Lab 09

Exercise 1:

```
public class Ball {

	private double x, y;
	private double distTraveledX, distTraveledY;
	private static double totDistXAllBalls, totDistYAllBalls;
	private static double lastX, lastY;

	public Ball(){
		x = 0;
		y = 0;
		distTraveledX = 0;
		distTraveledY = 0;
	}

	public Ball(double newX, double newY){
		x = newX;
		y = newY;
		distTraveledX = 0;
		distTraveledY = 0;
	}

	public double getX(){
		return x;
	}
	public double getY(){
		return y;
	}
	public double getDistTraveledX(){
		return distTraveledX;
	}
	public double getDistTraveledY(){
		return distTraveledY;
	}
	public static double getTotDistXAllBalls(){
		return totDistXAllBalls;
	}
	public static double getTotDistYAllBalls(){
		return totDistYAllBalls;
	}
	public void move(double xDisp, double yDisp){

		if(x + xDisp == lastX && y + yDisp == lastY){
			System.out.println("Error. Can't move!");
			return;
		}
		x += xDisp;
		y += yDisp;
		lastX = x;
		lastY = y;
		distTraveledX += Math.abs(xDisp);
		distTraveledY += Math.abs(yDisp);
		totDistXAllBalls += Math.abs(xDisp);
		totDistYAllBalls += Math.abs(yDisp);

	}

	public String toString(){
		return "Ball @ ( " + x + " , " + y + " ).";
	}


}
```

```
public class testBall {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

Ball b1 = new Ball(2,2);
b1.move(3, -2); //(5,0), distX = 3, distY = 2, allX = 3, allY = 2.
b1.move(2, -7); //(7,-7), distX = 5, distY = 9, allX = 5, allY = 9.
Ball b2 = new Ball();
b2.move(7, -7);//(0,0)
b2.move(2, 4);//(2,4), distX = 2, distY = 4, allX = 7, allY = 13.
System.out.println(b1.toString());
System.out.println(b2.toString());
System.out.println(Ball.getTotDistXAllBalls());
System.out.println(Ball.getTotDistYAllBalls());
    }

}
```

Exercise 2:

```java
import java.util.Scanner;

public class Species {

	private String name;
	private int population;
	private double growthRate;

	public Species(){
		name = "";
		population = 0;
		growthRate = 0;
	}

	public Species(String n, int p, double g){
		name = n;
		population = p;
		growthRate = g;
	}

	public void readInput(){
		Scanner input = new Scanner(System.in);
		System.out.println("Enter species name: ");
		name = input.nextLine();
		System.out.println("Enter population: ");
		population = input.nextInt();
		while(population < 0){
		System.out.println("Error. Population can't be negative"
				+ "\nEnter population: ");
			population = input.nextInt();
		}
		System.out.println("Enter growth rate: ");
		growthRate = input.nextDouble();
	}

	public void writeOutput(){
		System.out.println("Species name: " + name);
		System.out.println("Species population: " + population);
		System.out.println("Species growthRate: " + growthRate);
	}
	public String getName(){
		return name;
	}
	public int getPopulation(){
		return population;
	}
	public double getGrowthRate(){
		return growthRate;
	}

	public void setSpecies(String n, int p, double g){
		if(p < 0){
	System.out.println("Error. Negative population. Exit program.");
	System.exit(0);
		}
		name = n;
		population = p;
		growthRate = g;
	}

	public int predictPopulation(int years){
		double newPop = population;
		for(int i = 1; i <= years; i++){
			newPop += newPop * (growthRate/100);
		}
		if(newPop <= 0) newPop = 0;
		return (int)newPop;
	}
	public boolean equals(Species otherSpecies){
		return name.equalsIgnoreCase(otherSpecies.name);
	}
	public boolean isPopulationLargerThan(Species otherObject){
		return this.population > otherObject.population;
	}
	public boolean isExtinct(){
		return population == 0;
	}
}
```

```
public class testSpecies {

	public static void main(String[] args) {
		// TODO Auto-generated method stub

	Species X = new Species();
	X.readInput();
	Species ArabianOryx = new Species("Arabian Oryx", 1000, 0.25);

		while(X.equals(ArabianOryx)){
			System.out.println("Error. Enter another species");
			X.readInput();
		}

		if(X.isExtinct())
			System.out.println("This species is extinct");
		else{
			if(X.isPopulationLargerThan(ArabianOryx))
	System.out.println("This species is already larger than AOryx");
			else{
	if(X.getGrowthRate() <= ArabianOryx.getGrowthRate())
	System.out.println("This species will never surpass the AOryx");
				else{
					int years = 1;
				while(X.predictPopulation(years) <
			ArabianOryx.predictPopulation(years))
						years++;
		System.out.println("After " + years + " years species " +
				X.getName() + " will surpass the ArabianOryx");
				}
			}
		}

	}

}
```