

BST 234: Lab - 3

Divy Kangeyan

June 17, 2018

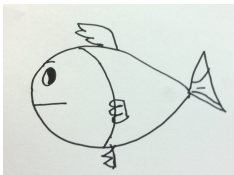
Classes in Python

- Classes define objects and their associated methods
- class object is defined as `class <name>:`

Class Objects

- Class objects support two operations
 - 1 Attribute references
 - 2 Instantiation
- We need to create an instance of the class and then we may reference its various attributes

Class Objects



- Class
- Class objects support two operations
 - 1 Attribute references : has gills, has scales



- 2 Instantiation:

Class Instantiation

- We need to create an instance of the class and then we may reference its various attributes

Class Instantiation

```
class MaxHeap(object):  
    #construct - empty array  
    def __init__(self, data=[]):  
        self.data=data
```

Calling the class

```
x = MaxHeap()  
data = x.data
```

Attribute Reference

- Given an instance, we may reference attributes (assign, retrieve, delete)
- Two types of attributes:
 - 1 Data attributes
 - 2 Methods

Attribute reference

```
class MaxHeap(object):  
    #construct - empty array  
    def __init__(self, data=[]):  
        self.data=data  
  
    def isEmpty(self):  
        return(len(self.data)==0)
```


Class Vs. Instance variables

- We saw how to reference methods on the previous slides
- Two types of variables exist
 - 1 Class variable(e.g. has gills, has scales)
 - 2 Instance variable(e.g. color of a muscle tissue ranges from pink to dark red, streamlined body)
- Class variables are shared by all instances of the class
- Instance variable are unique to each call

Summary

- Classes provide a structure for a process with associated methods and objects
- Attribute references and instantiation are operations supported by a class object
- Data attributes and methods are two types of attributes
- There are two types of variables: class variable (shared by all instance of a class) and instance variable (unique to each instantiation)