Resumed Inventory Code

Name \ Abdulrahman salah anwer

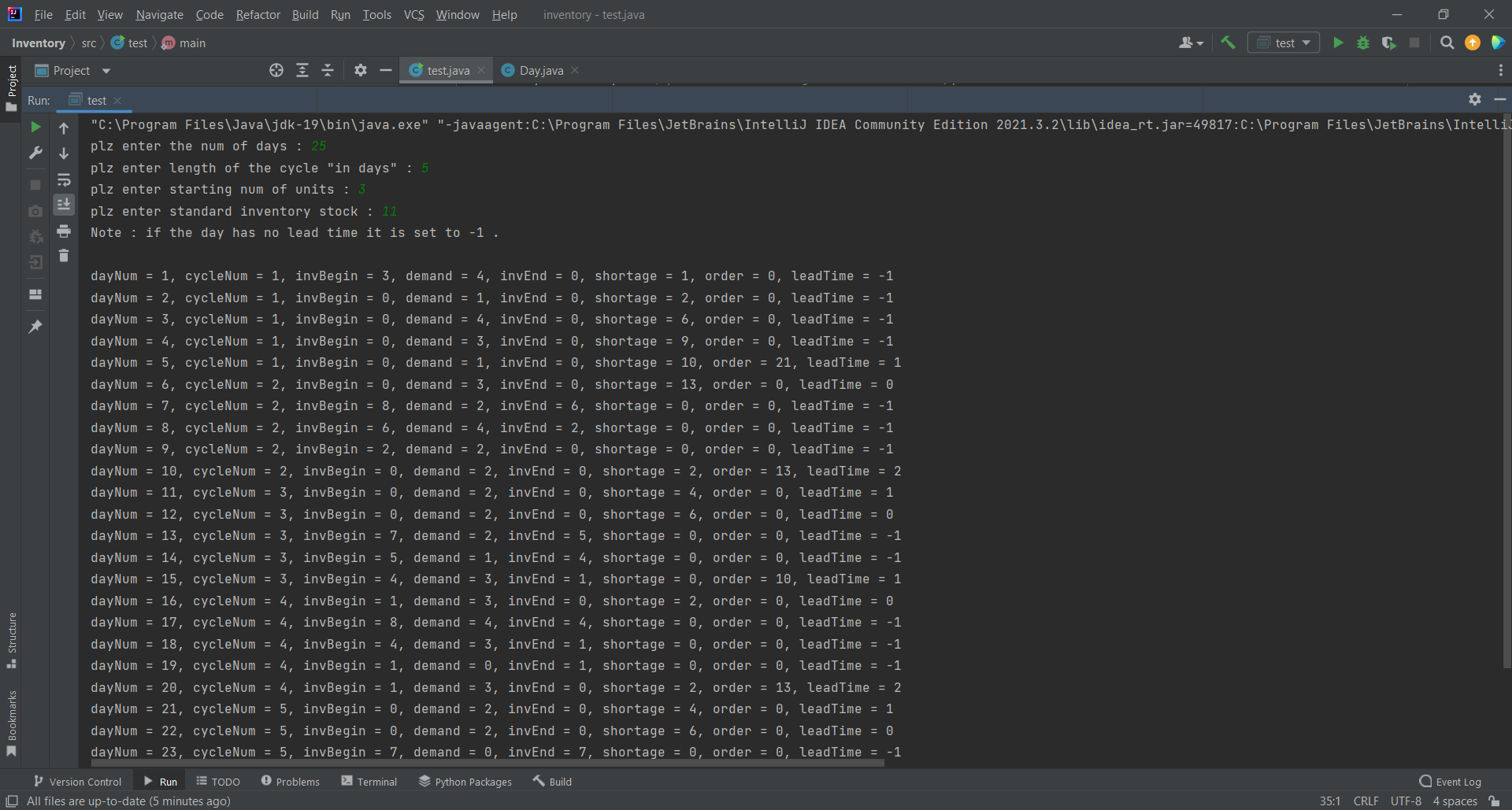
Main Class

import java.util.ArrayList;  
import java.util.Scanner;  
  
public class test {  
 public static void main(String[] args) {  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.print("plz enter the num of days : ");  
 int numOfDays = scan.nextInt();  
 System.*out*.print("plz enter length of the cycle \"in days\" : ");  
 int cycleLength = scan.nextInt();  
 System.*out*.print("plz enter starting num of units : ");  
 int startingNumOfUnits = scan.nextInt();  
 System.*out*.print("plz enter standard inventory stock : ");  
 int standardInvStock = scan.nextInt();  
 System.*out*.println("Note : if the day has no lead time it is set to -1 .");  
 System.*out*.println("");  
 ArrayList<Day> listOfDays = new ArrayList<Day>(numOfDays);  
  
 int dynamicCycleNum = 1;  
 int dayNum = 1;  
 for (int currDay = 0; currDay < numOfDays; currDay++) {  
 if (listOfDays.size() == 0) {  
 Day firstDay = new Day();  
 firstDay.dayNum = 1;  
 firstDay.cycleNum = 1;  
 firstDay.invBegin = startingNumOfUnits;  
 firstDay.demand = Day.*demand*();  
 if (firstDay.invBegin - firstDay.demand > 0) {  
 firstDay.invEnd = firstDay.invBegin - firstDay.demand;  
 } else {  
 firstDay.shortage = -1 \* (firstDay.invBegin - firstDay.demand);  
 }  
 firstDay.leadTime = -1;  
 firstDay.order = 0;  
 listOfDays.add(firstDay);  
 } else {  
 Day nextDay = new Day();  
 nextDay.dayNum = currDay + 1;  
 nextDay.cycleNum = dynamicCycleNum;  
 nextDay.invBegin = listOfDays.get(currDay - 1).invEnd;  
 nextDay.demand = Day.*demand*();  
 if (listOfDays.get(currDay - 1).leadTime > 0) {  
 nextDay.leadTime = listOfDays.get(currDay - 1).leadTime - 1;  
 } else if (listOfDays.get(currDay - 1).leadTime == 0) {  
 nextDay.invBegin = listOfDays.get(currDay - 1).invEnd + Day.*activeOrder* - listOfDays.get(currDay - 1).shortage;  
 if (nextDay.invBegin > standardInvStock) {  
 nextDay.invBegin = standardInvStock;  
 }  
 }  
 if (nextDay.invBegin - nextDay.demand > 0) {  
 nextDay.invEnd = nextDay.invBegin - nextDay.demand;  
 } else {  
 nextDay.shortage = (-1 \* (nextDay.invBegin - nextDay.demand)) + listOfDays.get(currDay - 1).shortage; //here we add to the current day's shortage the cumulative shortage , if the order has arrived and there is no shortage it will be set to 0 "the default value"  
 }  
 if (dayNum % cycleLength == 0) {  
 dynamicCycleNum++;  
 nextDay.order = standardInvStock - nextDay.invEnd + nextDay.shortage;  
 Day.*activeOrder* = nextDay.order;  
 nextDay.leadTime = Day.*leadTime*();  
 }  
 listOfDays.add(nextDay);  
 }  
 dayNum++;  
 }  
 for (Day day : listOfDays) {  
 System.*out*.println(day.toString());  
 }  
 }  
}

**Day Class**

import java.util.Random;  
  
public class Day {  
 int dayNum;  
 int cycleNum;  
 int invBegin;  
 int demand;  
 int invEnd;  
 int shortage;  
 int order;  
 int leadTime = -1;  
 static int *activeOrder*;  
 static int *shortagePerCycle*;  
  
 @Override  
 public String toString() {  
 return "dayNum = " + dayNum +  
 ", cycleNum = " + cycleNum +  
 ", invBegin = " + invBegin +  
 ", demand = " + demand +  
 ", invEnd = " + invEnd +  
 ", shortage = " + shortage +  
 ", order = " + order +  
 ", leadTime = " + leadTime;  
 }  
  
 public static int demand() {  
 Random rand = new Random();  
 int random\_number = rand.nextInt(100);  
 int demandValue = 0;  
 if (random\_number >= 0 && random\_number <= 9) {  
 demandValue = 0;  
 }  
 if (random\_number >= 10 && random\_number <= 34) {  
 demandValue = 1;  
 }  
 if (random\_number >= 35 && random\_number <= 69) {  
 demandValue = 2;  
 }  
 if (random\_number >= 70 && random\_number <= 90) {  
 demandValue = 3;  
 }  
 if (random\_number >= 91 && random\_number <= 99) {  
 demandValue = 4;  
 }  
 return demandValue;  
 }  
  
 public static int leadTime() {  
 Random rand = new Random();  
 int randomNumber = rand.nextInt(10);  
 int leadTimeValue = 0;  
 if (randomNumber >= 0 && randomNumber <= 5) {  
 leadTimeValue = 1;  
 }  
 if (randomNumber >= 6 && randomNumber <= 8) {  
 leadTimeValue = 2;  
 }  
 if (randomNumber == 9) {  
 leadTimeValue = 3;  
 }  
 return leadTimeValue;  
 }  
}

**The output**

****

**Done ..**