1. Discuss the application of data types: lists and tuples and strings.

I think the most significant thing to point out is that strings, unlike Java, are not immutable. They are not an array of characters; therefore, to access individual characters… a user would utilize their substring behavior (“Python Strings”, 2016).

Strings

Example:

var1 = “happy”

var2 = “day”

print var1[1:4]

RESULT:

app

Also, there is raw strings.

Example:

print r’\\anyone

RESULT:

[\\anyone](file:///\\anyone)

Triple quotes allow for both, special characters and explicit typing within a long string.

Example:

“””This is a nice \n and shiny

day in the Seattle area.”””

RESULT:

This is a nice

and shiny

day in the Seattle area.

Lists

Lists can be sliced, concatenated and are defined within square brackets with any datatype in any order (“Python Lists”, 2016).

Example:

list1 = [1, “word”, 10, [1,2,3]];

Values in a list are accessed by their index which start at 0 and increase by 1 until the nth element is reached.

Lists can be updated and appended to and again, the data type does not require specificity.

Tuples

Tuples are sequences/lists that optionally utilize parenthesis instead of curly braces during their instantiation process and they are immutable (“Python Tuples”, 2016).

An interesting tid bit about tuples is that you must insert a comma after the first value even if you don’t include any other values in the tuple.

Example:

tup1 = (1,);

Tuples also use splicing and do not act like substrings.

Example:

tup2 = (1,2,3,4,5);

print tup2[1:5]

RESULT:

1,2,3,4,5

Note: the tuple printed all-inclusive indexes and did not reduce the second argument by one for considerations of length/size like a substring would.

Lastly, tuples must be deleted in their entirety. They cannot be partially deleted or altered in anyway as they are immutable objects.

Example:

tup4 = (1,3);

del tup4;

print tup4;

RESULT:

error…. tup4 is not defined

1. What is a class, a method, a self? What is the relationship between classes and objects? Use examples to demonstrate. What is meant by the methods associated with a class/object? How do they relate to object-oriented programming?

A class is a “user-defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members (class variables and instance variables) and methods, accessed via dot notation (“Python Object Oriented”, 2016).

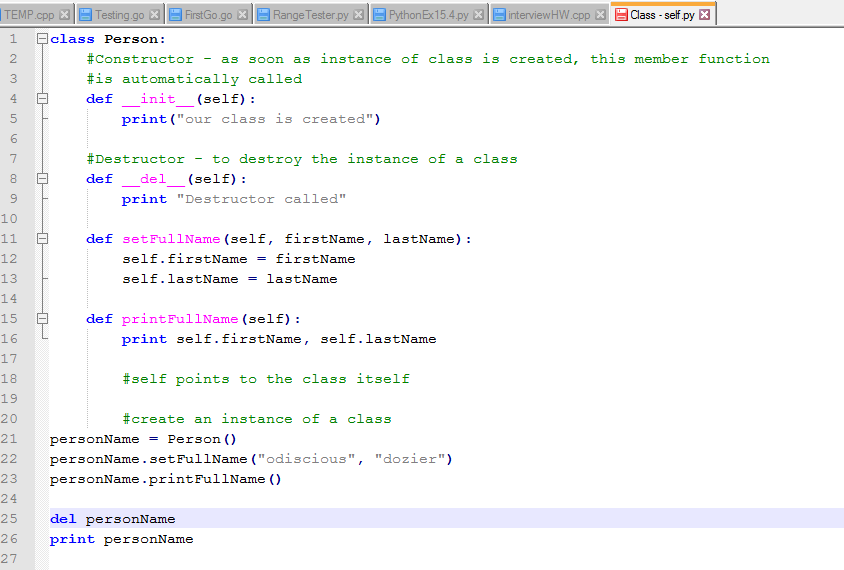
A method is a “special kind of function that is defined in a class definition” (“Python Object Oriented”, 2016). It can take zero or many arguments and has one return type.

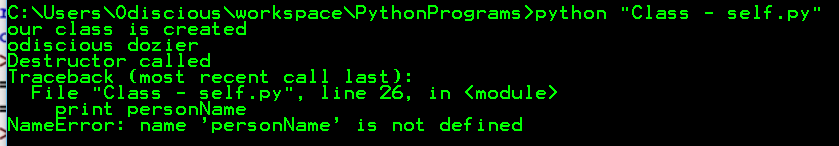
A self is a required parameter for a function. Like the ‘this’ keyword in Java, ‘self’ refers or points to the specific object the user is working with. It’s necessary for setting backing fields in the class (“Python Tutorial”, 2016).

See example given in question 3…

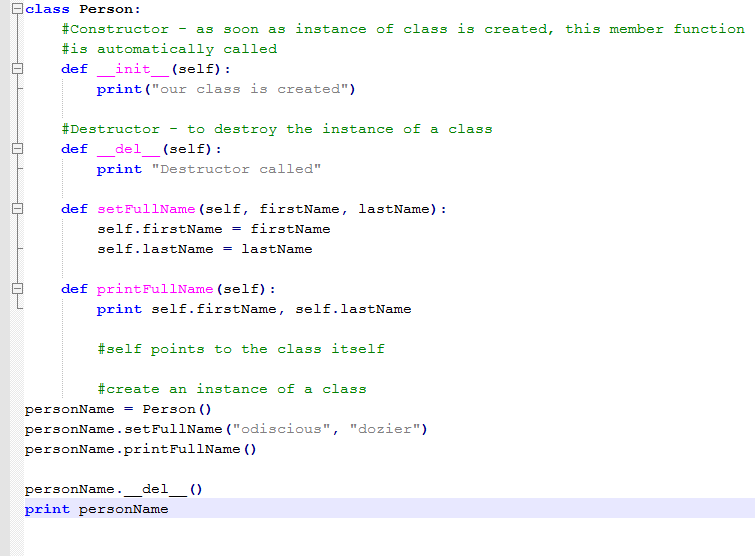
1. How does one define a new class and its methods? Which methods are inherited if no new methods are redefined?

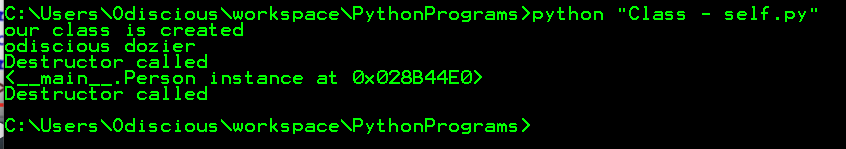
Using ‘del’ keyword gives expected results….





Using out \_\_del\_\_() function that we built gives expected results but with slight differences





REFERENCES

Python Lists. (2016). Retrieved March 19, 2016, from <http://www.tutorialspoint.com/python/python_lists.htm>

Python Object Oriented. (2016). Retrieved March 19, 2016, from <http://www.tutorialspoint.com/python/python_classes_objects.htm>

Python Strings. (2016). Retrieved March 19, 2016, from <http://www.tutorialspoint.com/python/python_strings.htm>

Python Tuples. (2016). Retrieved March 19, 2016, from <http://www.tutorialspoint.com/python/python_tuples.htm>

Python Tutorial for Beginners 15 - Classes and Self. (2016). Retrieved March 19, 2016, from <https://www.youtube.com/watch?v=cJq_kuAKPCs>