جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلوم
قسم الانظمة الطبية الذكية

Lecture: (6)

Attributes and Methods

Subject: Object oriented programming I

Class: Second

Dr. Maytham N. Meqdad

Accessing Attributes and Methods in Python

Attributes of a class are function objects that define corresponding methods of its instances. They are used to implement access controls of the classes.
Attributes of a class can also be accessed using the following built-in methods and functions :

1. **getattr() –** This function is used to access the attribute of object.
2. **hasattr() –** This function is used to check if an attribute exist or not.
3. **setattr() –** This function is used to set an attribute. If the attribute does not exist, then it would be created.
4. **delattr() –** This function is used to delete an attribute. If you are accessing the attribute after deleting it raises error “class has no attribute”.

The following methods are explained with the example given below:

```
# Python code for accessing attributes of
class
class emp:
    name='Harsh'
    salary='25000'
    def show(self):
        print (self.name)
        print (self.salary)
e1 = emp()
# Use getattr instead of e1.name
print (getattr(e1,'name'))
```

```
# returns true if object has attribute
print (hasattr(e1,'name'))

# sets an attribute
setattr(e1,'height',152)

# returns the value of attribute name
height
print (getattr(e1,'height'))

# delete the attribute
delattr(emp,'salary')
```

Output :

Harsh
True
152

Static methods : A static method is a method[member function] that don't use argument self at all. To declare a static method, proceed it with the statement "@staticmethod".

```
# Python code for accessing methods using static
method
class test:
    @staticmethod
    def square(x):
        test.result = x*x

# object 1 for class
t1=test()

# object 2 for class
t2 = test()
t1.square(2)

# printing result for square(2)
print (t1.result)
t2.square(3)

# printing result for square(3)
print (t2.result)

# printing the last value of result as we
declared the method static
print (t1.result)
```

Output :

```
4
9
9
```

Accessing attributes and methods of one class in another class

Accessing attributes and methods of one class in another class is done by passing the object of one class to another.
Explained with the example given below :

```
# Python code for Accessing attributes and methods
# of one class in another class

class ClassA():
    def __init__(self):
        self.var1 = 1
        self.var2 = 2

    def methodA(self):
        self.var1 = self.var1 + self.var2
        return self.var1

class ClassB(ClassA):
    def __init__(self, class_a):
        self.var1 = class_a.var1
        self.var2 = class_a.var2

object1 = ClassA()
# updates the value of var1
summ = object1.methodA()
```

```
# return the value of var1
print (summ)

# passes object of classA
object2 = ClassB(object1)

# return the values carried by var1,var2
print( object2.var1)
print (object2.var2)
```

Output :

```
3
3
2
```