PYTHON ASSIGNMENT 5

1) **What are docstrings in Python?**

Python docstrings are strings used right after the definition of a function, method, class, or module. They are used to document our code.

2) What is the purpose of is, not and in operators?

These operators help validate whether a given element is present in or is a member of the given sequence of data. This sequence of data can be a list, string or a tuple.

is Operator: It helps determine if a certain object belongs to a certain class/data type or if two operands are of the same object or not. We use this operator when in any condition we have to check for an object's identity. If the type/class of the operand on either side of the is operator is same, it evaluates to a true value, else to a false value.**in Operator:**It checks whether the value is present in the sequence of data or not. It evaluates to true value if the element is present in the sequence and to false value if the element is absent from the sequence. **not in Operator**:This operator checks for the absence of a value in a sequence. This is the exact opposite of the in operator. It evaluates to true when the element is not found or absent from the sequence and returns false when the element is found in the data sequence.

3) What is the usage of help() and dir() function in Python?

Help() and dir() both functions are accessible from the Python interpreter and used for viewing a consolidated dump of built-in functions.**help()** function: The help() function is used to display the documentation string and also facilitates you to see the help related to modules, keywords, attributes, etc.**dir()** function: The dir() function is used to display the defined symbols.

4) Whenever Python exits, why isn't all the memory de-allocated?

Objects referenced from the global namespaces of Python modules are not always deallocated when Python exits. This may happen if there are circular references. There are also certain bits of memory that are allocated by the C library that are impossible to free (e.g. a tool like Purify will complain about these). Python is, however, aggressive about cleaning up memory on exit and does try to destroy every single object.

- Whenever Python exits, especially those Python modules which are having circular references to other objects or the objects that are referenced from the global namespaces are not always de-allocated or freed.
- It is impossible to de-allocate those portions of memory that are reserved by the C library.
- On exit, because of having its own efficient clean up mechanism, Python would try to de-allocate/destroy every other object

They are different types of inheritance supported by Python:

- **Single Inheritance** - where a derived class acquires the members of a single super class.
- **Multi-level inheritance** - a derived class d1 in inherited from base class base1, and d2 are inherited from base2.
- **Hierarchical inheritance** - from one base class you can inherit any number of child classes
- **Multiple inheritance** - fa derived class is inherited from more than one base class.

5) What is a dictionary in python?

The built-in datatypes in Python is called dictionary. It defines one-to-one relationship between keys and values. Dictionaries contain pair of keys and their corresponding values. Dictionaries are indexed by keys.Use {} curly brackets to construct the dictionary, and [] square brackets to index it. Separate the key and value with colons : and with commas , between each pair. Keys must be quoted As with lists we can print out the dictionary by printing the reference to it. A dictionary maps a set of objects (keys) to another set of objects (values). A Python dictionary is a mapping of unique keys to values.Dictionaries are mutable, which means they can be changed. The values that the keys point to can be any Python value. Dictionaries are unordered, so the order that the keys are added doesn't necessarily reflect what order they may be reported back.

6) How can files be deleted in Python?

Removing the files or a single file from a particular directory when it is no longer required is the basic concept of deleting a file. Now to remove a file, you have three methods. Using one of the modules:

Using the os module in python

The os module allows you to use the operating system dependent functionalities.To use the os module to delete a file, we need to first import it, and then use the remove() function provided by the module to delete the file. It takes the file path as a parameter.You can not just delete a file, you can also delete a directory using the os module. On the other hand, if you wish to delete or clear a directory altogether, you can do it with the help of the rmdir() function of the os module. Note that the directory must be empty for this to work.

Using the shutil module in python

The shutil module is a high-level file operation module. You can perform functions like copying and removal on files and collections of files.This module can also be used like the os module to delete files or a directory. But here, it is not a necessity for the directory to be empty. If you delete a directory using shutil, you will be able to delete all the contents inside of it as well (files and subdirectories).The function that we use of the shutil module is rmtree().You cannot delete a single file with the shutil.rmtree() function.

Using the Pathlib Module in Python

Pathlib module is a very useful module for deleting / removing files.The pathlib module has many similarities with the os module, two of them being the remove and rmdir methods.Now, when working with this module, you must first create a Path object. When an instance of the Path class is created, a WindowsPath or PosixPath will be returned according to the machine you’re working on. For Windows OS, a WindowsPath object will be returned, and for non-windows OS like linux, PosixPath will be returned.