6. Classes: methods and the static keyword

Note: class member methods should NOT be `static` in this module
Resources:

Classes: methods

Classes: the static keyword

Problems

1. Create a class named "Vector" It should have the following attributes: x, y and z.
   (a) Create a method in the class that prints the values of x, y and z.
   (b) Create a method in the class that calculates the length of the vector.
   (c) Create a method in the class that normalizes the attributes of the vector. It should use the previous method in the calculation.
   (d) Create a method in the class that adds two vectors together. It should return the sum of the two vectors as a new vector. This method should NOT be `static` and it should only have ONE parameter.
2. What does the code in the following snippet do? And why does it lead to this behavior?

```
1  public class Person {
2
3      public int age;
4
5      public void setAge(int newAge) {
6          age = newAge;
7      }
8
9  }
10
11 public class Main {
12
13     public static void main(String[] args) {
14         Person.setAge(3);
15     }
16
17 }
```

3. What does the code in the following snippet do? And why does it lead to this behavior?

```
public class Person {

    public int personCount = 0;

    public static void registerPerson() {
        personCount++;
    }

}

public class Main {

    public static void main(String[] args) {
        Person.registerPerson();
    }

}
```

4. What does the code in the following snippet do? And why does it lead to this behavior?

```
public class Car {
    public static int carCount = 0;

    public static Car produce() {
        carCount++;
        Car car = new Car();
        return car;
    }
}

public class Main {

    public static void main(String[] args) {
        Car.produce();
        Car.produce();
        Car.produce();
    }

}
```

5. What are the effects of the static keyword?