable.

Objects of built-in types like (list, set, dict) are mutable. Custom classes are generally mutable.

2. Write sample syntax for Dictionary?

A={“srini”:’rao’, “sal” : 1000}

3. List Vs Tuple?

List – List is mutable **[**1,2,3,4,6**]**

Square brackets for List.

Best example: We can replace/append any value. You can mask middle 6 letters in credit card number, which is a value in List

Tuple – Immutable **(**1,2,3,4,5**)**

Parenthesis for Tuple.

Best example: Name in the Tuple, you cannot replace.

4. Local variable Vs Global Variable?

The variable, which is defined inside the function is called **local variable** and it is local to function.

The variable, which is defined outside the function is called **Global variable**.

order=”maggi” *==> Global*

```
def myfunction():
        order=”roti” ==> Local
myfunction()
```

5. What is Global Keyword in Python?

If we use “Global” keyword for a variable inside the function. Then it will access Global variable.

Suppose in the function, if we don’t mention the **Global** keyword, but if we try to use that, it gives error.

```
var = 321

# function to modify the variable

def modify():

    global var  ==> Using Global keyword

    var = var * 2

    print(var)

# calling the function

modify()
```

6. What happens if we try to replace a value in Tuple?

Tuple is immutable in Python, that means we cannot replace/modify it. If we try to do that, we will get an error.

TypeError: ‘tuple’ object does not support item assignment

7. What is List Comprehension?

It is concise way of creating list. It is popularly called List Comprehension.

Best Examples:

```
a=[n*2 for n in range(10)]

print(a)

myList = [ num**2 for num in range(1,10) if num %2==0]

print(myList)
```

Output:

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

[4, 16, 36, 64]

Similarly, you can check for Dictionary comprehension.

8. What is Zip() function?

The zip() function takes iterables (List, Tuple) as arguments and create iterator.

```
mylist=(100,200,300)
mytuple=(500,600,800)
myoutput=zip(mylist, mytuple)
myiterator=set(myoutput)
print(myiterator)
Output:
{(200, 600), (100, 500), (300, 800)}
```

9. What is iterator?

In general, iterators are repetition, and you can read iterables by using iter() and next() function

mylist=[1,2,3.4,5]

newlist=iter(mylist)

iter1=next(newlist)

print(iter1)

Output:

1

10. What is interpreter?

Python is interpreted language. It converts source code to machine understandable code (by using dedicated in-built virtual machine). Best examples are Python, Ruby, java.

Here is reference link for best explanation on interpreted language.

11. What are 'Yield' and "Generator' in Python?

Basically, it is a function. If we give yield in any function, that function is called 'Generator'

Script name abc.py

```
def course_generate():
  yield("C")
  yield("C++")
  yield("Java")
  yield("Python")
  yield("Php")
  yield("Vb.Net")
  yield("Asp.Net")
  yield("Android")
```

Script mygen.py

```
from course_generate import course_generate
course = course_generate() # function is called here
print(next(course)) # only first element will be printed i.e. "C"
```

12. What is Decorator?

Calling another function in a function is called Decorator.

13. What is Meta class?

With Meta class you can create Classes.

Python interview Questions on trees

1.What is tree?

Tree is an user-defined data structure

2. How the data structure looks like?

A Tree is a collection of nodes, each node consists of values and reference of nodes (known as **child nodes**).

3. How can we access nodes in a Tree?

All nodes in a tree are accessible through the root node of the tree.

4. Are trees mutable?

Trees are mutable.

5. Can we add or delete child (leaf) nodes?

Yes, you can add or delete child nodes.

6. Write code to create a node?

```
class Node():
   def __init__(self,val):
      self.val = val
      self.child = []
   def add_child(self,node):
      self.child.append(node)
```

The Node class has two methods. The init and add_child are two methods. **You know that init method is a constructor,** and it executes when you call a class (Here is Node) default.

The child nodes you are maintaining in a list. That's you can see in the init method. The add_child methods takes input and create a list of child nodes.

7. How to add child nodes?

Below is the logic to add child nodes.

```
class Node():
   def __init__(self,val):
      self.val = val
      self.child = []
   def add_child(self, node):
      self.child.append(node)


#step1
root = Node(1)
print(root)

#step2
print(root.child)

#step3
child1 = Node(2)
child2 = Node(3)
root.add_child(child1)
root.add_child(child2)
print (root.child)

#step4
for c in root.child:
   print(c.val)

print(root.val)
```

Output

The output generated from the code that we have written.

```
<__main__.Node object at 0x7f7cb279dee0>
[]
[<__main__.Node object at 0x7f7cb279df70>, <__main__.Node object at 0x7f7cb27c9400>]
2
3
1


** Process exited - Return Code: 0 **
Press Enter to exit terminal
```

8. What is the python library that is available to implement trees?

Anytree

9. What are the key points to remember?

- Tree is a user-defined data structure that can be implemented using a list or anytree library of Python.
- N-ary tree node can have up to n child nodes.
- Majorly used trees are binary trees and binary search trees.
- Binary and Binary Search trees can be implemented using the `binarytree` library available for Python.
- Binary trees and BST nodes can have 0 to 2 child nodes.
- Parent node in BST must have a value greater than the left child node and less value than the right child node.
- Trees have major use case in hierarchical data like file system, syntax tree in compilation, and so on.

10. What is BST (Binary search tree)?

The **Binary search tree** (**BST**) is a special type of binary tree. A tree is known as BST if it fulfills the following properties:

- Each node has maximum two child nodes.
- Left child has value less than parent node.
- Right child has value more than parent node

10 Must read python interview questions

Python Interview questions

Python is a powerful scripting language. And you can use it in the web development, and science applications. Below, you will find interview questions on core python and web-framework as well.

1. What is PEP8?

PEP8 is a set of rules to format Python code for better readability. Further, you can read the user guide on PEP8.

2. What is memory management, and explain in detail?

Python manages memory automatically - cleaning, allocating, and managing; the detailed mechanism is as follows:

In the interview, you need to tell as below:

- Python manages private heap space and is accessible to the only interpreter.
- In heap space, Python stores objects and data structures.
- The memory allocation is the responsibility of the memory manager
- Python heap divides into pools (1 heap=64 pool), and the Pool divides into blocks (256 KB).
- Python has an inbuilt garbage collector, which recycles the unused memory and frees the memory.

3. What are the different data types?

- int
- string
- list
- tuple
- dictionary
- Boolean

Check here more on data types.

4. What is Python Copy an object?

The '=' operator and COPY module you can use to copy objects. Here are the examples.

5. What are the various types of inheritances?

There are different types of inheritances:

- Single inheritance
- Multiple-inheritance

- Multi-level inheritance
- Hierarchical inheritance
- Hybrid inheritance

Here is the example for each inheritance.

6. What are the differences between List and Tuple?

The list is mutable. But, tuple is immutable. Check out more differences between these two.

7. What is List Comprehension?

It is a concise way of creating a list and is popularly called List Comprehension.

Example:

```
a=[n*2 for n in range(10)]
print(a)
myList = [ num**2 for num in range(1,10) if num %2==0]
print(myList)
```

Python List comprehension examples.

8. What is dictionary comprehension?

The method of transforming one Dictionary into another Dictionary is called dictionary comprehension. Here are more examples of Dictionary comprehension.

9. What is slicing?

You can slice an object using the slice() method, which allows three parameters - start, length, and separator you can give.

Here is the best example for the slice method.

10. What is a negative index?

Internally python assigns to '-1' for the last occurrence of a string. Python assigns a negative index (-1, -2, and so on) in reverse order. Here is the best example for a negative index.

References

- Python for everybody