Worksheet: 3d `Vector` Class

Below is our previous `Vector` class code.

***Code Block 1**: The `Vector` class*

```
1  public class Vector {
2     public double x;
3     public double y;
4     public Vector(double x, double y) {
5        this.x = x;
6        this.y = y;
7     }
8     public void add(Vector v) {
9        this.x += v.x;
10       this.y += v.y;
11    }
12 }
```

Update the `Vector` class so that it may be used to store either two-dimensional vectors or three-dimensional vectors. Accomplish this by updating in the following ways:

- Add a third instance field, named `z`, of type `double` to hold the third dimension of the vector.
- Update the `add` method to account for the additional `z` dimension.
- ***Overload*** the `add` method with a method that takes in three double parameters (use the names `x`, `y`, and `z`), and add these values to the vector.
- ***Overload*** the constructor with a second constructor that takes three parameters of type `double` to initialize the three instance fields. (Leave the original constructor so that the user of the class need not initialize the third dimension, but rather leave the value as zero).
- Add a method named `length` that returns the calculated length of the vector as a type `double`. (The length of a vector, `d`, is given by the formula: $d=\sqrt{x^2+y^2+z^2}$).

A UML class diagram of the original `Vector` class and the updated `Vector` class are given here:

Original Vector Class

Vector
+x: real +y: real
+Vector(x: real, y:real) +add(v: Vector)

Updated Vector Class

Vector
+x: real +y: real +z: real
+Vector(x: real, y:real) +Vector(x: real, y:real, z:real) +add(v: Vector) +add(x: real, y:real, z:real) +length()

Write the Java code for the updated `Vector` class on the reverse of this page.