Classes and Iterators

October 10th, 2017 | Sandra Diaz
Introduction to Classes

• In object oriented programming, classes represent objects with attributes (variables) and possible actions they can perform or you can perform on them (functions).

• Classes encapsulate functionality in well defined units with a purpose, scope and well defined interaction capabilities (via its functions)
Introduction to Classes

- A class is defined using the `class` keyword, and the class definition usually contains a number of class method definitions (a function in a class).
- Each class method must have an argument `self` as its first argument. This object is a self-reference.
- Special class methods:
  - `__init__`
  - `__str__`
  - [http://docs.python.org/2/reference/datamodel.html#special-method-names](http://docs.python.org/2/reference/datamodel.html#special-method-names)
Introduction to iterators

• Iterable objects are objects which can be accessed element-wise using an iterator
• They implement the __iter__ method
• Examples of iterable objects: lists, strings, dictionaries, files, etc
• for, while use iterators to access elements in the iterable objects
Introduction to iterators

• To obtain an iterator from an object, one can use the `iter` function

• The `next` method is used to access the next element in the object
References

(1) Based on the work by J.R. Johansson http://jrjohansson.github.io