Programming Project 0: Geometry
Object-oriented programming basics

Write a class to store the geometry, masses, and nuclear charges of a molecule. Afterwards, write a small program to test your class. Make sure each class method is called at least once.

Extra Files

file name	description
`molecule.xyz`	sample `.xyz` file for testing out your program
`masses.py`	provides functions `get_mass` and `get_charge` which return the mass and charge of an atom, e.g.:

```
>>> import masses
>>> masses.get_mass("O")
15.99491461956
>>> masses.get_charge("O")
8
```

Class description

Member variables.

name	type	description
`units`	`str`	either `"Angstrom"` or `"Bohr"`, specifying the distance units used for spatial coordinates
`natom`	`int`	the number of atoms
`labels`	`list` of `strs`	a list of uppercase atomic symbols, following the order of the `.xyz` file
`masses`	`list` of `floats`	a list of atomic masses, following the order of the `.xyz` file
`charges`	`list` of `ints`	a list of atomic charges, following the order of the `.xyz` file
`geom`	`numpy.matrix`	an `natom` $\times$ 3 matrix containing the Cartesian coordinates of each atom, following the order of the `.xyz` file

Methods.

method	description
Constructor (`__init__`)	takes `str` contents of an `.xyz` file as input; initializes all member variables and fills them with their correct values
`to_bohr`	converts the distance units to Bohr, changing member variables `units` and `geom` if necessary
`to_angstrom`	converts the distance units to Angstroms, changing member variables `units` and `geom` if necessary
`copy`	returns `Molecule` object, which is a fresh copy of `self`