Object oriented language:

Python is an object oriented language. This means that technically every variable is an object. An object is an instance of a class. We can understand objects and classes by referring to basic C. In C if we want to make an integer we instantiate a variable of type int. So we can imagine that int is a class and if I call my variable x then x is an object of the int class. So the command 'int x' in C makes an object called x of type int. Similarly in C we can do 'char ch' which means make a variable (object) called ch of type char.

We can go beyond a int or char. Suppose I have a class that contains an int called x and a char called ch.

Class Cl: {
    int x
    char ch
}

If I make an object of type class Cl called c then I can access the variables x and ch inside the class. So if I type c.x I am accessing the int in the class. If I am typing c.ch I am accessing the char in the class.

class Cl: {
    int x
    int y
    add(): return x+y;
}

Cl c;
c.x=5
C.y=6
print(c.add())

The function add() in the class is called a method. All functions in a class are called methods and invoked by object.method().

-----------------------------

In Python we have methods and functions. Functions are generic and can be applied to different data types whereas methods are specific to class types.

-----------------------------

Two dimensional matrices:
In Python we represent them with list of lists. So for example the matrix

\[
\begin{array}{ccc}
3 & 4 & 10 \\
2 & 20 & 19
\end{array}
\]

is given by \( M = \left[ [3, 4, 10], [2, 20, 19] \right] \)

05/17/2017

A list is an ordered set of items whereas a dictionary is a set of (key,value) pairs. The (key,value) are unordered and we can only look up a value if we have the key.