

# Lab 04 – Test Plan

Suproteek Banerjee

September 20, 2025

## 1 SumOfNumbers

Write a program that asks the user for a positive nonzero integer value and calculates the sum of all integers from 1 up to the number entered.

### IPO (Input, Processing, Output)

- **Input:** Positive nonzero integer
- **Processing:** Loop from 1 to input, sum numbers
- **Output:** Display total sum

### Java Program

```
1 // SumOfNumbers.java
2 import java.util.Scanner;
3
4 public class SumOfNumbers {
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         System.out.print("Enter a positive nonzero integer: ");
8         int number = sc.nextInt();
9         while (number <= 0) {
10             System.out.print("Invalid input. Please enter a positive
11                 nonzero integer: ");
12             number = sc.nextInt();
13         }
14         int sum = 0;
15         for (int i = 1; i <= number; i++) {
16             sum += i;
17         }
18         System.out.println("The sum of numbers from 1 to " + number + " is
19             " + sum);
20     }
21 }
```

## Test Plan

Input	Expected Output	Explanation
5	15	$1+2+3+4+5 = 15$
10	55	Sum of 1 to 10
1	1	Only one number, sum = 1
50	1275	Sum of 1 to 50

## 2 DistanceTraveled

Distance = Speed  $\times$  Time. Display distance traveled for each hour.

### Java Program

```
1 // DistanceTraveled.java
2 import java.util.Scanner;
3
4 class DistanceTraveled {
5     private int speed, hours;
6     public DistanceTraveled(int speed, int hours) { this.speed = speed;
7         this.hours = hours; }
8     public int getDistance(int hour) { return speed * hour; }
9     public int getHours() { return hours; }
10 }
11
12 public class DistanceTraveledDemo {
13     public static void main(String[] args) {
14         Scanner sc = new Scanner(System.in);
15         System.out.print("Enter speed: "); int speed = sc.nextInt();
16         while(speed<0){System.out.print("Invalid! Enter again: "); speed=
17             sc.nextInt();}
18         System.out.print("Enter hours: "); int hours=sc.nextInt();
19         while(hours<1){System.out.print("Invalid! Enter again: "); hours=
20             sc.nextInt();}
21         DistanceTraveled trip = new DistanceTraveled(speed,hours);
22         System.out.println("\nHour\tDistance Traveled");
23         System.out.println("-----");
24         for(int h=1;h<=trip.getHours();h++){
25             System.out.println(h + "\t" + trip.getDistance(h));
26         }
27     }
28 }
```

### Test Plan

Speed (mph)	Hours	Expected Output
40	3	Hour 3 $\rightarrow$ 120
55	5	Hour 5 $\rightarrow$ 275
60	1	Hour 1 $\rightarrow$ 60
30	4	Hour 4 $\rightarrow$ 120

### 3 PenniesForPay

Calculate daily pay doubling each day.

#### Java Program

```
1 // PenniesForPay.java
2 import java.util.Scanner;
3 import java.text.DecimalFormat;
4
5 public class PenniesForPay {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.print("Enter number of days: "); int days = sc.nextInt();
9         while(days<1){System.out.print("Invalid! Enter again: "); days=sc.nextInt();}
10        DecimalFormat df = new DecimalFormat("$#,##0.00");
11        double dailyPay=0.01, totalPay=0;
12        System.out.println("\nDay\tDaily Pay");
13        System.out.println("-----");
14        for(int d=1;d<=days;d++){
15            System.out.println(d + "\t" + df.format(dailyPay));
16            totalPay += dailyPay;
17            dailyPay *= 2;
18        }
19        System.out.println("\nTotal pay: " + df.format(totalPay));
20    }
21 }
```

#### Test Plan

Days	Expected Total	Explanation
1	\$0.01	Only day 1
5	\$0.31	0.01+0.02+0.04+0.08+0.16
10	\$10.23	Sum of 10 days doubling
15	\$327.67	Sum of 15 days doubling

## 4 HotelOccupancy

Calculate hotel occupancy rates.

### Java Program

```
1 // HotelOccupancy.java
2 import java.util.Scanner;
3 import java.text.DecimalFormat;
4
5 class Hotel {
6     private int totalFloors, totalRooms, totalOccupied;
7     public Hotel(int floors){totalFloors=floors; totalRooms=0;
8         totalOccupied=0;}
9     public void addFloorData(int rooms,int occupied){totalRooms+=rooms;
10         totalOccupied+=occupied;}
11     public int getVacantRooms(){return totalRooms-totalOccupied;}
12     public double getOccupancyRate(){return (double) totalOccupied/
13         totalRooms;}
14     public void displayReport(){
15         DecimalFormat df=new DecimalFormat("0.00%");
16         System.out.println("\n--- Hotel Occupancy Report ---");
17         System.out.println("Total Rooms : "+totalRooms);
18         System.out.println("Occupied Rooms: "+totalOccupied);
19         System.out.println("Vacant Rooms : "+getVacantRooms());
20         System.out.println("Occupancy Rate: "+df.format(getOccupancyRate()
21             ));
22     }
23 }
24
25 public class HotelOccupancy {
26     public static void main(String[] args){
27         Scanner sc = new Scanner(System.in);
28         System.out.print("Enter the number of floors: "); int floors = sc.
29             nextInt();
30         while(floors<1){System.out.print("Invalid! Enter again: "); floors
31             =sc.nextInt();}
32         Hotel hotel=new Hotel(floors);
33         for(int i=1;i<=floors;i++){
34             System.out.print("Enter number of rooms on floor "+i+": ");
35             int rooms=sc.nextInt();
36             while(rooms<10){System.out.print("Invalid! Enter again: ");
37                 rooms=sc.nextInt();}
38             System.out.print("Enter number of occupied rooms on floor "+i+
39                 ": "); int occ=sc.nextInt();
40             while(occ<0 || occ>rooms){System.out.print("Invalid! Enter
41                 again: "); occ=sc.nextInt();}
42             hotel.addFloorData(rooms,occ);
43         }
44         hotel.displayReport();
45         sc.close();
46     }
47 }
```

---

## Test Plan

Input	Expected Output	Explanation
5 floors, 2 rooms each	Occupied 4, Vacant 6	50% occupancy
3 floors, 3 rooms each	Occupied 5, Vacant 4	55.6% occupancy
10 floors, 10 rooms each	Occupied 80, Vacant 20	80% occupancy

## 5 PopulationGrowth

Predict population size over days.

### Java Program

```
1 // PopulationGrowth.java
2 import java.util.Scanner;
3
4 class Population {
5     private int start;
6     private double increase;
7     private int days;
8     public Population(int start, double increase, int days){this.start=start
9         ; this.increase=increase; this.days=days;}
10    public void displayPopulation(){
11        double pop=start;
12        System.out.println("\nDay\tPopulation");
13        for(int d=1;d<=days;d++){
14            System.out.printf("%d\t%.2f\n",d,pop);
15            pop+=pop*(increase/100.0);
16        }
17    }
18 }
19
20 public class PopulationGrowth{
21     public static void main(String[] args){
22         Scanner sc=new Scanner(System.in);
23         System.out.print("Enter starting size (>=2): "); int start=sc.
24             nextInt();
25         while(start<2){System.out.print("Invalid! Enter again: "); start=
26             sc.nextInt();}
27         System.out.print("Enter daily increase %: "); double inc=sc.
28             nextDouble();
29         while(inc<0){System.out.print("Invalid! Enter again: "); inc=sc.
30             nextDouble();}
31         System.out.print("Enter number of days (>=1): "); int days=sc.
32             nextInt();
33         while(days<1){System.out.print("Invalid! Enter again: "); days=sc.
34             nextInt();}
35         Population p=new Population(start,inc,days);
36         p.displayPopulation();
37         sc.close();
38     }
39 }
```

## Test Plan

Initial Population	Growth Rate (%)	Expected Output
1000	2	1020
500	5	525
200	10	220
10000	0	10000



## 6 AverageRainfall

Compute average rainfall over years.

### Java Program

```
1 // AverageRainfall.java
2 import java.util.Scanner;
3
4 public class AverageRainfall{
5     public static void main(String[] args){
6         Scanner sc=new Scanner(System.in);
7         System.out.print("Enter number of years (>=1): "); int years=sc.
            nextInt();
8         while(years<1){System.out.print("Invalid! Enter again: ");
9             years = sc.nextInt(); }
10
11         double totalRainfall = 0;
12         int totalMonths = years * 12;
13
14         for (int y = 1; y <= years; y++) {
15             System.out.println("Year " + y + ":");
16             for (int m = 1; m <= 12; m++) {
17                 System.out.print("   Enter rainfall for month " + m + " (in
                    inches): ");
18                 double rainfall = sc.nextDouble();
19                 while (rainfall < 0) {
20                     System.out.print("       Invalid input! Rainfall cannot
                        be negative: ");
21                     rainfall = sc.nextDouble();
22                 }
23                 totalRainfall += rainfall;
24             }
25         }
26
27         double averageRainfall = totalRainfall / totalMonths;
28
29         System.out.println("\n--- Rainfall Report ---");
30         System.out.println("Total months: " + totalMonths);
31         System.out.println("Total inches of rainfall: " + totalRainfall);
32         System.out.printf("Average rainfall per month: %.2f inches\n",
            averageRainfall);
33
34         sc.close();
35     }
36 }
```

## Test Plan

Input	Expected Output	Explanation
Years = 1, monthly rainfall = 1 each	Total months: 12, Total: 12, Average: 1.00	Each month 1 inch
Years = 2, monthly rainfall = 2 each	Total months: 24, Total: 48, Average: 2.00	Two years, all 2 inches
Years = 1, monthly rainfall = 0..11	Total months: 12, Total: 66, Average: 5.50	Sum of months 0+1+...+11 = 66
Years = 3, monthly rainfall varying	Correct total, average calculated	Tests nested loop correctness