**Description tasks**

1. Describe the mechanisms used in inheritance

One of the main mechanisms behind the feature of inheritance is that it works a lot like a family tree, why the classes also are related to each other in that way. Therefore, the children classes are what their parent is, but at the same have the ability to be different from each other, why they also should be treated in that way. That is another feature of inheritance called polymorphism.

5) Describe the program

Firstly, a main method is created. In that runs a for loop incrementing i by one as long as it’s less than 10. During the loop’s runtime, an integer variable named *random* is being set equal to the getRandomInteger() method and printed. Right after the loop, the message “Random integers generated” is printed. Then the generateRandomInteger() method is defined. In there, a new Random object with the name randomGenerator is being instantiated. Then randomGenerator is returned with a bound of 10.

7) Describe the *static* keyword

The *static* keyword represents that a member is belonging to itself only rather than to an instance of the type. This leads to only one instance of that static member is created and shared across all the instances of that class.

Example (here used on a field/an attribute):

public class Car  
 {  
 private String name;  
 private String engine;  
  
 static public int *numberOfCars*;  
  
 public Car(String name, String engine)  
 {  
 this.name = name;  
 this.engine = engine;  
 *numberOfCars*++;  
 }  
 }

In this Car class example, there are three attributes (instance variables). Whenever a new object is being instantiated from this class (or blueprint), it’ll have a distinct copy of attributes/instance variables. But the interesting part here is the variable counting the number of cars, which will increment by one, each time a blueprint is instantiated, since it has the keyword *static* in front of it.