

# Python Librarie

**Pimentech**

<http://www.pimentech.net>

December 2, 2009

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Types</b>	<b>1</b>
2.1	Object . . . . .	1
2.2	Map . . . . .	1
2.3	Set . . . . .	2
2.4	Graph . . . . .	3

## 1 Introduction

## 2 Types

### 2.1 Object

Creating an object instance of top class Object :

```
>>> from object import *
>>> object1 = Object('object1')
```

XML representation :

```
>>> object1
<object name='object1'/>
```

`__str__` returns `object.name` :

```
>>> str(object1)
'object1'
```

The hash code of the instance is equal to that of its name :

```
>>> hash(object1) == hash('object1')
1
```

## 2.2 Map

Inherits from Object.

```
>>> from map import *
>>> map = Map('a map')
```

Inserting :

```
>>> object2 = Object('object2')
```

```
>>> map['object1_key']=object1
>>> map['object2_key']=object2
```

```
>>> map['object1_key']
<object name='object1' />
```

XML representation :

```
>>> map
<map name='a map'>
<pair>
<key>'object2_key'</key>
<value><object name='object2' /></value>
</pair>
<pair>
<key>'object1_key'</key>
<value><object name='object1' /></value>
</pair>
</map>
```

```
>>> map.keys()
['object2_key', 'object1_key']
```

```
>>> map.values()
[<object name='object2' />, <object name='object1' />]
```

```
>>> map.items()
[('object2_key', <object name='object2' />), ('object1_key', <object name='object1' />)]
```

## 2.3 Set

Inherits from Map.

```
>>> from set import *
>>> set = Set('a set')
```

**Inserting an object :**

```
>>> set.insert(object1)
<object name='object1' />
```

```
>>> set.insert(object2)
<object name='object2' />
```

**XML representation :**

```
>>> set
<set name='a set'>
<element><object name='object2' /></element>
<element><object name='object1' /></element>
</set>
```

```
>>> set['object1']
<object name='object1' />
```

```
>>> set[object1]
<object name='object1' />
```

```
>>> set.values()
[<object name='object2' />, <object name='object1' />]
```

## 2.4 Graph

Inherits from Set. A graph is a set of vertices.

```
>>> from graph import *
>>> graph = Graph('a graph')
```

**inserting a vertex of content object1 :**

```
>>> graph.insert(object1)
<vertex name='object1'>
<pair>
<value><object name='object1' /></value>
<edges>
</edges>
</pair>
</vertex>
```

### XML representation of a graph :

```
>>> graph
<set name='a graph'>
<element><vertex name='object1'>
<pair>
<value><object name='object1' /></value>
<edges>
</edges>
</pair>
</vertex></element>
</set>
```

### XML representation of a vertex :

```
>>> graph['object1']
<vertex name='object1'>
<pair>
<value><object name='object1' /></value>
<edges>
</edges>
</pair>
</vertex>
```

### getting the content of a vertex :

```
>>> graph['object1'].object
<object name='object1' />
```

### inserting an edge :

```
>>> graph.insert_edge(object1, 'a label', object2)
```

### XML representation of a vertex with outgoing edges :

```
>>> graph['object1']
<vertex name='object1'>
<pair>
<value><object name='object1' /></value>
<edges>
<pair>
<label>'a label'</label>
<vertices>
<vertex name='object2' />
</vertices>
</pair>
</edges>
```

```
</pair>  
</vertex>
```

getting outgoing labels from object1 :

```
>>> v = graph['object1']  
>>> v.keys()  
['a label']
```

set of vertices reachable from v by edges labeled by 'a label' :

```
>>> v['a label']  
<set name='Set'>  
<element><vertex name='object2'>  
<pair>  
<value><object name='object2' /></value>  
<edges>  
</edges>  
</pair>  
</vertex></element>  
</set>
```