1Q. Why tuples has less memory comapre to list?

ans: As tuples are **immutable object** it is stored in single block of memory so it requires less memory compare to list

list is **mutable object**, it is stored in two blocks of memory (one is while declaring a list and other is while modifying the list).so it requires more memory compare to tuple

2Q. Why tuples are faster while iterating compare to lists?

Q. What is difference between list and tuples?

Q. How to send an Emails and bulk emails in python?

Q. \*arg and \*\*kwargs difference.

if we are not aware of number of argument required by the function we can pass \*arg(\*anyname) which stores single or multiple argument in form tuple and

\*\*kwargs(\*\*anyname) is used pass multiple key arguments

Q. default, keyword arguments:

default arguments are used to specify default value to the argument.If the function is called with value then the function is executed with provided value,

otherwise, it is executed with the default value

Keyword arguments are also referred to as named arguments which are used to specify the names for arguments while calling the function

so these arguments are passed by itâ€™s name instead of their position.

Q. Copy Object (deep copy and shallow Copy)

python support two types of copies i.e, shallow copy and deep copy which are imported from built-in module copy. copy() method is used shallow copy the object

and deepcopy() is used to deep copy the object.

shallow copy: shallow copy creates new object from original object with its reference, so any changes made in the values of new object

will also reflected in original object

deep copy: deep copy creates new object from orginial object without its reference, so any changes made in values of new object

will not get reflected in orginal object

Q. What is Scope in Python?

ans:The part of the program where the variable is accessible is called its **scope** and it is determined by where the variable is declared.

Python has three different variable scopes:

**Global scope**:A variable declared outside all functions has a GLOBAL SCOPE. It is accessible throughout the file, and also

inside any file which imports that file.

**Local scope:**A variable declared within a function has a LOCAL SCOPE. It is accessible within the function,and exists as long as the function is executing.

**Enclosing scope:** a variables declared inside nested function is called enclosed scope. These are used when a variable is neither in the local scope

nor in global scope.

Q. Indexing and Negative Indexing (How can You reverse a string through indexing?)

ans:indexing is used to for accessing specific character in the string by passing position of that character in square brackets.

**positive indexing**: if the character of string accessed in forward direction i.e., from left to right indexing starts from zero to n-1 where is length of string

**negative indexing**: if the character of string acccessed in backward direction i.e., from right to left indexing starts from -1 to -n

Q. range and Xrange difference?

Ans:

|  |  |
| --- | --- |
| range | xrange |
| In python version 3 range function behaves like xrange function in python version 2.x with core functionality remains same but difference in terms of return type | In python version 2.x both range and xrange are built in function |
| range() function return list of integers | xrange() function returns a generator object. This generator object can be used to return a list of integers by using for loop |
| range() function consumes more memory as it returns a list of integers  Eg: a=range(1,13,2)  print(sys.getsizeof(a)) #out:120 | xrange() function consumes less memory compare to range()  Eg: b=xrange(1,13,2)  print(sys.getsizeof(a)) #out:40 |
| We can perform addition, deletion and slicing operations on range()  Eg: a=range(1,13,2)  print(f”slicing of {a[3:8])}”) | We cannot perform on xrange() as the return type is generator function  Eg: b=range(1,13,2)  print(f”slicing of {b[3:8])}”) |
| range() function are slower than xrange() | xrange() evaluates the generator object only when it is required therefore it is faster in implementation than range()a |

Q. Class, Method , Self and Init

Ans: **class** is a blueprint or template which contains common properties and methods of the objects. Class is the logical entity that is it does not require any memory allocation for the class until its object is created.

To create a class 'class' keyword is used and followed by Classname and semicolon. class name must follow camelcase.

**Method:** methods are the functions inside the class which defines the behaviour of objects

**self:**it is a parameter and it is used to represent the instance of the class. It is the default variable that is always pointing to the current object.

By using self, we can access the instance variable and instance method of the object. While defining constructor and instance method, the self is their first parameter.

**\_\_init()\_\_:** is a special type of method, the purpose of the \_\_init()\_\_ is to declare and initialize instance variables. It can take at least one argument that is self.

The \_\_init(self)\_\_ method is called the constructor in Python.

Q. inheritence concepts?

Ans**: Inheritence** allows creation of derived which inherits all methods and properties from a base class and it also contain its own methods and properties

python supports Single Inheritance, Multiple level Inheritance, Multi level Inheritance, Hierarchical Inheritance

**Single Inheritance:**

If a derived class inherits methods and properties from a single-base class is known as single inheritance here we have one base and one derived class

**Hierarchical Inheritance:**

If more than one derived class inherits the properties and methods of a single base class then it is known as hierarchical inheritance.

Here we have one base class and multiple derived classes

**Multiple level Inheritance:**

If one derived class can inherits methods and properties of multiple base classes. Then it is known as multiple Inheritance.

Here we have one dervied class and multiple base class

**Multi level Inheritance:**

If a derived class is inherited from another derived class then it is called multilevel inheritance.. Suppose A, B, C are the three classes. A is the superclass,

B is the derived class of A, C is the derived class of B. In other words, we can say a chain of classes.

Q. Polymorphism?

Ans: polymorphism means implementing multiple functionalities with same name is called polymorphism, In python polymorphism can be achieved in three ways:

**Method overloading**:a method with the same name but different number of arguments is called method overloading. In other programming languages like JAVA it supports Method overloading directly .But python doesn't support Method overloading directly. So to use Method Overloading in python we have to use a concept called as default argument.

**Operator overloading:** operators are used to performing specific operations on given operands. if the same operator performs different operations based on different types of operands is called operator overloading.

**Method overriding:** implementing super class methods within the subclass with same name and same number of arguments but with different functionality then subclass method overrides the super class method this process is known as method overriding.

Q. What is pass in Python?

Ans: pass is used as a placeholder when implementing new function so that functionality can be implemented later. Pass statement executes empty block without error

Q. Can We write n- number of init methods inside a class?

Q. Method resolution order?

Q. How to call Override method in Base class?

Q. Dundler methods?.

Q. Abstracion.

Q. Aggregation and Composition.

Q. What is Object() method?

Q. is cyclic inheritence possible in python?

Q. What are the access modifiers?

Q. What is function overloading and Operator Overloading ? What is difference?

Q. What is a ternary operator?

Q. What are the different types of arguments? What is meant by call by value and call by reference?

Q. What is method overriding?

Q. What are modules and packages in Python?

Q. What is self in Python?

Q. What is \_\_init\_\_?

Q. What is break, continue and pass in Python?

Q. What are Dict and List comprehensions?

Q. What is pickling and unpickling?

Q. What is the difference between .py and .pyc files?

.py files contain the source code of a program. Whereas, .pyc file contains the bytecode of your program. We get bytecode after compilation of .py file (source code)

.pyc files are not created for all the files that you run. It is only created for the files that you import.

Q.Write python function which takes a variable number of arguments.

Q. WAP (Write a program) which takes a sequence of numbers and check if all numbers are unique.

Q. Write a program to check and return the pairs of a given array A whose sum value is equal to a target value N.

Q. Reverse a list in different approaches ?

Q. Differentiate between deep and shallow copies?

Q. What is main function in python? How do you invoke it?

Q. Write a program to check and return the pairs of a given array A whose sum value is equal to a target value N?

Q. Write a program to get all the indexes of a value in a given list?

Q. Write a Program to combine two different dictionaries. While combining, if you find the same keys, you can add the values of these same keys. Output the new dictionary

Q. myJavaProgram -- > my\_java\_program and vice versa?

Q. Swap all zeroes to last in the string?

[1,0,1,0,0,1,1,0] --- [1,1,1,1,0,0,0,0]

if i[

Q. list down all the prime numbers which are below to given number?

Q. return only the nth number prime number?

Q. return the factorial function in recusrion mode?

Q. #Given a number n find the number of

#pairs (x,y) where both x and y are less than n and highest common factor (hcf) of x and y is 1?

Q. reverse the integer? 34523 -- >32543

Q. &

&&

&&&

&&&&

&&&&&

for i in range(5):

for j in range(i,5):

print(" ",end="")

for k in range(i+1):

print("&", end="")

print("")

&&&&

&&&

&&

&

for i in range(4):

for j in range(i,4):

print("&",end="")

print("")

\*

\*\*

\*\*\*

\*\*\*\*

for i in range(4):

for j in range(i+1):

print("\*", end="")

print("")

####

####

####

####

for i in range(4):

for j in range(4):

print("#", end="")

print("")

Q. list\_input = [2,4,12,6,2,9,10]

# out\_put = [2,8,-1,3,7,1,-1] --- If element has highest element then write the difference btween element ad the highest element of first occurence

Q.sort a list.

Q. find second max element in a list.