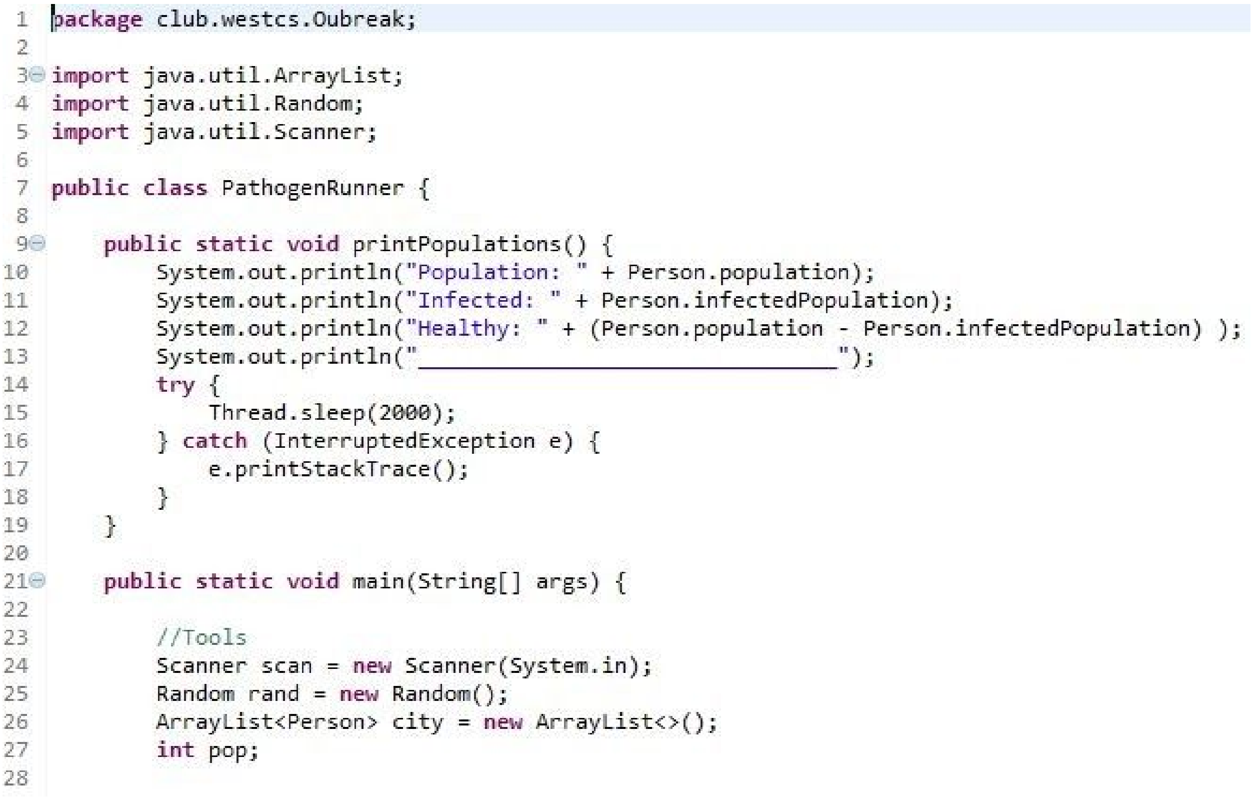
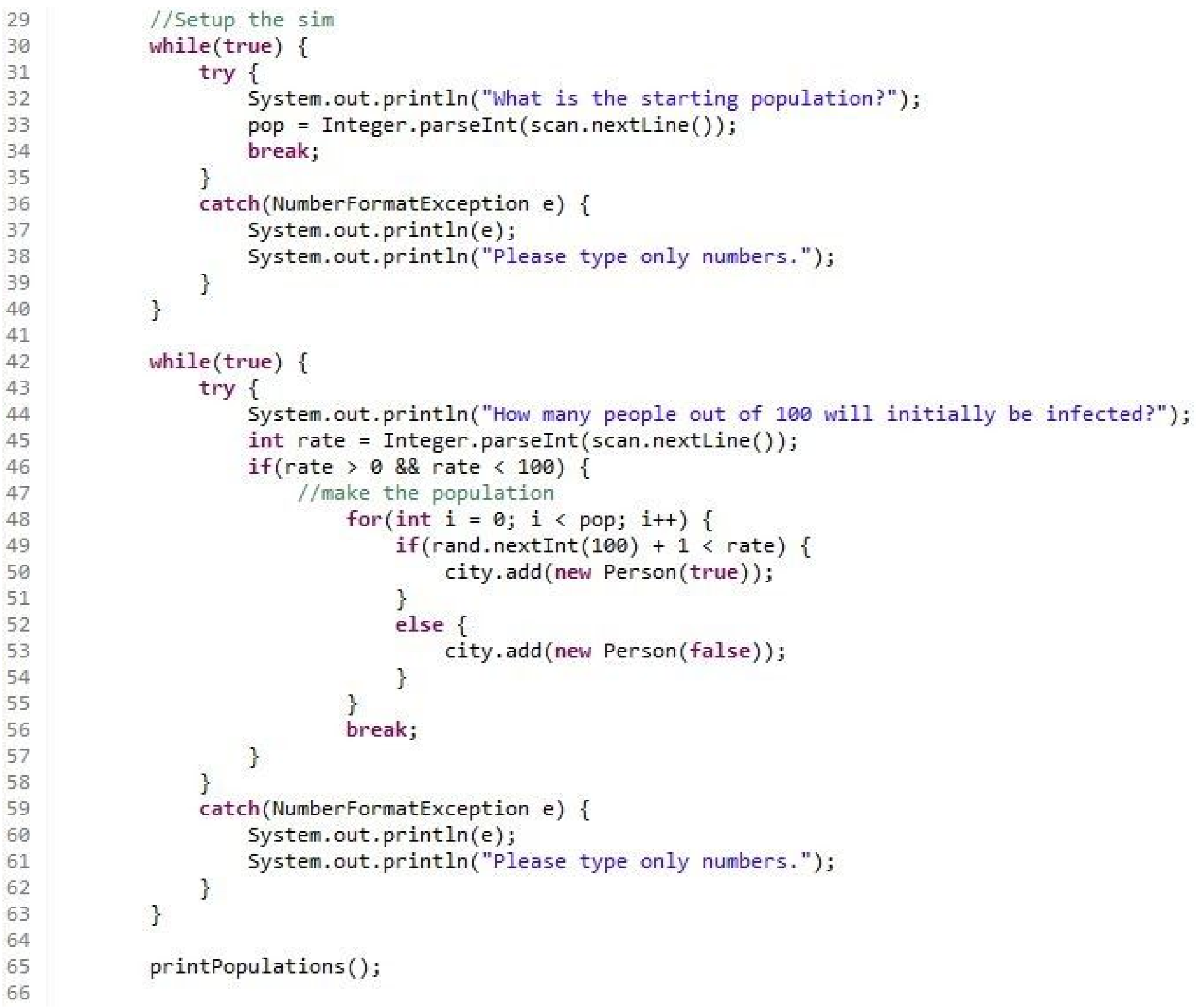
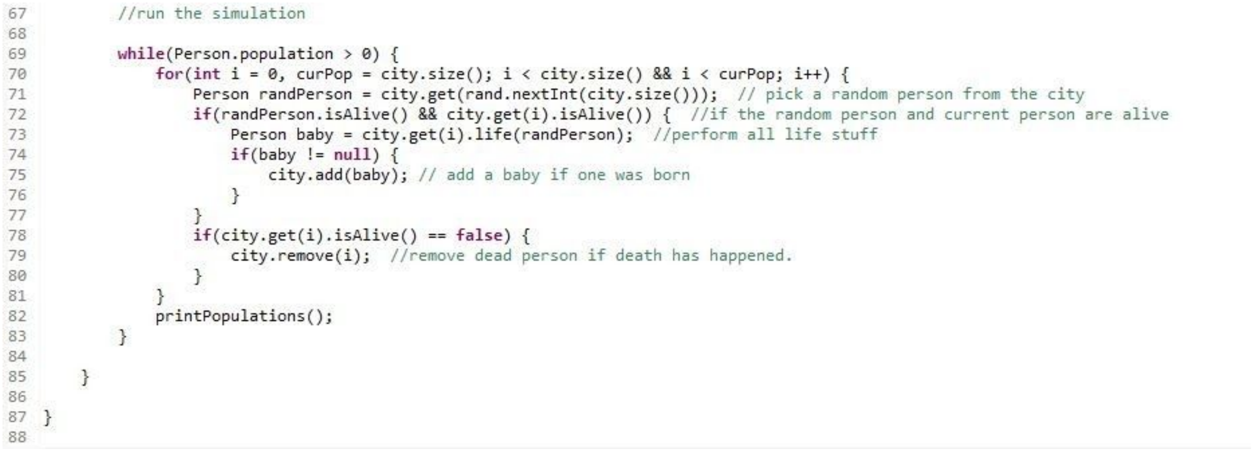


```java
package club.westcs.Oubreak;

import java.util.ArrayList;
import java.util.Random;
import java.util.Scanner;

public class PathogenRunner {

    public static void printPopulations() {
        System.out.println("Population: " + Person.population);
        System.out.println("Infected: " + Person.infectedPopulation);
        System.out.println("Healthy: " + (Person.population - Person.infectedPopulation) );
        System.out.println("________________________________");
        try {
            Thread.sleep(2000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {

        //Tools
        Scanner scan = new Scanner(System.in);
        Random rand = new Random();
        ArrayList<Person> city = new ArrayList<>();
        int pop;

```



```java
        //Setup the sim
        while(true) {
            try {
                System.out.println("What is the starting population?");
                pop = Integer.parseInt(scan.nextLine());
                break;
            }
            catch(NumberFormatException e) {
                System.out.println(e);
                System.out.println("Please type only numbers.");
            }
        }

        while(true) {
            try {
                System.out.println("How many people out of 100 will initially be infected?");
                int rate = Integer.parseInt(scan.nextLine());
                if(rate > 0 && rate < 100) {
                    //make the population
                        for(int i = 0; i < pop; i++) {
                            if(rand.nextInt(100) + 1 < rate) {
                                city.add(new Person(true));
                            }
                            else {
                                city.add(new Person(false));
                            }
                        }
                        break;
                }
            }
            catch(NumberFormatException e) {
                System.out.println(e);
                System.out.println("Please type only numbers.");
            }
        }

        printPopulations();

```



```java
        //run the simulation

        while(Person.population > 0) {
            for(int i = 0, curPop = city.size(); i < city.size() && i < curPop; i++) {
                Person randPerson = city.get(rand.nextInt(city.size()));  // pick a random person from the city
                if(randPerson.isAlive() && city.get(i).isAlive()) {  //if the random person and current person are alive
                    Person baby = city.get(i).life(randPerson);  //perform all life stuff
                    if(baby != null) {
                        city.add(baby); // add a baby if one was born
                    }
                }
                if(city.get(i).isAlive() == false) {
                    city.remove(i);  //remove dead person if death has happened.
                }
            }
            printPopulations();
        }

    }

}

```