# Assignment 5

1) What are docstrings in Python?

Python documentation strings (or docstrings) provide a convenient way of associating documentation with Python modules, functions, classes, and methods.

It’s specified in source code that is used, like a comment, to document a specific segment of code. Unlike conventional source code comments, the docstring should describe what the function does, not how.

Docstring look like:

The doc string line should begin with a capital letter and end with a period.

The first line should be a short description.

If there are more lines in the documentation string, the second line should be blank, visually separating the summary from the rest of the description.

The following lines should be one or more paragraphs describing the object’s calling conventions, its side effects, etc.

Declaring Docstrings: The docstrings are declared using ”’triple single quotes”’ or “””triple double quotes””” just below the class, method or function declaration. All functions should have a docstring.

Accessing Docstrings: The docstrings can be accessed using the __doc__ method of the object or using the help function.

The below examples demonstrates how to declare and access a docstring.

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

Not Operator:- It is logical operators and its meaning is true if operand is false.

Is Operator:- It is identity operators and its meaning is true if the operands are identical.

In Operator:- It is membership operators and its meaning is true if value/variable is found in the sequence i.e. the ‘in’ operator is used to check if a value exists in a sequence or not and it evaluates to true if it finds a variable in the specified sequence otherwise false.

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

help() function help, when it is provided with a program-name or a module-name or a function-name as an argument, it displays the documentation of the argument as help.

The dir() function returns a list of all the attributes within in an object - these could be either data values or methods, and the list includes private and magic methods.

4) Whenever Python exits, why isn’t all the memory deallocated?

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 not possible 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 every object.

5) What is a dictionary in Python?

Dictionaries are used to store data values in key:value pairs.

A dictionary is a collection which is ordered*, changeable and do not allow duplicates.

6) How can files be deleted in Python?

To delete a file, you must import the OS module, and run its os.remove() function:

To avoid getting an error, you might want to check if the file exists before you try to delete it

To delete an entire folder, use the os.rmdir() method