Python Review
Part II

Recap

- import
- def
- return
- Booleans, conditionals, branching
- Loops : for, while, range
- Files : open, close, readlines, split

Classes

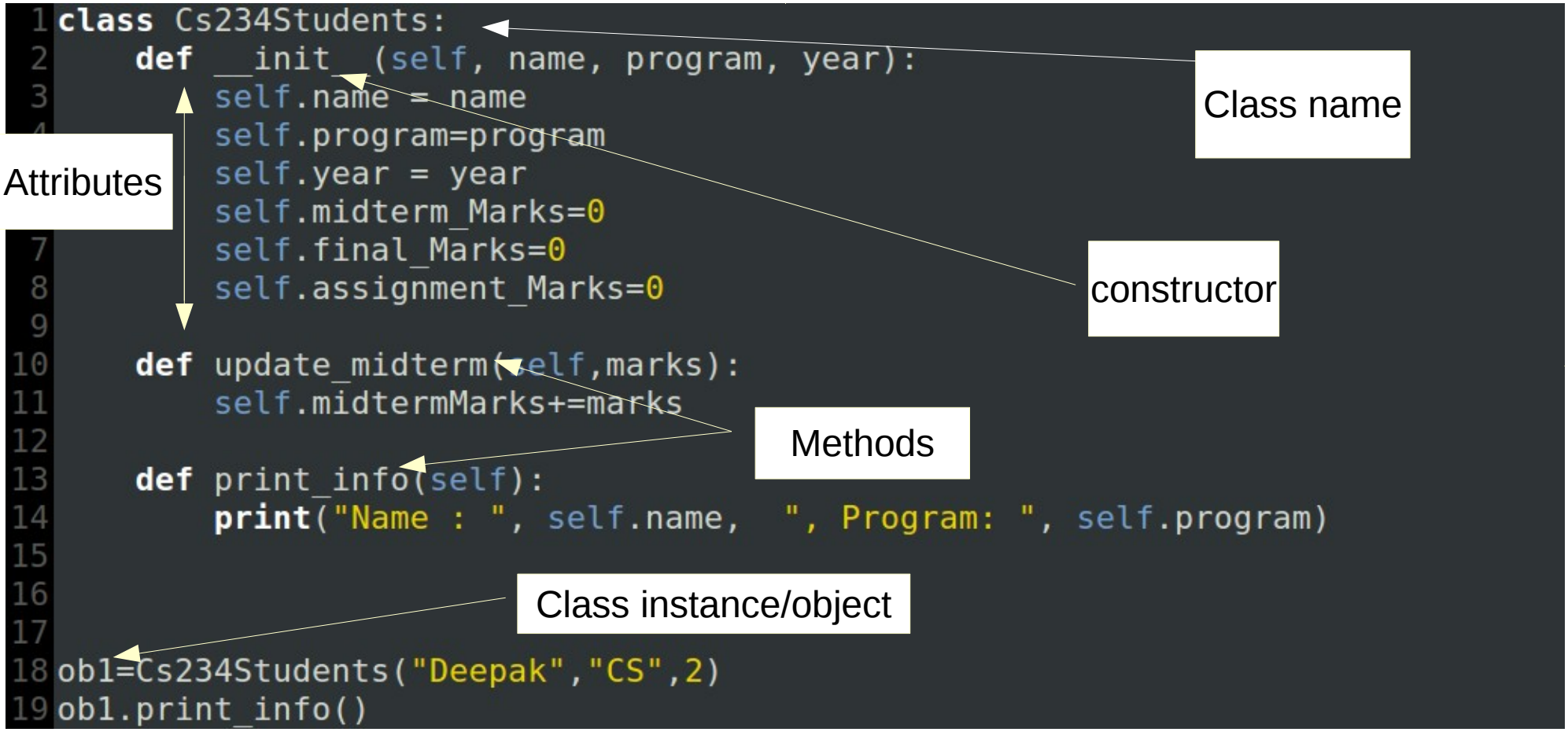
- Classes enable us to define blueprint for a custom data type
- So, far you have seen datatypes such as integers, decimals, strings, e.t.c.
- A class can have all of these wrapped under one name.

Example

Class CS234Students

- Name
- Program
- Year
- Midterm Marks
- Final Marks
- Assignment Marks

Classes in Python



Key points

- __init__ : is called whenever a new class object is created. It is optional to have it in the program

- Important to have (self) as an argruments to all methods in Python classes

- All the class methods and attributes are accessed by the dot ('.') operator.

Recursion


We must go one level deeper
And then what?
We must go one level deeper
And then what?
RECURSION

Formally

- Recursion is the process of defining a problem (or the solution to a problem) in terms of (a simpler version of) itself.
- Recursion is achived when a function calls itself.
- It is important to have a base case, so that the recursion terminates at some point.

Factorial

- To calculate n!
- Break down into simpler sub problems.
- We know n!=n*(n-1)!.
- Thus, if we knew (n-1)!, we could calculate n!
- Call the function recursivly untill n becomes 1, at which point return 1.

Lists

- Most versatile datatype in Python.
- List of comma-separated values (items) between square brackets.
- Items in the list can be of different types.
- ```
  list1 = ['physics', 'chemistry', 1997, 2000];
  list2 = [1, 2, 3, 4, 5, 6, 7 ];

  print "list1[0]: ", list1[0]
  print "list2[1:5]: ", list2[1:5]
  ```
  ```

- Update the list using the assignment operator list[5]='Hello world'
- list.append() and '+' method adds elements to the end of the.
- list1.extend(list2) merges 2 lists.
-

```
x = [1, 2, 3]
x.append([4, 5])
print (x)
```

gives you: `[1, 2, 3, [4, 5]]`

```
x = [1, 2, 3]
x.extend([4, 5])
print (x)
```

gives you: `[1, 2, 3, 4, 5]`

Stackoverflow.com

Tuples

- Tuples are like lists except that once created, they cannot be changed.
- tup1 = (12, 34.56);
- tup2 = ('abc', 'xyz');

  # Following action is not valid for tuples

  # tup1[0] = 100;
- But its possible it to to add 2 tuples.

  tup3 = tup1 + tup2;

  print tup3

Dictionary

- A dictionary in Python is a key value pair.
- Each key is separated from its value by a colon (:), the items are separated by commas, and the whole thing is enclosed in curly braces.
- An empty dictionary without any items is written with just two curly braces, like this: {}.
- Keys are unique within a dictionary while values may not be. The values of a dictionary can be of any type, but the keys must be of an immutable data type such as strings, numbers, or tuples.

Practice Question 1

- Print all n-digit strictly increasing numbers

  Input: n = 2

  Output:

  01 02 03 04 05 06 07 08 09 12 13 14 15 16 17 18 19 23 24 25 26 27 28

  29 34 35 36 37 38 39 45 46 47 48 49 56 57 58 59 67 68 69 78 79 89

Practice question 2

- Generate all possible combinations of a given number
- Find x^y in O(log(n)) time.

- Very useful when we want a mapping between 2 entities for which we want fast O(1) retrieval.

```
In [1]: dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}

In [2]:

In [2]: print "dict['Name']: ", dict['Name']
dict['Name']:  Zara

In [3]: print "dict['Age']: ", dict['Age']
dict['Age']:  7
```