|  |
| --- |
| What are the ways to write a function using call by reference?  Arguments in python are passed as an assignment. This assignment creates an object that has no relationship between an argument name in source and target. The procedure to write the function using call by reference includes: The tuple result can be returned to the object which called it. The example below shows it:  def function(a, b): a = 'value'  b = b + 1  # a and b are local variables that are used to assign the new objects return a, b  # This is the function that is used to return the value stored in b  - The use of global variables allows the function to be called as reference but this is not the safe method to call any function.  - The use of mutable (they are the classes that consists of changeable objects) objects are used to pass the function by reference.  def function(a): a[0] = 'string'  a[1] = a[1] + 1  # The ‘a’ array give reference to the mutable list and it changes the changes that are shared args = ['string', 10] func1(args) print args[0], args[1]  #This prints the value stored in the array of ‘a’ |
| What are the commands that are used to copy an object in Python?  The command that is used to copy an object in python includes:  - copy.copy() function: This makes a copy of the file from source to destination. It returns a shallow copy of the parameter that is passed.  - copy.deepcopy(): This also creates a copy of the object from source to destination. It returns a deep copy of the parameter that is passed to the function.  The dictionary consists of all the objects and the copy() method which is used as:  newdict = olddict.copy()  The assignment statement doesn’t copy any object but it creates a binding between the target and the object that is used for the mutable items. Copy is required to keep a copy of it using the modules that is provided to give generic and shallow operations. |
| What is the difference between deep and shallow copy?  - Shallow copy is used when a new instance type gets created and it keeps the values that are copied in the new instance. Whereas, deep copy is used to store the values that are already copied.  - Shallow copy is used to copy the reference pointers just like it copies the values. These references point to the original objects and the changes made in any member of the class will also affect the original copy of it. Whereas, deep copy doesn’t copy the reference pointers to the objects. Deep copy makes the reference to an object and the new object that is pointed by some other object gets stored. The changes made in the original copy won’t affect any other copy that uses the object.  - Shallow copy allows faster execution of the program and it depends on the size of the data that is used. Whereas, deep copy makes it slower due to making certain copies for each object that is been called. |
| Write a program to find out the name of an object in python.  The object doesn’t have any name and there is no way the can be found out for objects. The assignment is used to bind a name to the value that includes the name of the object that has to be bound by a value. If the value is callable then the statements are made true and then the program followed can be used to find the reference name of an object.  class try: pass B = A a = B() b = a print b <\_\_main\_\_.try instance at 0x16D07CC> print b  The class consists of name and the names are invoked by using the the variable B that creates an instance for the class try. The method is to find out from all the namespaces that the object exists and then print the name of the object. |
| How can the ternary operators be used in python?  The ternary operator is the operator that is used to show the conditional statements. This consists of the true or false values with a statement that has to be evaluated for it. The operator will be given as:  [on\_true] if [expression] else [on\_false] x, y = 25, 50 big = x if x < y else y  This is the lowest priority operator that is used in making a decision that is based on the values of true or false. The expression gets evaluated like if x<y else y, in this case if x<y is true then the value is returned as big=x and if it is incorrect then big=y will be sent as a result. |
| How the string does get converted to a number?  - To convert the string into a number the built-in functions are used like int() constructor. It is a data type that is used like int (‘1’) ==1.  - float() is also used to show the number in the format as float(‘1’)=1. - The number by default are interpreted as decimal and if it is represented by int(‘0x1’) then it gives an error as ValueError. In this the int(string,base) function takes the parameter to convert string to number in this the process will be like int(‘0x1’,16)==16. If the base parameter is defined as 0 then it is indicated by an octal and 0x indicates it as hexadecimal number.  - There is function eval() that can be used to convert string into number but it is a bit slower and present many security risks like \_\_import\_\_('os').system("rm -rf$HOME") - use of this will delete the home directory of the system. |
| What is the function of negative index?  The sequences in python are indexed and it consists of the positive as well as negative numbers. The numbers that are positive uses ‘0’ that is uses as first index and ‘1’ as the second index and the process goes on like that. The index for the negative number starts from ‘-1’ that represents the last index in the sequence and ‘-2’ as the penultimate index and the sequence carries forward like the positive number. The negative index is used to remove any new-line spaces from the string and allow the string to except the last character that is given as S[:-1]. The negative index is also used to show the index to represent the string in correct order. |
| Write a program to check whether the object is of a class or its subclass.  There is a method which is built-in to show the instances of an object that consists of many classes by providing a tuple in a table instead of individual classes. The method is given as isinstance(obj,cls) and in more details given as:  isinstance(obj, (class1, class2, ...)) that is used to check about the object’s presence 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)).   It is not preferred to use the class instead user-defined classes are made that allow easy object-oriented style to define the behavior of the object’s class. These perform different thing that is based on the class. The function differs from one class to another class.  To find out the object of the particular class the following program is used:  def search(obj): if isinstance(obj, box): # This is the code that is given for the box and write the program in the object elif isinstance(obj, Document): # This is the code that searches the document and writes the values in it elif  obj.search() #This is the function used to search the object’s class. |
| Explain delegation in Python  Delegation is an object oriented technique (also called a design pattern). Let's say you have an object x and want to change the behaviour of just one of its methods. You can create a new class that provides a new implementation of the method you're interested in changing and delegates all other methods to the corresponding method of x. The example shows a class that captures the behavior of the file and converts data from lower to uppercase.  class upcase: def \_\_init\_\_(self, out): self.\_out = out def write(self, s): self.\_outfile.write(s.upper()) def \_\_getattr\_\_(self, name): return getattr(self.\_out, name)  The write() method that is used in the upcase class converts the string to the uppercase before calling another method. The delegation is being given using the self.\_\_outfile object. |
| What is the function of “self”?  “Self” is a variable that represent the instance of the object to itself. In most of the object oriented programming language, this is passed as to the methods as a hidden parameters that is defined by an object. But, in python it is declare it and pass it explicitly. It is the first argument that gets created in the instance of the class A and the parameters to the methods are passed automatically. It refers to separate instance of the variable for individual objects. This is the first argument that is used in the class instance and the “self” method is defined explicitly to all the methods that are used and present. The variables are referred as “self.xxx”. |
| How is “self” explicitly defined in a method?  “Self” is a reference variable and an instance attribute that is used instead of the local variable inside the class. The function or the variable of the self like self.x or self.meth() can be used in case the class is not known. There are no variables declared as local. It doesn’t have any syntax and it allow the reference to be passed explicity or call the method for the class that is in use. The use of writebaseclass.methodname(self, <argument list>) shows that the method of \_init\_() can be extended to the base class methods. This also solves the problem that is syntactic by using the assignment and the local variables. This tells a way to the interpreter the values that are to be used for the instance variables and local variables. The use of explicit self.var solves the problem mentioned above. |
| What is the use of join() for a string rather than list or tuple method?  The functions and the methods that are used for the functionality uses the string module. This string module is represented as by using the join function in it:  ", ".join(['1', '2', '4', '8', '16']) that results in "1, 2, 4, 8, 16"  The string variable that is used provide a fixed string literal to allow the names that are used to be bounded to the strings. join() is a string method that is used to provide a separator string to use the function over the sequence of the string and insert the function to an adjacent elements. The method uses any number of arguments that follow some rules that has to be put up for the sequence objects that the class defines for itself. The join is used for the string module that is used to join the string characters together as it is given in the program. The example is given as:  string.join(['1', '2', '4', '8', '16'], ", ") |
| What is the process of compilation and linking in python?  The compiling and linking allows the new extensions to be compiled properly without any error and the linking can be done only when it passes the compiled procedure. If the dynamic loading is used then it depends on the style that is being provided with the system. The python interpreter can be used to provide the dynamic loading of the configuration setup files and will rebuild the interpreter.  **The steps that is required in this as:** - Create a file with any name and in any lanugage that is supported by the compiler of your system. For example comp.c - Place this file in the Modules/ directory of the distribution which is getting used.  - Add a line in the file Setup.local that is present in the Modules/ directory.  - Run the file using spam comp.o - After successful run of this rebuild the interpreter by using the make command on the top-level directory.  - If the file is changed then run rebuildMakefile by using the command as ‘make Makefile’. |
| What is the procedure to extract values from the object used in python?  To extract the value it requires the object type to be defined and according to the object type only the values will be fetched.  **The values will be extracted as:** - If the object is a tuple then PyTuple\_Size() method is used that returns the length of the values and another method PyTuple\_GetItem() returns the data item that is stored at a specific index.  - If the object is a list then PyListSize() is having the same function that is defined for the tuple and PyList\_GetItem() that also return the data items at a specified index. - Strings uses PyString\_Size() to return the length of the value and PyString\_AsString() that return the pointer to its value.  - To check the type of the object and the extracted values use of methods like PyString\_Check(), PyTuple\_Check(), PyList\_Check(), etc are used. |
| What are the steps required to make a script executable on Unix?  **The steps that are required to make a script executable are to:** - First create a script file and write the code that has to be executed in it. - Make the file mode as executable by making the first line starts with #! this is the line that python interpreter reads.  - Set the permission for the file by using chmod +x file. The file uses the line that is the most important line to be used:  #!/usr/local/bin/python  - This explains the pathname that is given to the python interpreter and it is independent of the environment programs.  - Absolute pathname should be included so that the interpreter can interpret and execute the code accordingly. The sample code that is written:  #! /bin/sh # Write your code here exec python $0 ${1+"$@"} # Write the function that need to be included. |
| How does global value mutation used for thread-safety?  The global interpreter lock is used to allow the running of the thread one at a time. This is internal to the program only and used to distribute the functionality along all the virtual machines that are used. Python allows the switching between the threads to be performed by using the byte code instructions that are used to provide platform-independence. The sys.setcheckinterval() method is used that allow the switching to occur during the implementation of the program and the instruction. This provides the understanding in the field of accounting to use the byte code implementation that makes it portable to use. The atomicity can be provided such that the shared variables can be given as built-in data types. |
| Write a program to read and write the binary data using python?  The module that is used to write and read the binary data is known as struct. This module allows the functionality and with it many functionalities to be used that consists of the string class. This class contains the binary data that is in the form of numbers that gets converted in python objects for use and vice versa. The program can read or write the binary data is:  import struct f = open(file-name, "rb")  # This Open() method allows the file to get opened in binary mode to make it portable for # use.  s = f.read(8) x, y, z = struct.unpack(">hhl", s)  The ‘>” is used to show the format string that allows the string to be converted in big-endian data form. For homogenous list of data the array module can be used that will allow the data to be kept more organized fashion. |
| What is the process to run sub-process with pipes that connect both input and output?  The popen2() module is used to run the sub-process but due to some difficulty in processing like creation of deadlock that keep a process blocked that wait for the output from the child and child is waiting for the input. The dead lock occurs due to the fact that parent and child doesn’t have the synchronization and both are waiting to get the processor to provide the resources to one another. Use of popen3() method allow the reading of stdout and stderr to take place where the internal buffer increases and there is no read() takes place to share the resources. popen2() take care of the deadlock by providing the methods like wait() and waitpid() that finishes a process first and when a request comes it hands over the responsibility to the process that is waiting for the resources.  The program is used to show the process and run it.  import popen2 fromchild, tochild = popen2.popen2("command") tochild.write("input\n") tochild.flush() output = fromchild.readline() |
| What are the different ways to generate random numbers?  Random module is the standard module that is used to generate the random number.  The method is defined as:  import random random.random()  The statement random.random() method return the floating point number that is in the range of [0, 1). The function generates the random float numbers. The methods that are used with the random class are the bound methods of the hidden instances. The instances of the Random can be done to show the multi-threading programs that creates different instance of individual threads. The other random generators that are used in this are: - randrange(a, b): it chooses an integer and define the range in-between [a, b). It returns the elements by selecting it randomly from the range that is specified. It doesn’t build a range object.  - uniform(a, b): it chooses a floating point number that is defined in the range of [a,b).Iyt returns the floating point number - normalvariate(mean, sdev): it is used for the normal distribution where the mu is a mean and the sdev is a sigma that is used for standard deviation.  - The Random class that is used and instantiated creates an independent multiple random number generators. |
| Write a program to show the singleton pattern used in python.  Singleton patter is used to provide a mechanism that limits the number of instances that can be used by one class. It also allows the same object to be shared between many different parts of the code. This allows the global variables to be used as the actual data that is used is hidden by the singleton class interface. The singleton class interface can have only one public member and one class method Handle. Private constructors are not used to create an object that is used outside the class. The process waits for the static member function to create new instances and return the singleton object.  The code that is used to call the singleton object is:  Singleton& Singleton::Handle()  {    if( !psingle )     {        psingle = new Singleton;    }    return \*psingle; } |