Exercise 05: Object-Oriented Programming in Python

Exercise 05.1

1. Create a class hierarchy of geometrical objects. It should contain a class for a general geometrical object, for a Cuboid (with given lengths a, b, c), an Ellipsoid (with r1, r2, r3), a Cube and a Sphere. Each object should have the following features:

   - attributes: `middlepoint` (of type `numpy.array` with shape `(3,)`), `color` (of type `str`), `density` (of type `float` or `int`, > 0), `temperature` (of type `float` or `int`, > 0)
   - a method to check if all attributes match the above specifications
   - a method to calculate the volume
   - a method to calculate the mass
   - a method to move in a given direction (3d vector specified as as `numpy.ndarray`)

   Each method should be implemented as high as possible in the class hierarchy.

2. Create one instance for each of these classes. Thereby, invent some reasonable values for the attributes.

3. Execute and ensure a meanigful result for each of the methods for your instance of `Cuboid` and `Sphere`.

4. Create 10 `Cube`-instances located on random positions in the space and store them in a list.

5. Add a method to calculate the distance between the middlepoints of two objects to each class (Where ist put best?). Test it by calculating the distance between a `Cube` at (3, 0, 0) and a `Sphere` at (0, 4, 0).