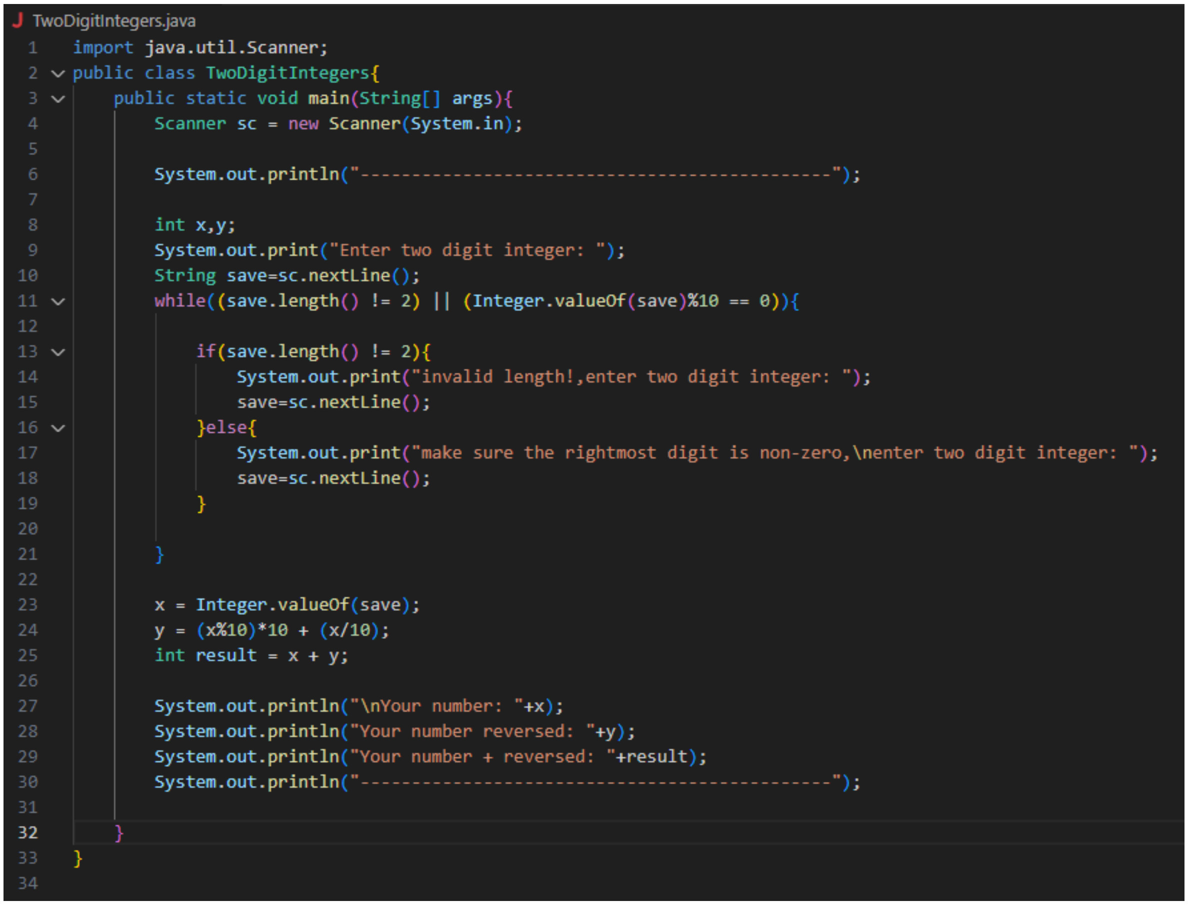
Java Fundamentals

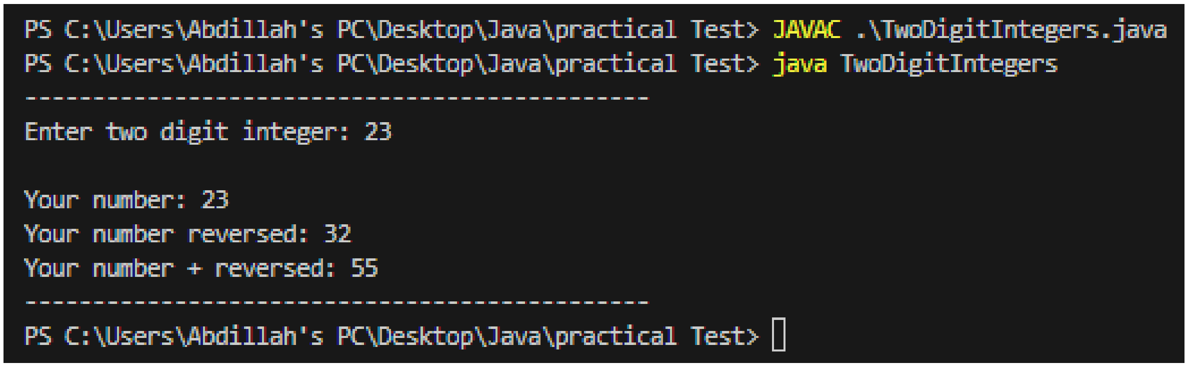
1. Write a program to input a 2 digit integer, call it x, where the rightmost digit is non-zero. Compute the integer y which has the same digits as x, but in reverse order. Print out x, y and x+y.

Program:



```java
import java.util.Scanner;
public class TwoDigitIntegers{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("----------------------------------------------");

        int x,y;
        System.out.print("Enter two digit integer: ");
        String save=sc.nextLine();
        while((save.length() != 2) || (Integer.valueOf(save)%10 == 0)){

            if(save.length() != 2){
                System.out.print("invalid length!,enter two digit integer: ");
                save=sc.nextLine();
            }else{
                System.out.print("make sure the rightmost digit is non-zero,\nenter two digit integer: ");
                save=sc.nextLine();
            }

        }

        x = Integer.valueOf(save);
        y = (x%10)*10 + (x/10);
        int result = x + y;

        System.out.println("\nYour number: "+x);
        System.out.println("Your number reversed: "+y);
        System.out.println("Your number + reversed: "+result);
        System.out.println("----------------------------------------------");

    }
}

```

Output:

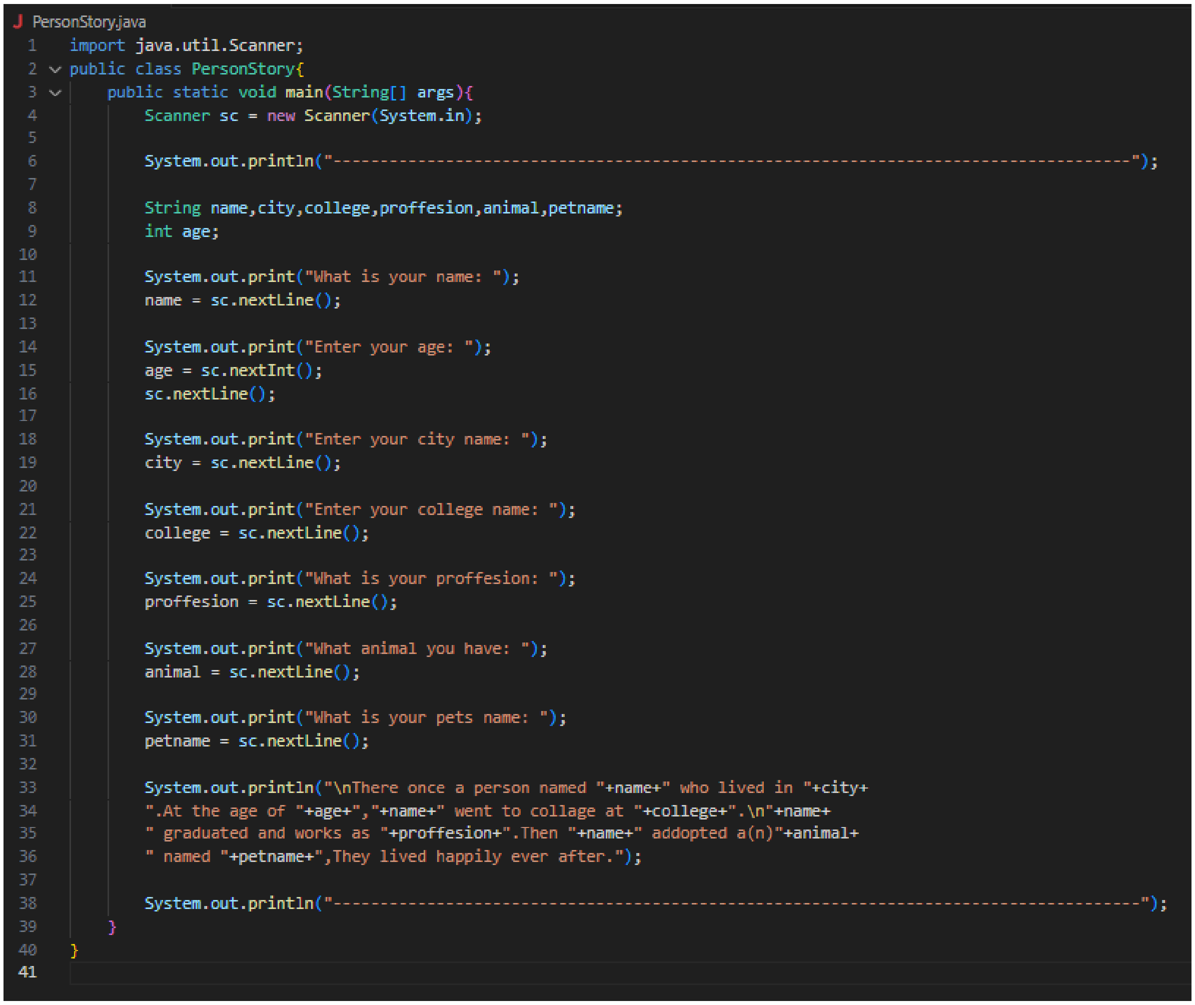


```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> JAVAC .\TwoDigitIntegers.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java TwoDigitIntegers
----------------------------------------------
Enter two digit integer: 23

Your number: 23
Your number reversed: 32
Your number + reversed: 55
----------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

2. Write a program that plays a word game with the user. The program should ask the user to enter the following:

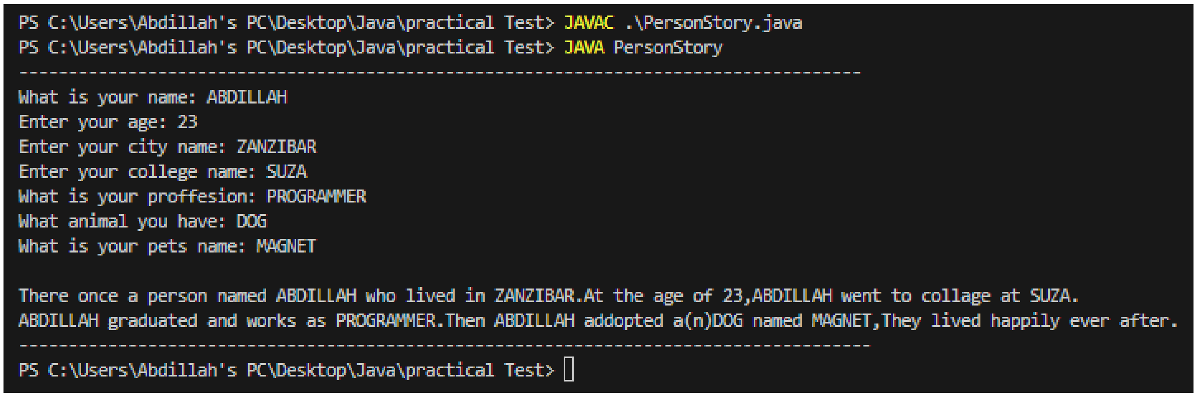
His or her name, His or her age, The name of a city, The name of a college, A profession, A type of animal, A pet’s name After the user has entered these items, the program should display the following story, inserting the user’s input into the appropriate locations There once was a person named **NAME** who lived in **CITY**. At the age of **AGE, NAME** went to college at **COLLEGE**. **NAME** graduated and went to work as a PROFESSION. Then, **NAME** adopted a(n) ANIMAL named **PETNAME**. They both lived happily ever after!

Program:



```java
import java.util.Scanner;
public class PersonStory{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("-------------------------------------------------------------------------------------");

        String name,city,college,proffesion,animal,petname;
        int age;

        System.out.print("What is your name: ");
        name = sc.nextLine();

        System.out.print("Enter your age: ");
        age = sc.nextInt();
        sc.nextLine();

        System.out.print("Enter your city name: ");
        city = sc.nextLine();

        System.out.print("Enter your college name: ");
        college = sc.nextLine();

        System.out.print("What is your proffesion: ");
        proffesion = sc.nextLine();

        System.out.print("What animal you have: ");
        animal = sc.nextLine();

        System.out.print("What is your pets name: ");
        petname = sc.nextLine();

        System.out.println("\nThere once a person named "+name+" who lived in "+city+
        ".At the age of "+age+","+name+" went to collage at "+college+".\n"+name+
        " graduated and works as "+proffesion+".Then "+name+" addopted a(n)"+animal+
        " named "+petname+",They lived happily ever after.");

        System.out.println("--------------------------------------------------------------------------------------");
    }
}

```

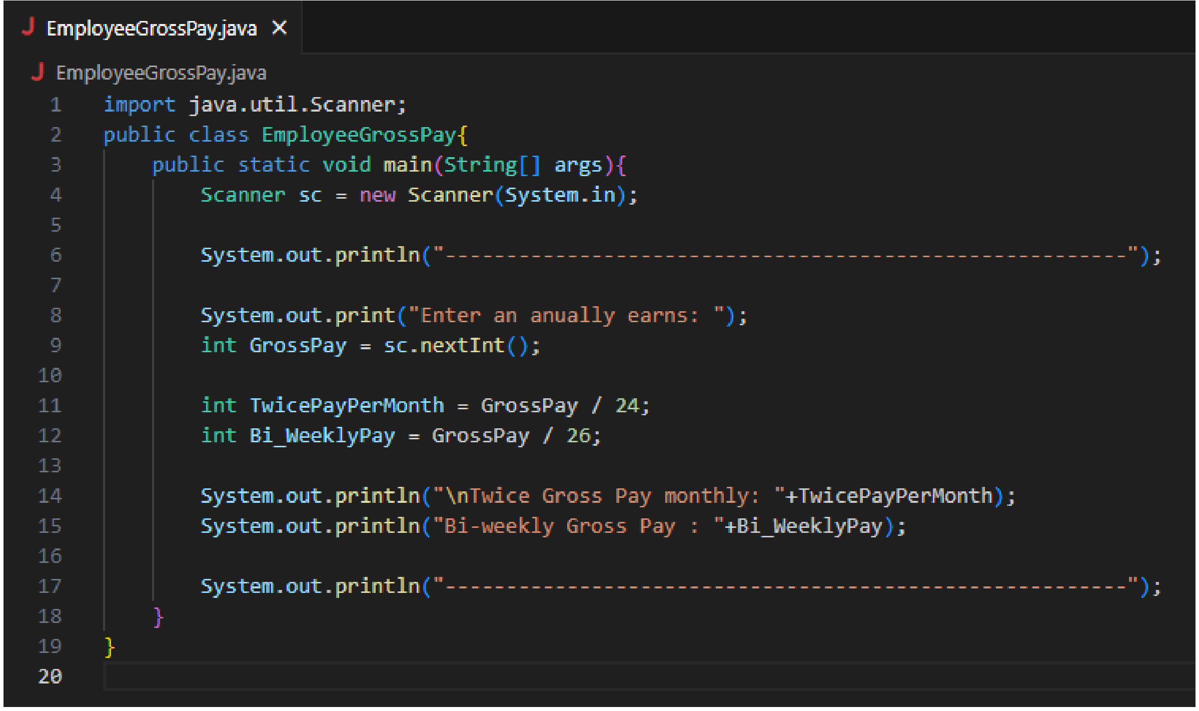
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> JAVAC .\PersonStory.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> JAVA PersonStory
----------------------------------------------------------------------------------------
What is your name: ABDILLAH
Enter your age: 23
Enter your city name: ZANZIBAR
Enter your college name: SUZA
What is your proffesion: PROGRAMMER
What animal you have: DOG
What is your pets name: MAGNET

There once a person named ABDILLAH who lived in ZANZIBAR.At the age of 23,ABDILLAH went to collage at SUZA.
ABDILLAH graduated and works as PROGRAMMER.Then ABDILLAH addopted a(n)DOG named MAGNET,They lived happily ever after.
-----------------------------------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

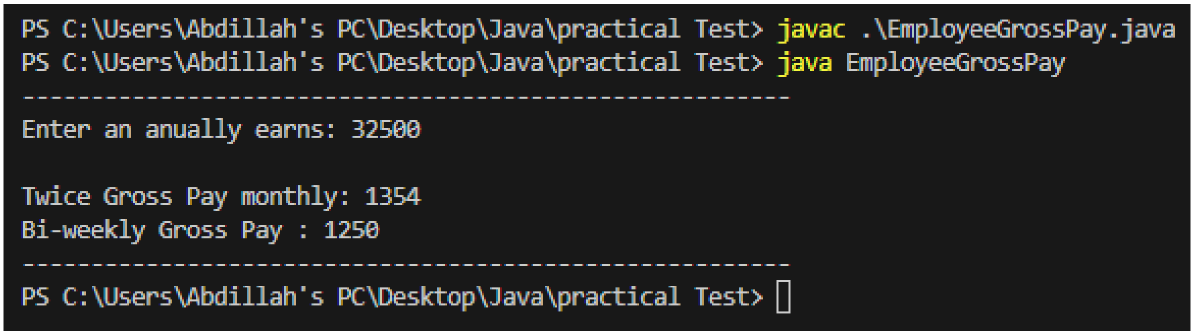
3. A particular employee earns $32,500 annually. Write a program that determines and displays what the amount of his gross pay will be for each pay period if he is paid twice a month (24 pay checks per year) and if he is paid bi-weekly (26 check per year).

Program:



```java
import java.util.Scanner;
public class EmployeeGrossPay{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("--------------------------------------------------------");

        System.out.print("Enter an anually earns: ");
        int GrossPay = sc.nextInt();

        int TwicePayPerMonth = GrossPay / 24;
        int Bi_WeeklyPay = GrossPay / 26;

        System.out.println("\nTwice Gross Pay monthly: "+TwicePayPerMonth);
        System.out.println("Bi-weekly Gross Pay : "+Bi_WeeklyPay);

        System.out.println("--------------------------------------------------------");
    }
}

```

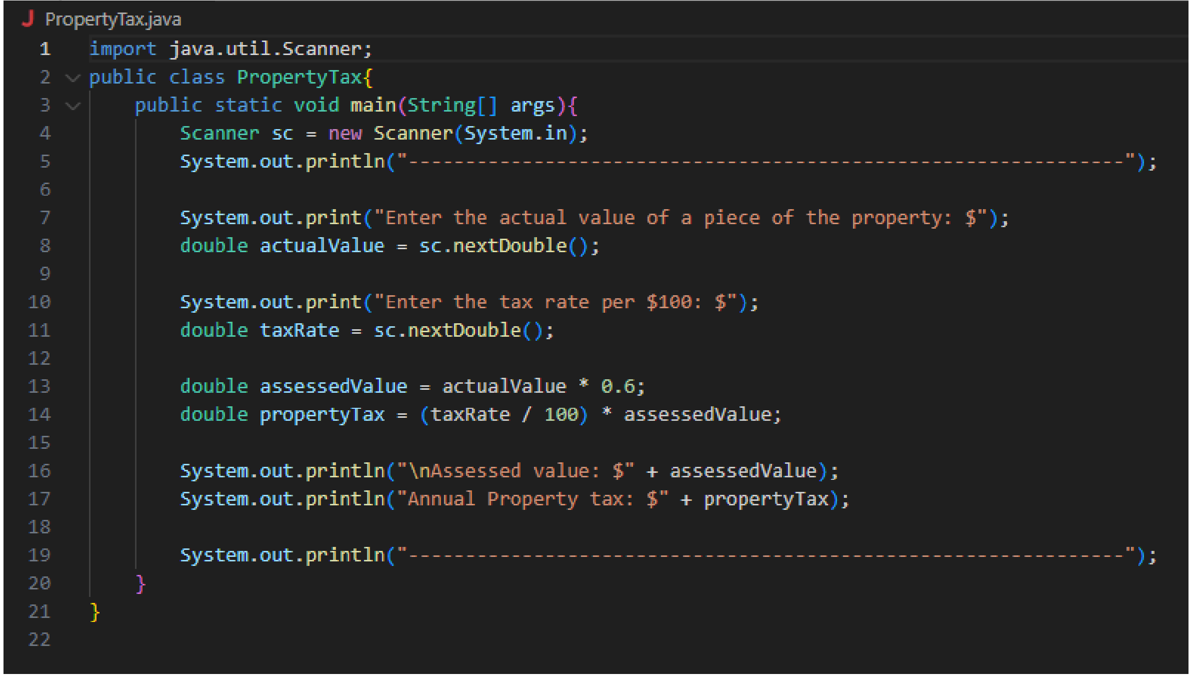
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\EmployeeGrossPay.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java EmployeeGrossPay
--------------------------------------------------------
Enter an anually earns: 32500

Twice Gross Pay monthly: 1354
Bi-weekly Gross Pay : 1250
--------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

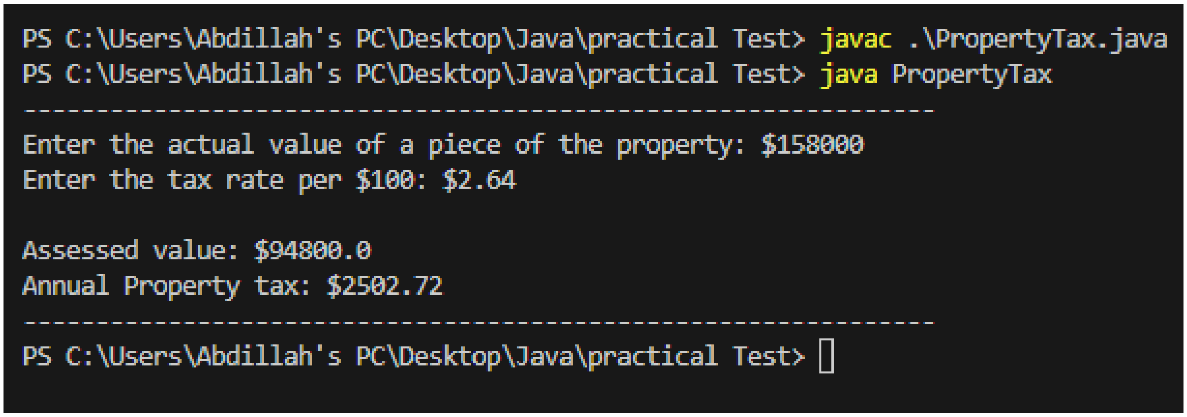
4. A county collects property taxes on the assessed value of property, which is 60 percent of its actual value. For example, if a house is valued at $158,000.00 its assessed value is $94,800. This is the amount the homeowner pays tax on. If the tax rate is $2.64 for each $100.00 of assessed value, the annual property tax for this house would be $2502.72. Write a program that asks the user for the actual value of a piece of property and the current tax rate for each $100.00 of assessed value. The program should then calculate and display how much annual property tax the homeowner will be charged for his property.

Program:



```java
import java.util.Scanner;
public class PropertyTax{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("---------------------------------------------------------------");

        System.out.print("Enter the actual value of a piece of the property: $");
        double actualValue = sc.nextDouble();

        System.out.print("Enter the tax rate per $100: $");
        double taxRate = sc.nextDouble();

        double assessedValue = actualValue * 0.6;
        double propertyTax = (taxRate / 100) * assessedValue;

        System.out.println("\nAssessed value: $" + assessedValue);
        System.out.println("Annual Property tax: $" + propertyTax);

        System.out.println("---------------------------------------------------------------");
    }
}

```

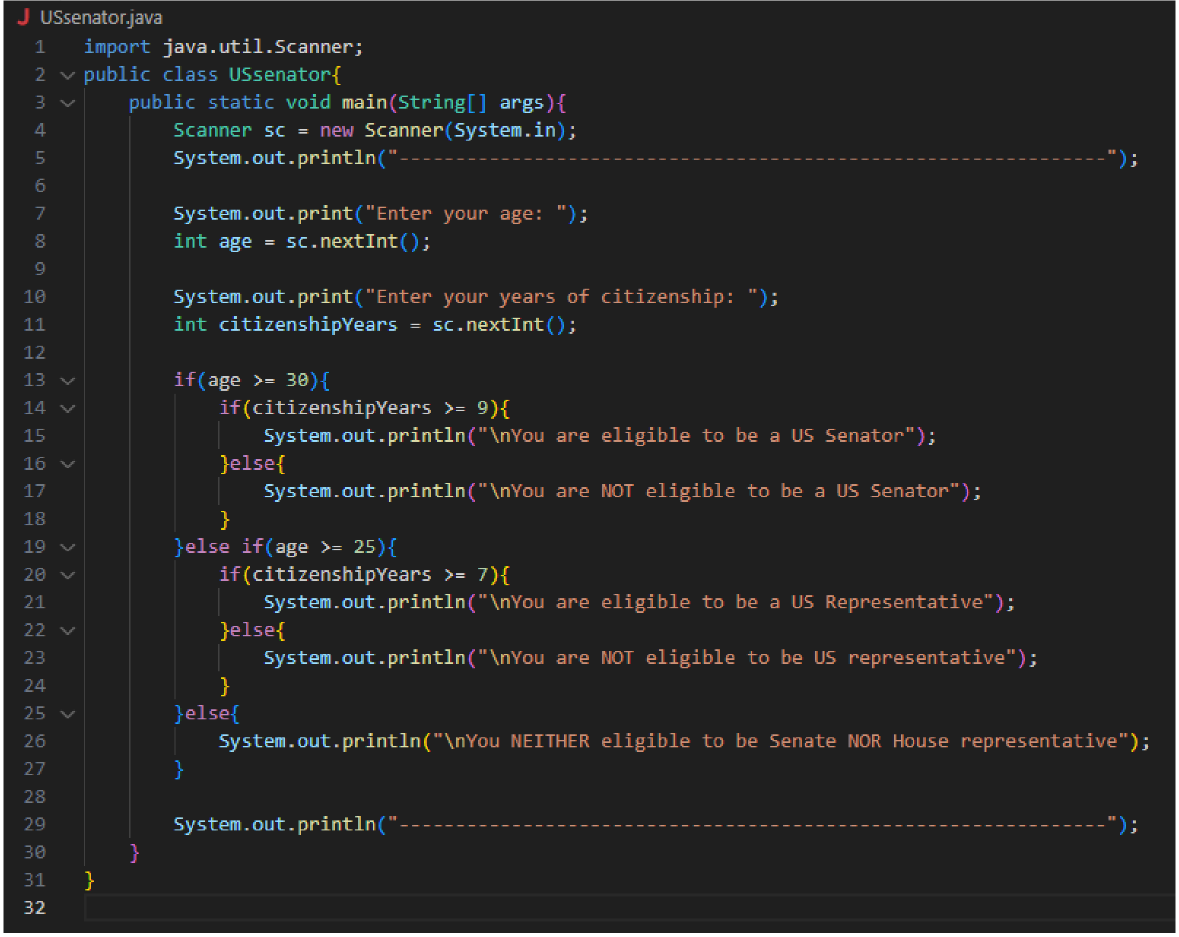
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\PropertyTax.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java PropertyTax
---------------------------------------------------------------
Enter the actual value of a piece of the property: $158000
Enter the tax rate per $100: $2.64

Assessed value: $94800.0
Annual Property tax: $2502.72
---------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

5. A person is eligible to be a US senator if they are at least 30 years old and have been a US citizen for at least 9 years. To be a US representative these numbers are 25 and 7, respectively. Write a program that accepts a person's age and years of citizenship as input and outputs their eligibility for the Senate and House.

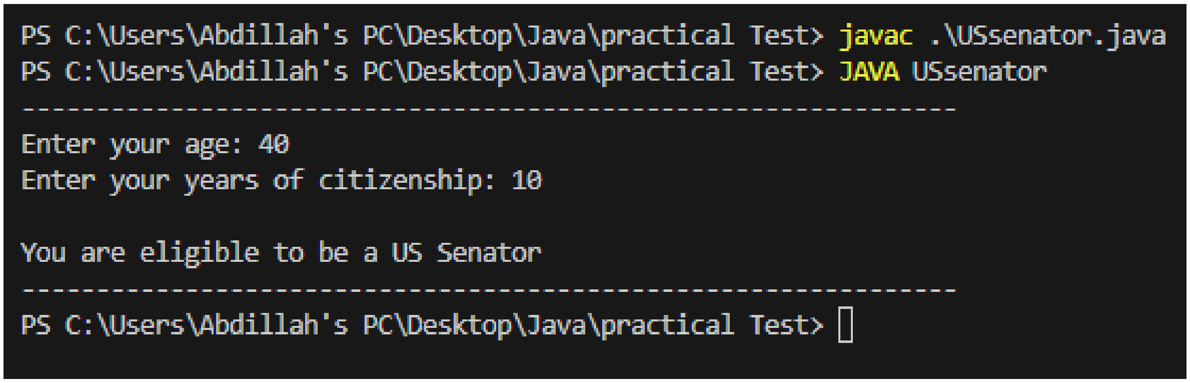
Program:



USsenator.java

```java
import java.util.Scanner;
public class USsenator{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("---------------------------------------------------------------");

        System.out.print("Enter your age: ");
        int age = sc.nextInt();

        System.out.print("Enter your years of citizenship: ");
        int citizenshipYears = sc.nextInt();

        if(age >= 30){
            if(citizenshipYears >= 9){
                System.out.println("\nYou are eligible to be a US Senator");
            }else{
                System.out.println("\nYou are NOT eligible to be a US Senator");
            }
        }else if(age >= 25){
            if(citizenshipYears >= 7){
                System.out.println("\nYou are eligible to be a US Representative");
            }else{
                System.out.println("\nYou are NOT eligible to be US representative");
            }
        }else{
            System.out.println("\nYou NEITHER eligible to be Senate NOR House representative");
        }

        System.out.println("---------------------------------------------------------------");
    }
}

```

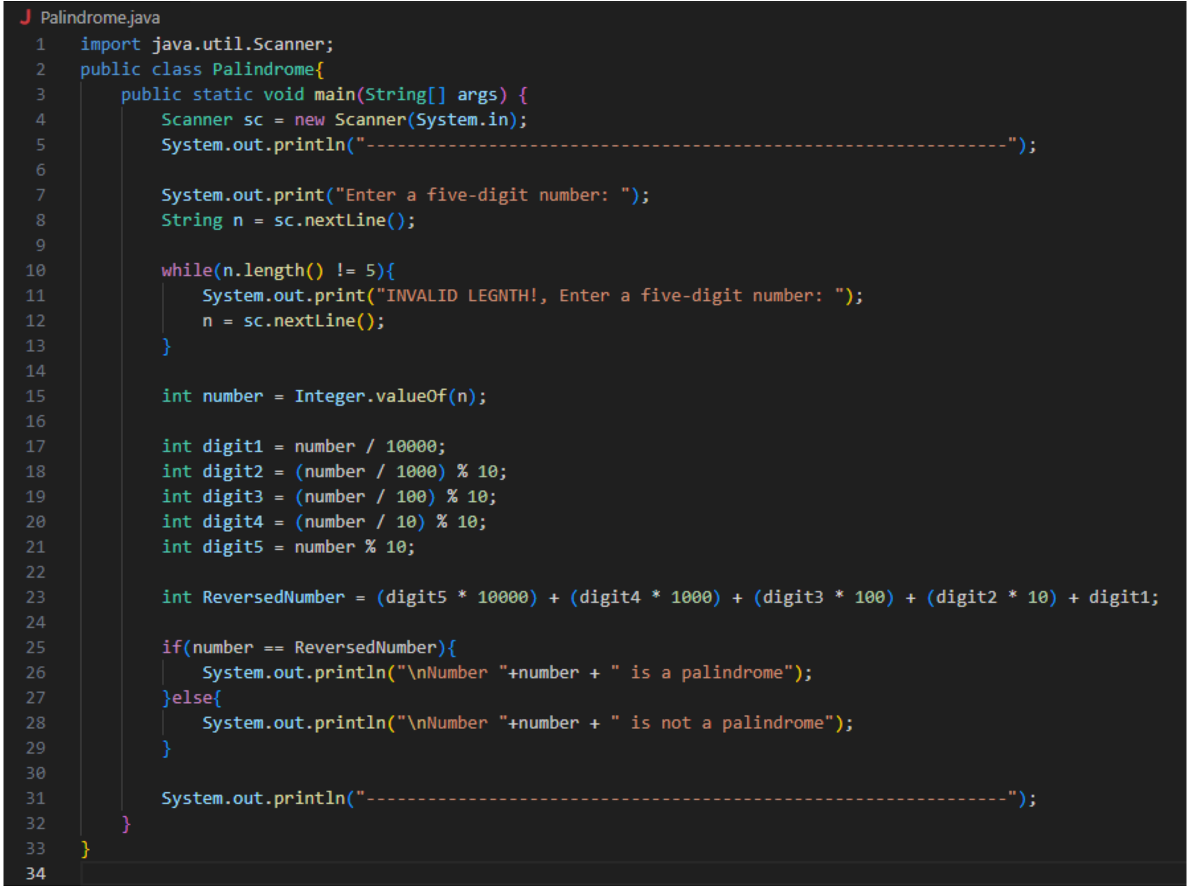
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\USsenator.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> JAVA USsenator
----------------------------------------------------------------
Enter your age: 40
Enter your years of citizenship: 10

You are eligible to be a US Senator
----------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

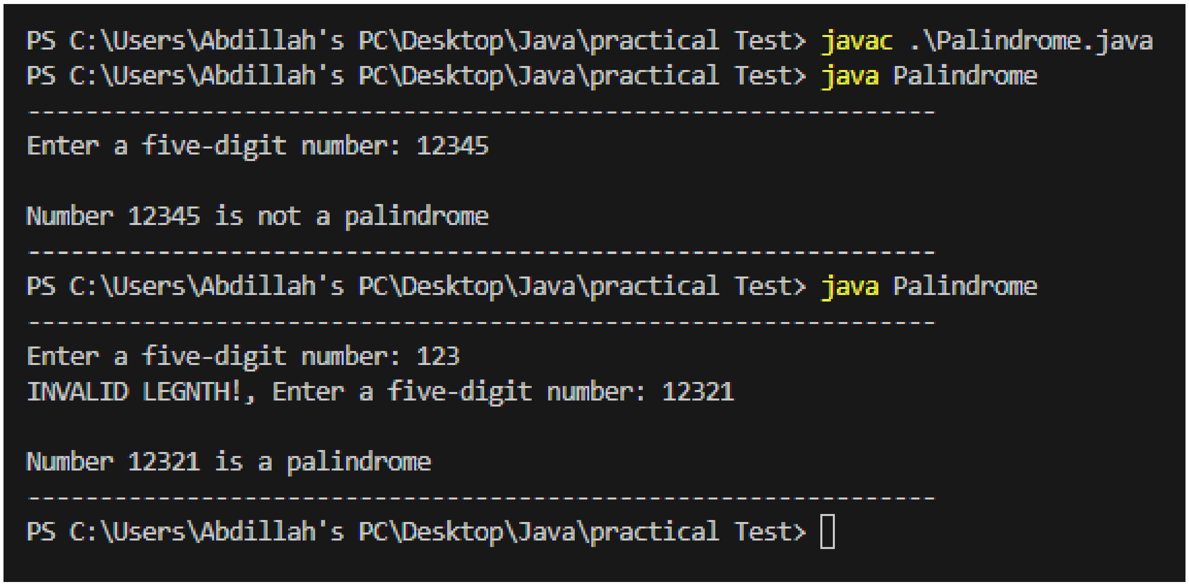
6. A palindrome is a number or a text phrase that reads the same backwards as forwards. For example, each of the following five-digit integers is a palindrome: 12321, 55555, 45554 and 11611. Write a program that reads in a five-digit integer and determines whether it is a palindrome.

Program:



```java
import java.util.Scanner;
public class Palindrome{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("----------------------------------------------------------------");

        System.out.print("Enter a five-digit number: ");
        String n = sc.nextLine();

        while(n.length() != 5){
            System.out.print("INVALID LEGNTH!, Enter a five-digit number: ");
            n = sc.nextLine();
        }

        int number = Integer.valueOf(n);

        int digit1 = number / 10000;
        int digit2 = (number / 1000) % 10;
        int digit3 = (number / 100) % 10;
        int digit4 = (number / 10) % 10;
        int digit5 = number % 10;

        int ReversedNumber = (digit5 * 10000) + (digit4 * 1000) + (digit3 * 100) + (digit2 * 10) + digit1;

        if(number == ReversedNumber){
            System.out.println("\nNumber "+number + " is a palindrome");
        }else{
            System.out.println("\nNumber "+number + " is not a palindrome");
        }

        System.out.println("----------------------------------------------------------------");
    }
}

```

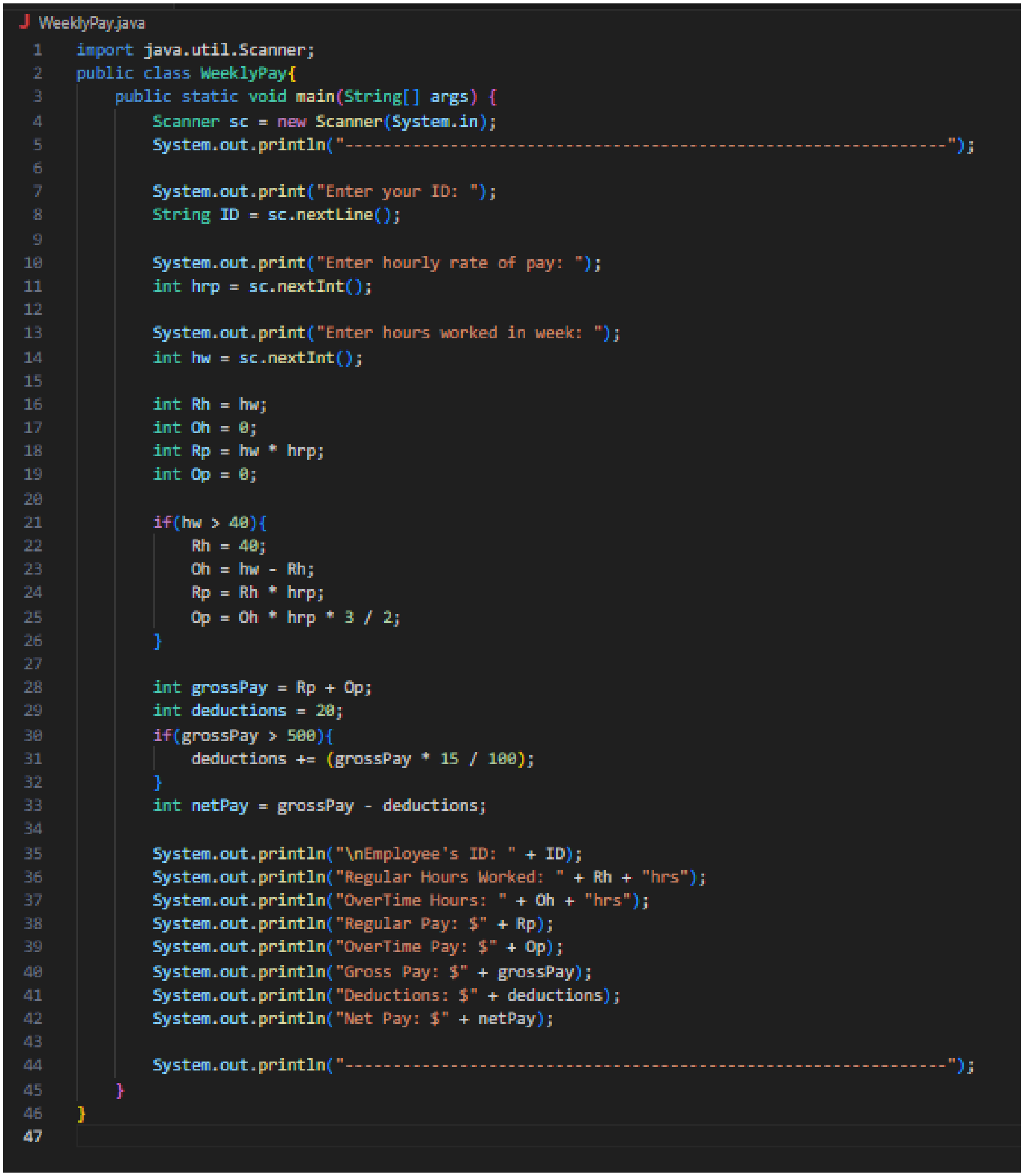
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\Palindrome.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java Palindrome
---------------------------------------------------------------
Enter a five-digit number: 12345

Number 12345 is not a palindrome
---------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java Palindrome
---------------------------------------------------------------
Enter a five-digit number: 123
INVALID LEGNTH!, Enter a five-digit number: 12321

Number 12321 is a palindrome
---------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

7. The owners of the Kwality Supermarket would like to have a program that computes the weekly gross pay of their employees. The user will enter an employee’s ID number, the hourly rate of pay, and the number of hours worked for the week. In addition, Kwality Supermarkets would like the program to compute the employee’s net pay and overtime pay. Overtime hours, any hours over 40, are paid at 1.5 the regular hourly rate. Net pay is Gross minus deductions. Assume that deductions are made up of income tax (at 15% of gross if the gross exceeds 500.00) and a 20 parking charge.

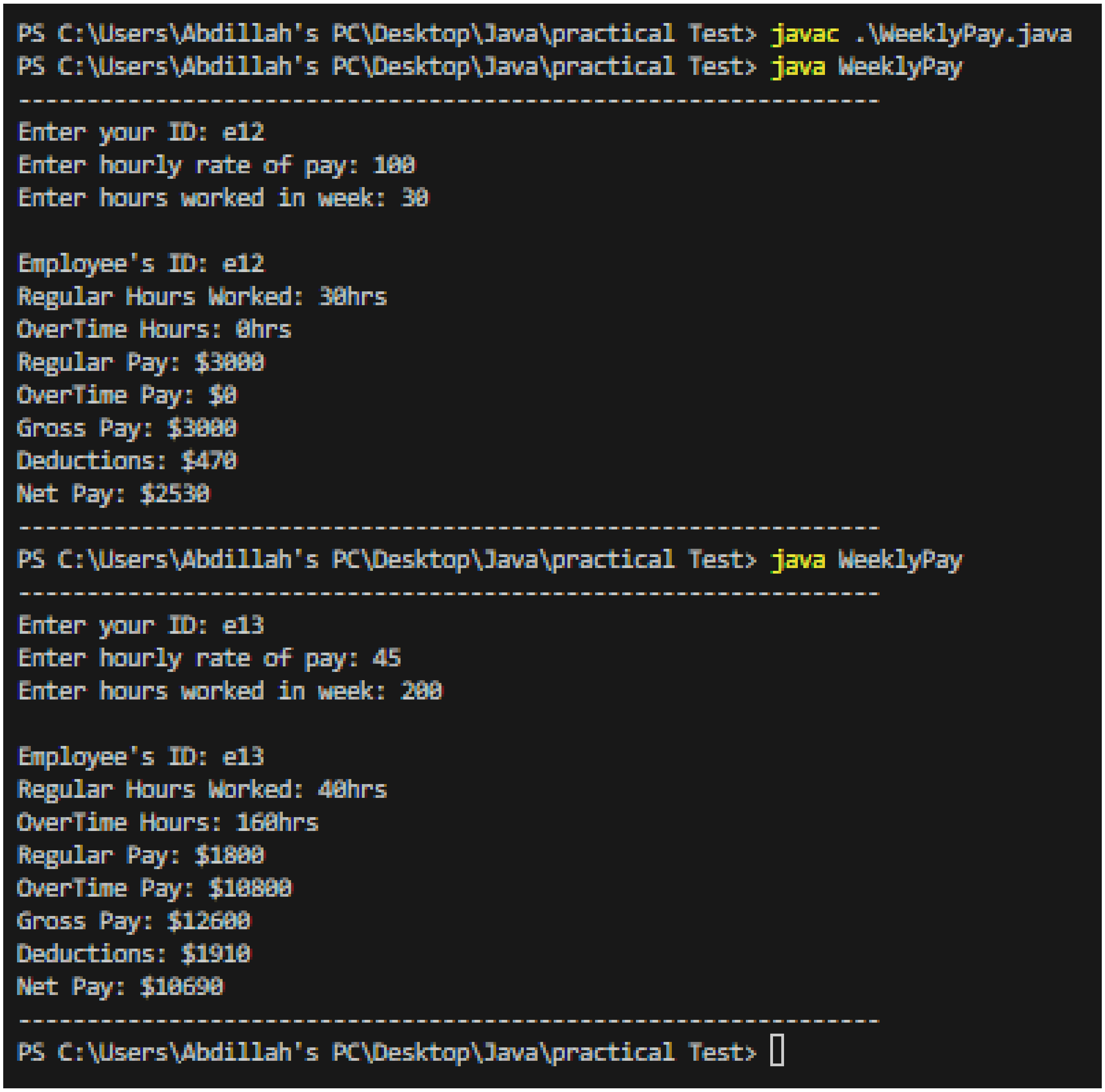
Program:



WeeklyPay.java

```java
import java.util.Scanner;
public class WeeklyPay{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("---------------------------------------------------------------");

        System.out.print("Enter your ID: ");
        String ID = sc.nextLine();

        System.out.print("Enter hourly rate of pay: ");
        int hrp = sc.nextInt();

        System.out.print("Enter hours worked in week: ");
        int hw = sc.nextInt();

        int Rh = hw;
        int Oh = 0;
        int Rp = hw * hrp;
        int Op = 0;

        if(hw > 40){
            Rh = 40;
            Oh = hw - Rh;
            Rp = Rh * hrp;
            Op = Oh * hrp * 3 / 2;
        }

        int grossPay = Rp + Op;
        int deductions = 20;
        if(grossPay > 500){
            deductions += (grossPay * 15 / 100);
        }
        int netPay = grossPay - deductions;

        System.out.println("\nEmployee's ID: " + ID);
        System.out.println("Regular Hours Worked: " + Rh + "hrs");
        System.out.println("OverTime Hours: " + Oh + "hrs");
        System.out.println("Regular Pay: $" + Rp);
        System.out.println("OverTime Pay: $" + Op);
        System.out.println("Gross Pay: $" + grossPay);
        System.out.println("Deductions: $" + deductions);
        System.out.println("Net Pay: $" + netPay);

        System.out.println("---------------------------------------------------------------");
    }
}

```

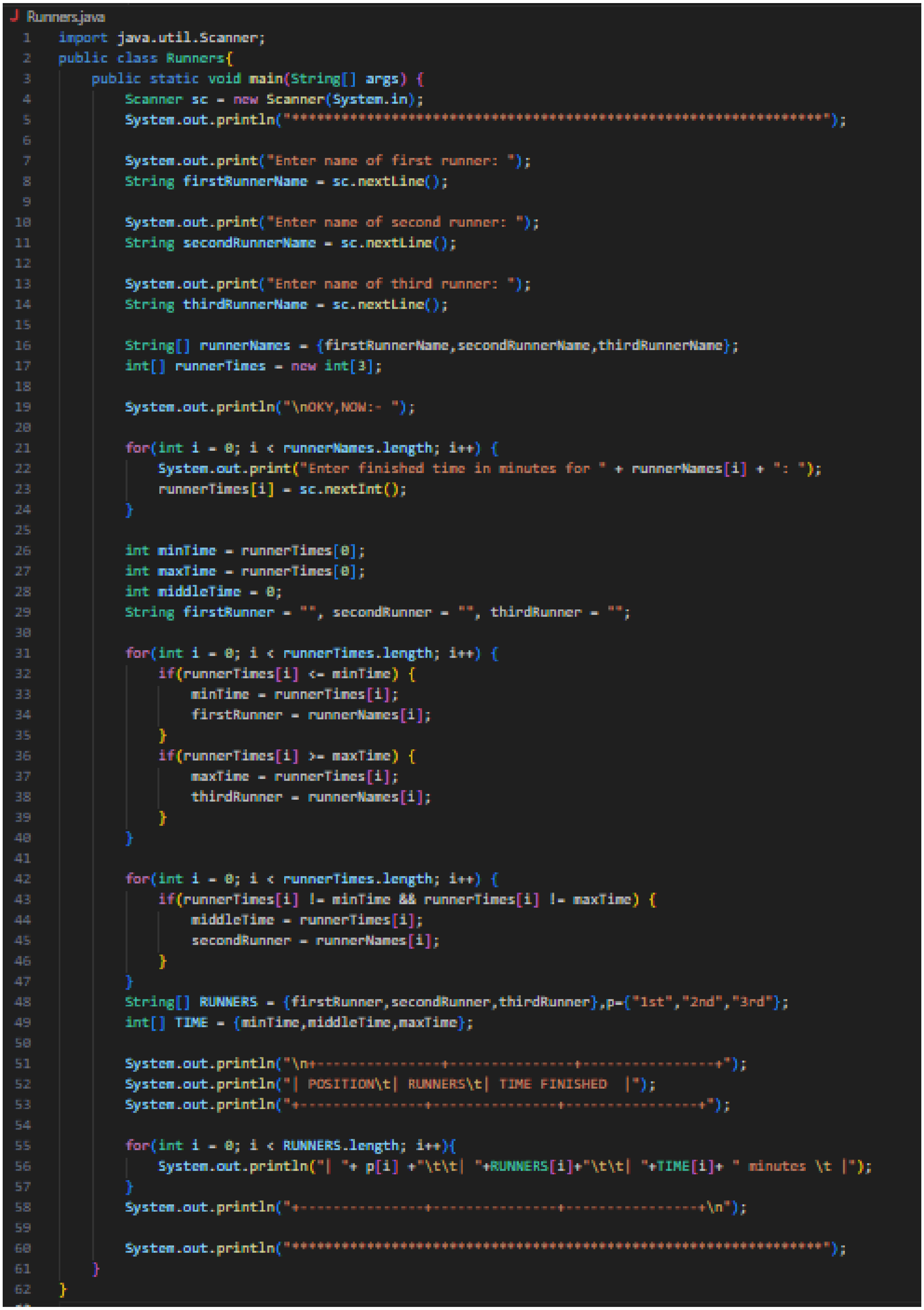
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\WeeklyPay.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java WeeklyPay
---------------------------------------------------------------
Enter your ID: e12
Enter hourly rate of pay: 100
Enter hours worked in week: 30

Employee's ID: e12
Regular Hours Worked: 30hrs
OverTime Hours: 0hrs
Regular Pay: $3000
OverTime Pay: $0
Gross Pay: $3000
Deductions: $470
Net Pay: $2530
---------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java WeeklyPay
---------------------------------------------------------------
Enter your ID: e13
Enter hourly rate of pay: 45
Enter hours worked in week: 200

Employee's ID: e13
Regular Hours Worked: 40hrs
OverTime Hours: 160hrs
Regular Pay: $1800
OverTime Pay: $10800
Gross Pay: $12600
Deductions: $1910
Net Pay: $10690
---------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

8. Write a program that asks for the names of three runners and the time, in minutes, it took each of them to finish a race. The program should display the names of the runners in the order that they finished.

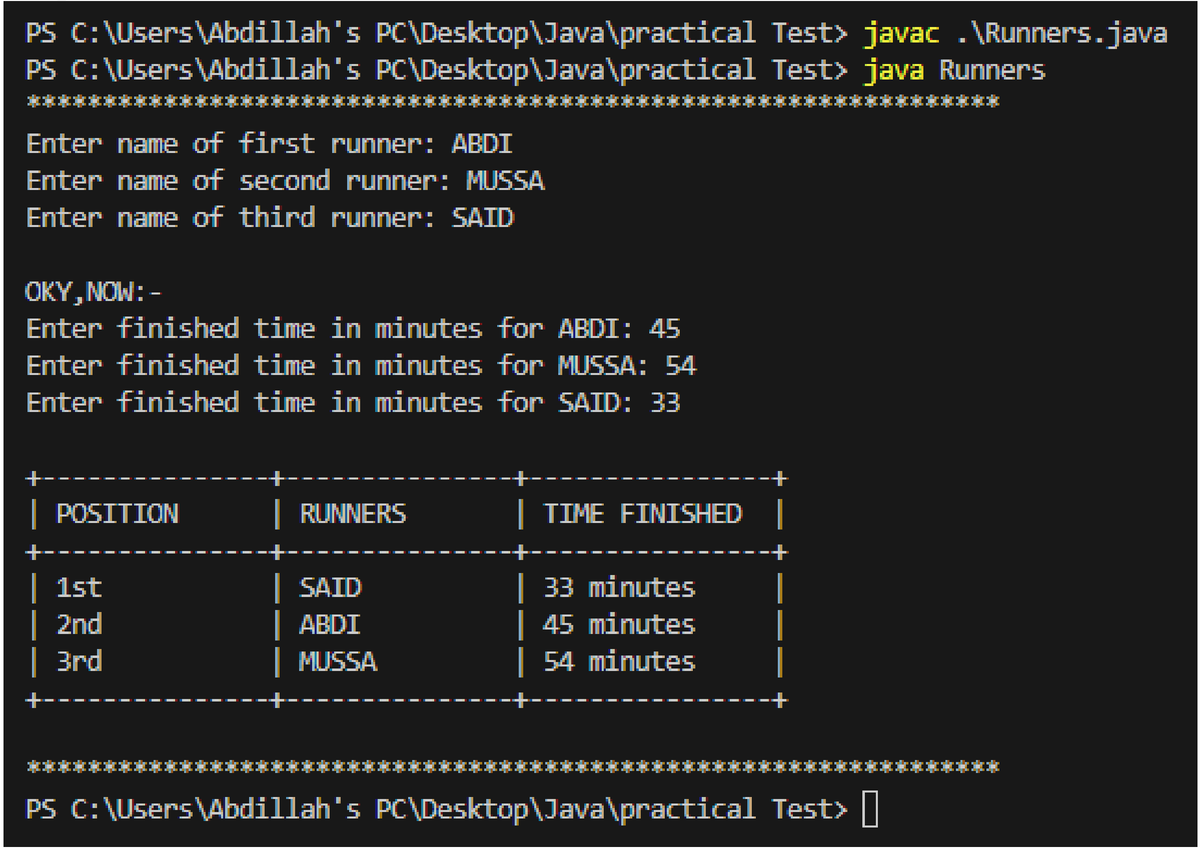
Program:



Runners.java

```java
import java.util.Scanner;
public class Runners{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("*****************************************************************");

        System.out.print("Enter name of first runner: ");
        String firstRunnerName = sc.nextLine();

        System.out.print("Enter name of second runner: ");
        String secondRunnerName = sc.nextLine();

        System.out.print("Enter name of third runner: ");
        String thirdRunnerName = sc.nextLine();

        String[] runnerNames = {firstRunnerName,secondRunnerName,thirdRunnerName};
        int[] runnerTimes = new int[3];

        System.out.println("\nOKY,NOW:- ");

        for(int i = 0; i < runnerNames.length; i++) {
            System.out.print("Enter finished time in minutes for " + runnerNames[i] + ": ");
            runnerTimes[i] = sc.nextInt();
        }

        int minTime = runnerTimes[0];
        int maxTime = runnerTimes[0];
        int middleTime = 0;
        String firstRunner = "", secondRunner = "", thirdRunner = "";

        for(int i = 0; i < runnerTimes.length; i++) {
            if(runnerTimes[i] <= minTime) {
                minTime = runnerTimes[i];
                firstRunner = runnerNames[i];
            }
            if(runnerTimes[i] >= maxTime) {
                maxTime = runnerTimes[i];
                thirdRunner = runnerNames[i];
            }
        }

        for(int i = 0; i < runnerTimes.length; i++) {
            if(runnerTimes[i] != minTime && runnerTimes[i] != maxTime) {
                middleTime = runnerTimes[i];
                secondRunner = runnerNames[i];
            }
        }
        String[] RUNNERS = {firstRunner,secondRunner,thirdRunner},p={"1st","2nd","3rd"};
        int[] TIME = {minTime,middleTime,maxTime};

        System.out.println("\n+---------------+---------------+-----------------+");
        System.out.println("| POSITION\t| RUNNERS\t| TIME FINISHED  |");
        System.out.println("+---------------+---------------+-----------------+");

        for(int i = 0; i < RUNNERS.length; i++){
            System.out.println("| "+ p[i] +"\t\t| "+RUNNERS[i]+"\t\t| "+TIME[i]+ " minutes \t |");
        }
        System.out.println("+---------------+---------------+-----------------+\n");

        System.out.println("*****************************************************************");
    }
}
```

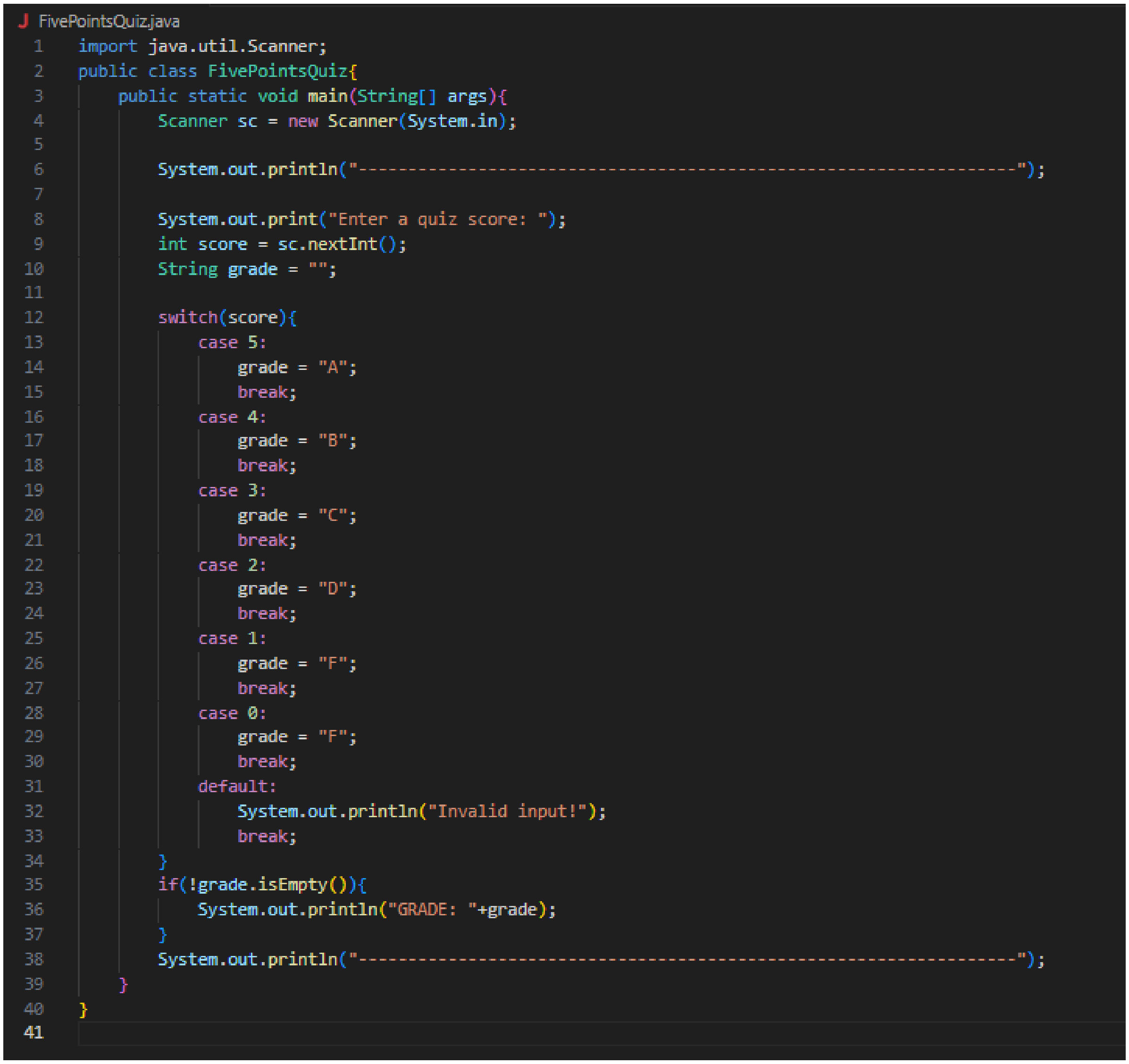
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\Runners.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java Runners
****************************************************************
Enter name of first runner: ABDI
Enter name of second runner: MUSSA
Enter name of third runner: SAID

OKY,NOW:-
Enter finished time in minutes for ABDI: 45
Enter finished time in minutes for MUSSA: 54
Enter finished time in minutes for SAID: 33

+---------------+---------------+-----------------+
| POSITION      | RUNNERS       | TIME FINISHED   |
+---------------+---------------+-----------------+
| 1st           | SAID          | 33 minutes      |
| 2nd           | ABDI          | 45 minutes      |
| 3rd           | MUSSA         | 54 minutes      |
+---------------+---------------+-----------------+

****************************************************************
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

9. A certain CS professor gives 5-point quizzes that are graded on the scale 5-A, 4-B, 3-C, 2-D, 1-F, 0-F. Write a program that accepts a quiz score as an input and uses a decision structure to calculate the corresponding grade.

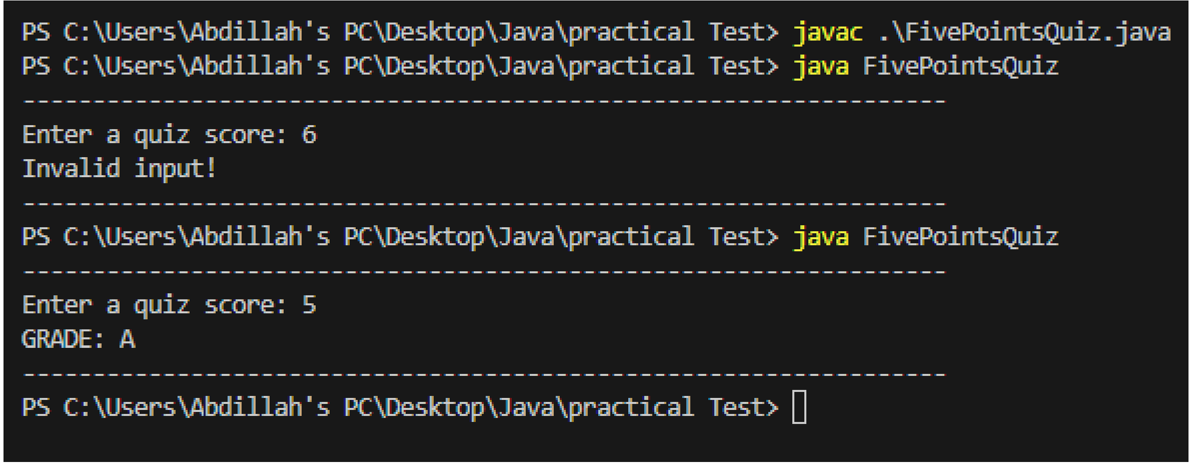
Program:



FivePointsQuiz.java

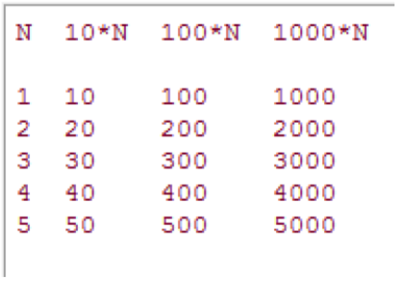
```java
import java.util.Scanner;
public class FivePointsQuiz{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("------------------------------------------------------------------");

        System.out.print("Enter a quiz score: ");
        int score = sc.nextInt();
        String grade = "";

        switch(score){
            case 5:
                grade = "A";
                break;
            case 4:
                grade = "B";
                break;
            case 3:
                grade = "C";
                break;
            case 2:
                grade = "D";
                break;
            case 1:
                grade = "F";
                break;
            case 0:
                grade = "F";
                break;
            default:
                System.out.println("Invalid input!");
                break;
        }
        if(!grade.isEmpty()){
            System.out.println("GRADE: "+grade);
        }
        System.out.println("------------------------------------------------------------------");
    }
}
```

Output:



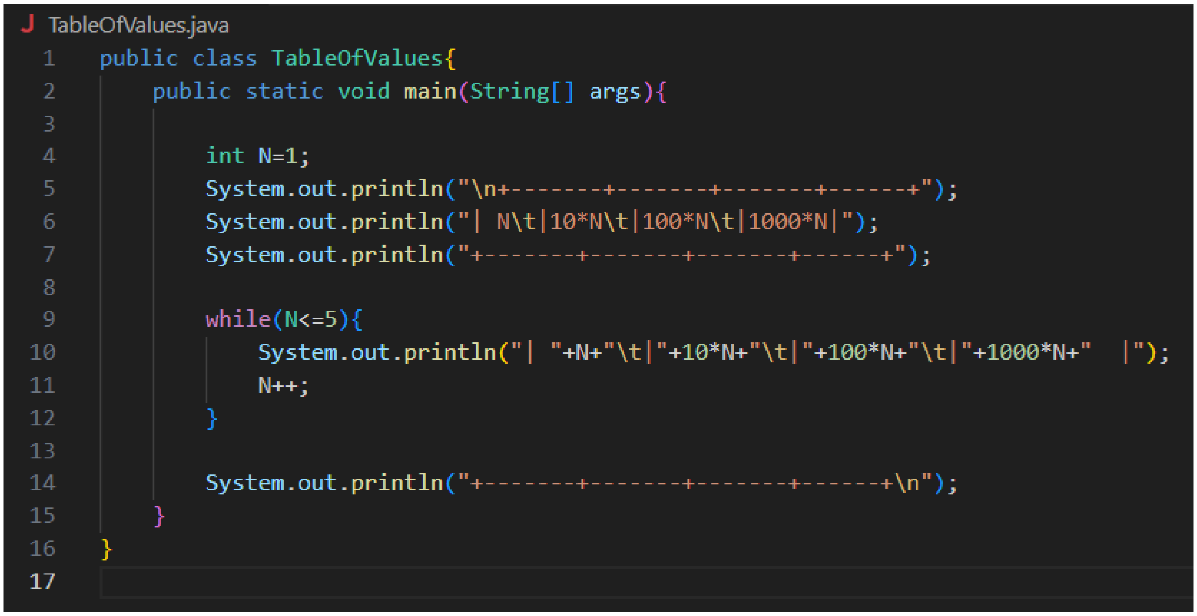
```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\FivePointsQuiz.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java FivePointsQuiz
------------------------------------------------------------------
Enter a quiz score: 6
Invalid input!
------------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java FivePointsQuiz
------------------------------------------------------------------
Enter a quiz score: 5
GRADE: A
------------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

10. Write a Java program that uses a while statement and the tab escape sequence \t to print the following table of values:



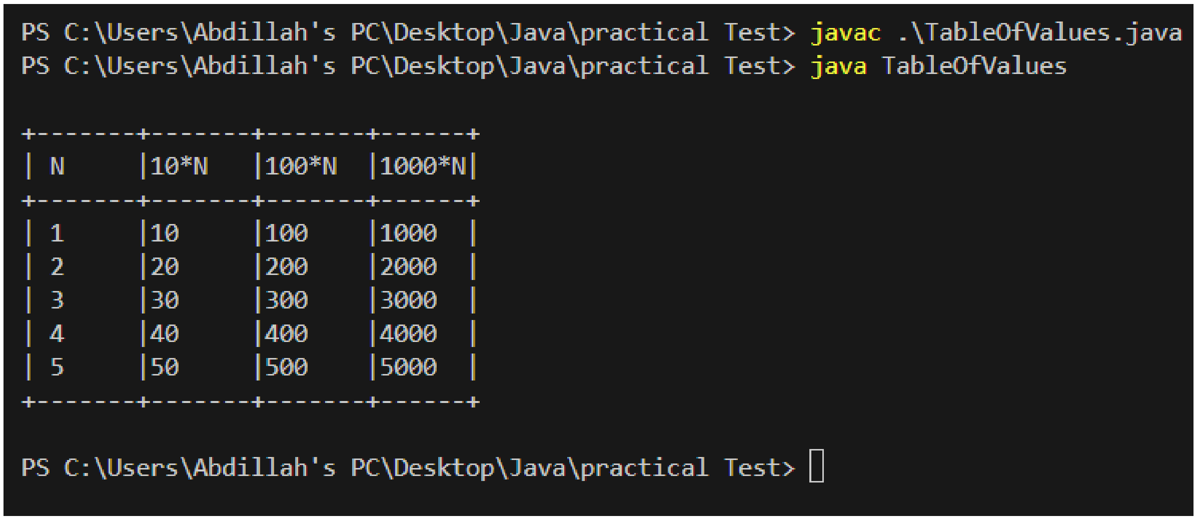
```
N   10*N   100*N   1000*N

1   10     100     1000
2   20     200     2000
3   30     300     3000
4   40     400     4000
5   50     500     5000
```

Program:



```java
public class TableOfValues{
    public static void main(String[] args){

        int N=1;
        System.out.println("\n+-------+-------+-------+------+");
        System.out.println("| N\t|10*N\t|100*N\t|1000*N|");
        System.out.println("+-------+-------+-------+------+");

        while(N<=5){
            System.out.println("| "+N+"\t|"+10*N+"\t|"+100*N+"\t|"+1000*N+"   |");
            N++;
        }

        System.out.println("+-------+-------+-------+------+\n");
    }
}
```

Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\TableOfValues.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java TableOfValues

+-------+-------+-------+------+
| N     |10*N   |100*N  |1000*N|
+-------+-------+-------+------+
| 1     |10     |100    |1000   |
| 2     |20     |200    |2000   |
| 3     |30     |300    |3000   |
| 4     |40     |400    |4000   |
| 5     |50     |500    |5000   |
+-------+-------+-------+------+

PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

11. The distance a vehicle travels can be calculated as follows :

    *distance = speed * time*

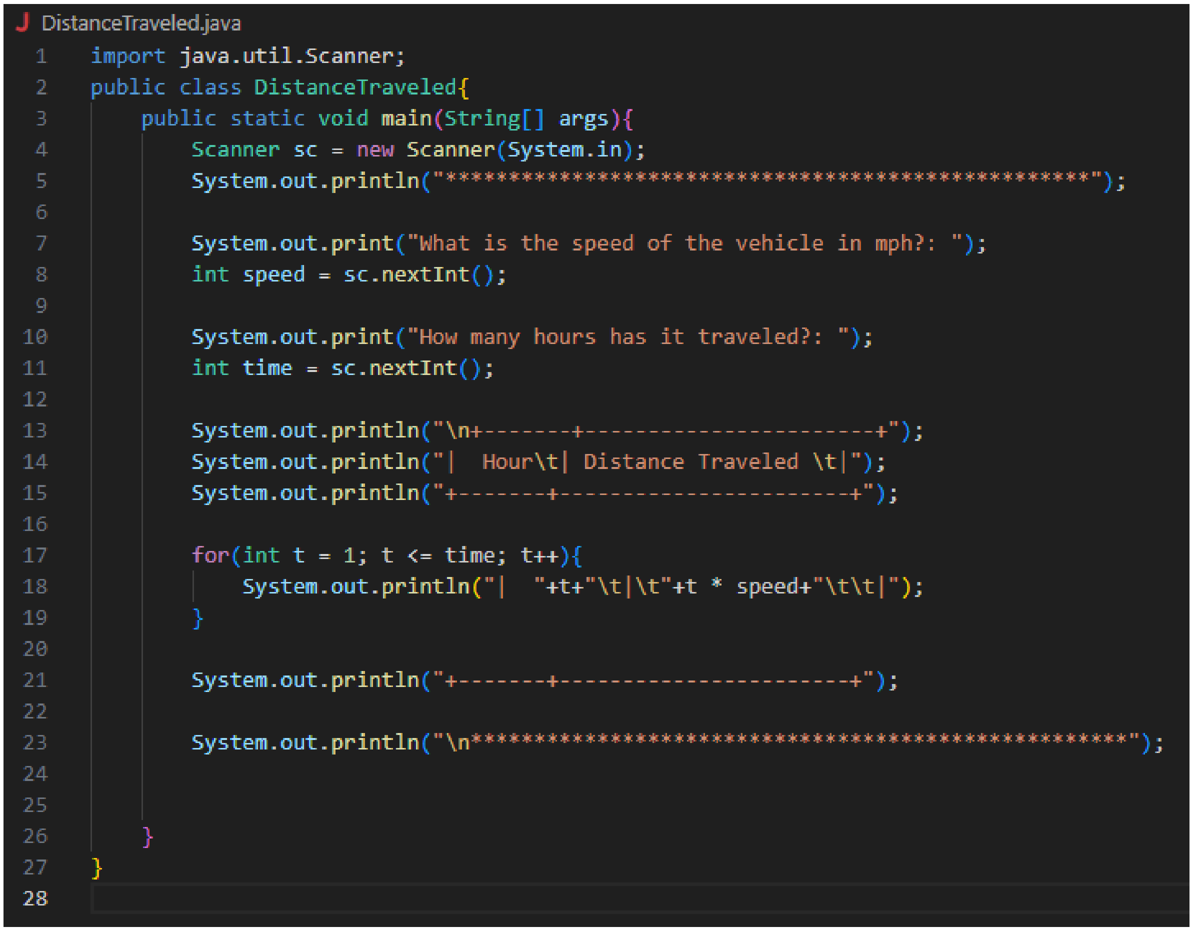
For example, if a train travels 40 miles per hour for three hours, the distance traveled is 120 miles. Write a program that asks the user for the speed of a vehicle (in miles per hour) and the number of hours it has traveled. It should then use a loop to display the distance the vehicle has traveled for each hour of that time period. Here is an example of the desired output:

What is the speed of the vehicle in mph? **40**e

How many hours has it traveled? **3**e

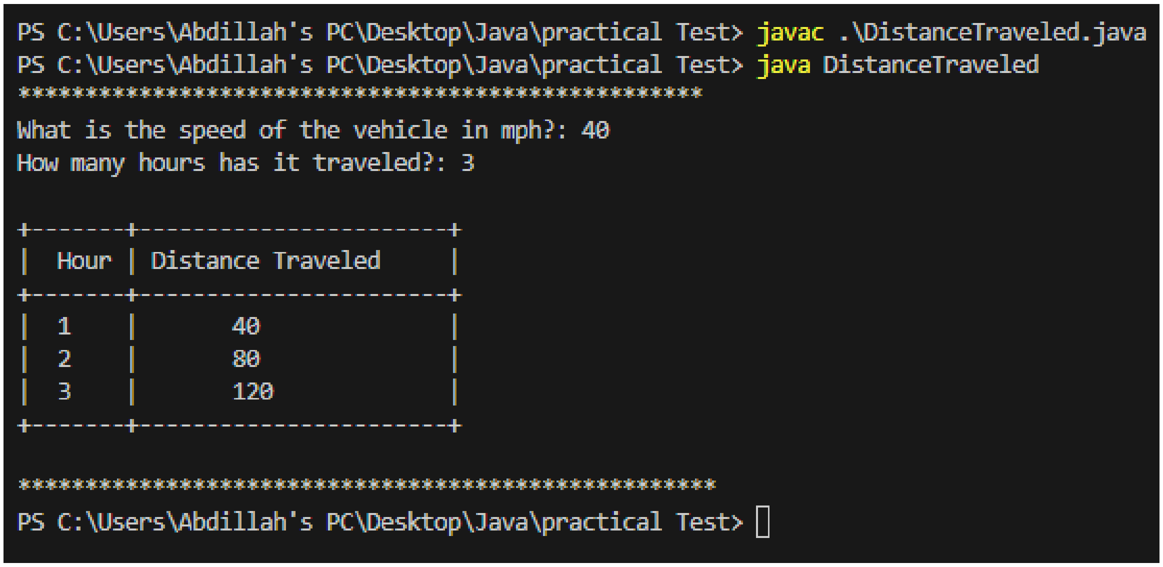
```
Hour    Distance Traveled
1        40
2        80
3        120
```

Program:



```java
import java.util.Scanner;
public class DistanceTraveled{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("***************************************************");

        System.out.print("What is the speed of the vehicle in mph?: ");
        int speed = sc.nextInt();

        System.out.print("How many hours has it traveled?: ");
        int time = sc.nextInt();

        System.out.println("\n+-------+-----------------------+");
        System.out.println("|  Hour\t| Distance Traveled \t|");
        System.out.println("+-------+-----------------------+");

        for(int t = 1; t <= time; t++){
            System.out.println("|  "+t+"\t|\t"+t * speed+"\t\t|");
        }

        System.out.println("+-------+-----------------------+");

        System.out.println("\n****************************************************");


    }
}

```

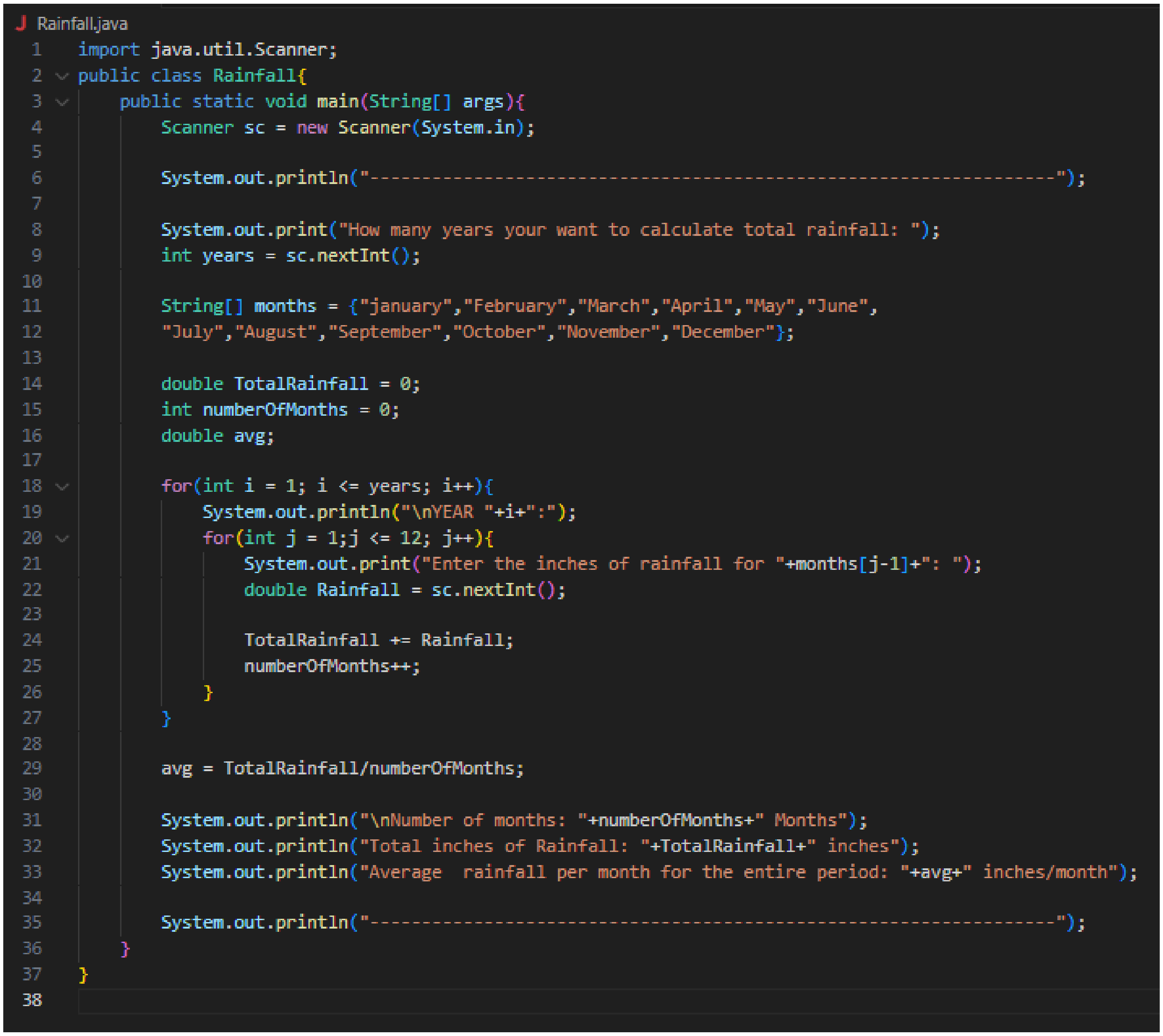
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\DistanceTraveled.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java DistanceTraveled
***************************************************
What is the speed of the vehicle in mph?: 40
How many hours has it traveled?: 3

+-------+-----------------------+
|  Hour | Distance Traveled     |
+-------+-----------------------+
|  1    |       40              |
|  2    |       80              |
|  3    |       120             |
+-------+-----------------------+

****************************************************
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

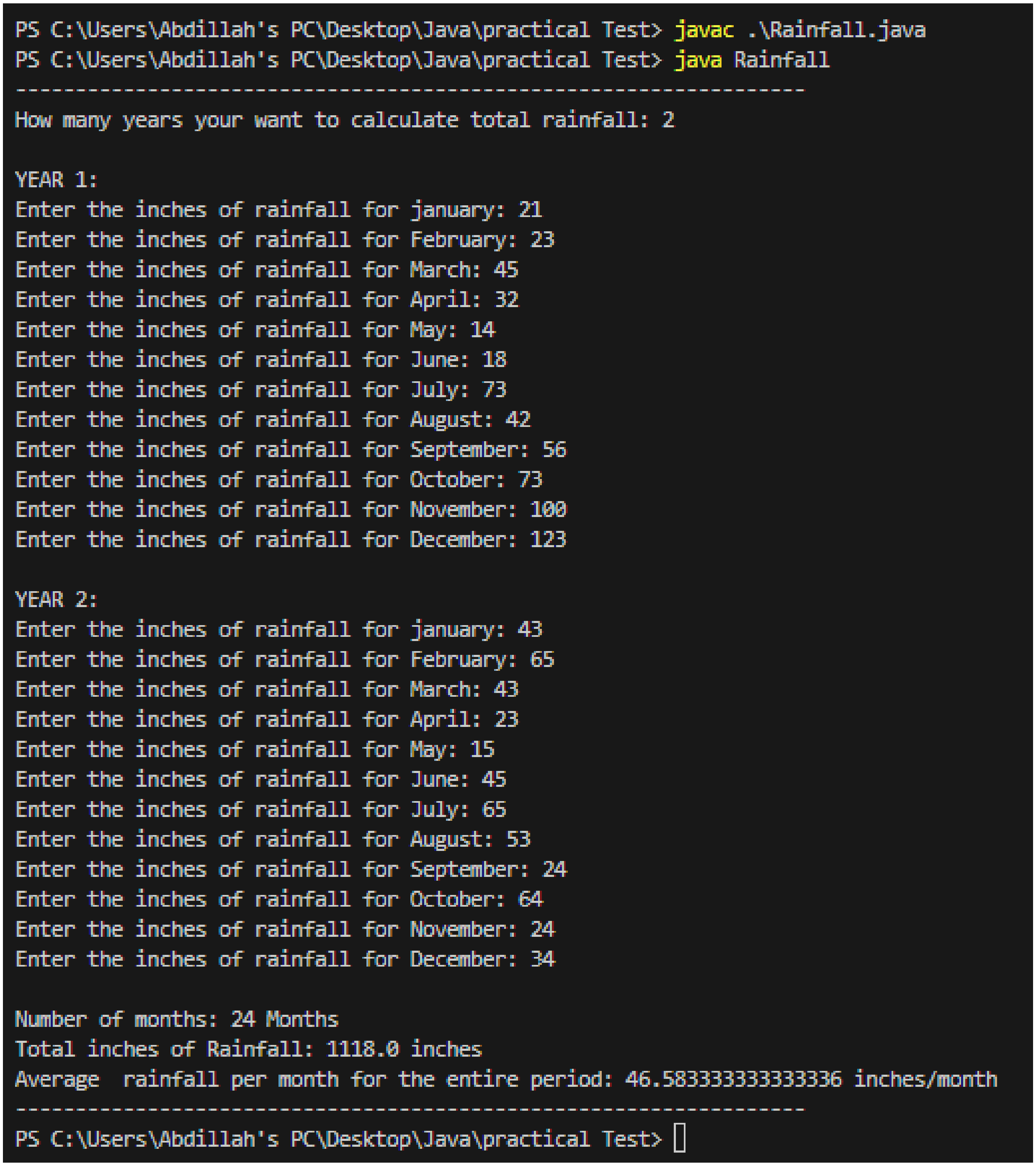
12. Write a program that uses nested loops to collect data and calculate the average rainfall over a period of years. The program should first ask for the number of years. The outer loop will iterate once for each year. The inner loop will iterate twelve times, once for each month. Each iteration of the inner loop will ask the user for the inches of rainfall for that month. After all iterations, the program should display the number of months, the total inches of rainfall, and the average rainfall per month for the entire period.

    Program:



```java
import java.util.Scanner;
public class Rainfall{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("-------------------------------------------------------------------");

        System.out.print("How many years your want to calculate total rainfall: ");
        int years = sc.nextInt();

        String[] months = {"january","February","March","April","May","June",
        "July","August","September","October","November","December"};

        double TotalRainfall = 0;
        int numberOfMonths = 0;
        double avg;

        for(int i = 1; i <= years; i++){
            System.out.println("\nYEAR "+i+":");
            for(int j = 1;j <= 12; j++){
                System.out.print("Enter the inches of rainfall for "+months[j-1]+": ");
                double Rainfall = sc.nextInt();

                TotalRainfall += Rainfall;
                numberOfMonths++;
            }
        }

        avg = TotalRainfall/numberOfMonths;

        System.out.println("\nNumber of months: "+numberOfMonths+" Months");
        System.out.println("Total inches of Rainfall: "+TotalRainfall+" inches");
        System.out.println("Average  rainfall per month for the entire period: "+avg+" inches/month");

        System.out.println("-------------------------------------------------------------------");
    }
}
```

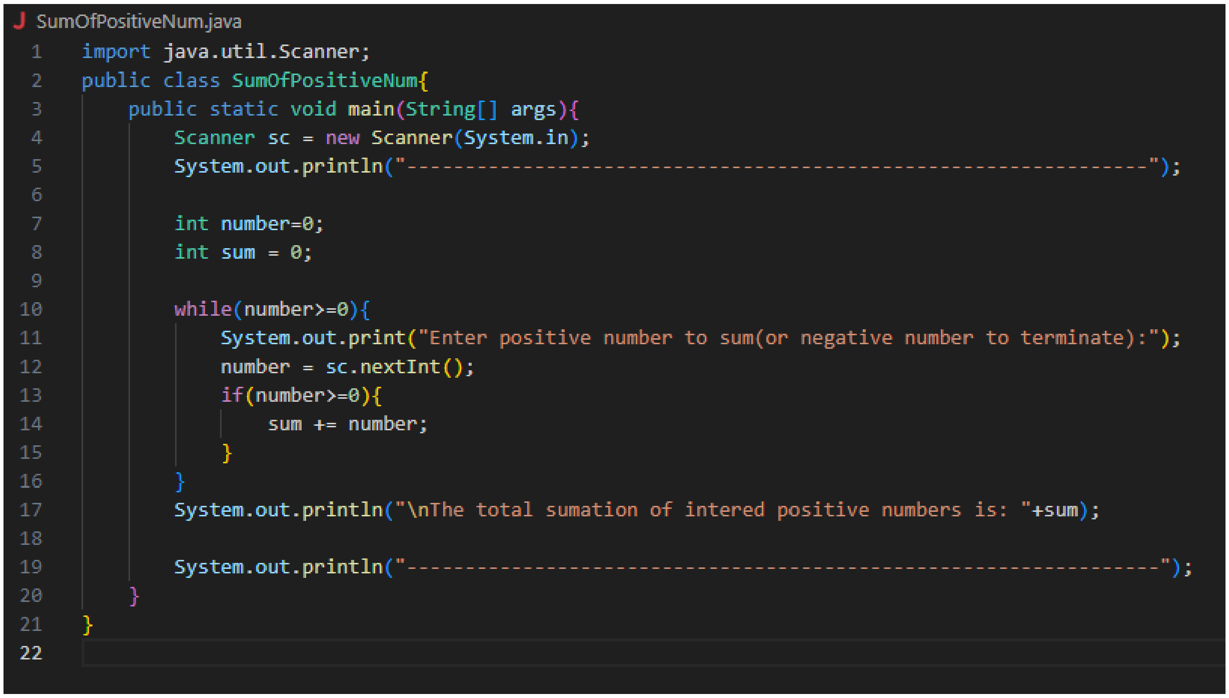
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\Rainfall.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java Rainfall
------------------------------------------------------------------
How many years your want to calculate total rainfall: 2

YEAR 1:
Enter the inches of rainfall for january: 21
Enter the inches of rainfall for February: 23
Enter the inches of rainfall for March: 45
Enter the inches of rainfall for April: 32
Enter the inches of rainfall for May: 14
Enter the inches of rainfall for June: 18
Enter the inches of rainfall for July: 73
Enter the inches of rainfall for August: 42
Enter the inches of rainfall for September: 56
Enter the inches of rainfall for October: 73
Enter the inches of rainfall for November: 100
Enter the inches of rainfall for December: 123

YEAR 2:
Enter the inches of rainfall for january: 43
Enter the inches of rainfall for February: 65
Enter the inches of rainfall for March: 43
Enter the inches of rainfall for April: 23
Enter the inches of rainfall for May: 15
Enter the inches of rainfall for June: 45
Enter the inches of rainfall for July: 65
Enter the inches of rainfall for August: 53
Enter the inches of rainfall for September: 24
Enter the inches of rainfall for October: 64
Enter the inches of rainfall for November: 24
Enter the inches of rainfall for December: 34

Number of months: 24 Months
Total inches of Rainfall: 1118.0 inches
Average  rainfall per month for the entire period: 46.583333333333336 inches/month
------------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

13. Write a program with a loop that asks the user to enter a series of positive numbers. The user should enter a negative number to signal the end of the series. After all the positive numbers have been entered, the program should display their sum.

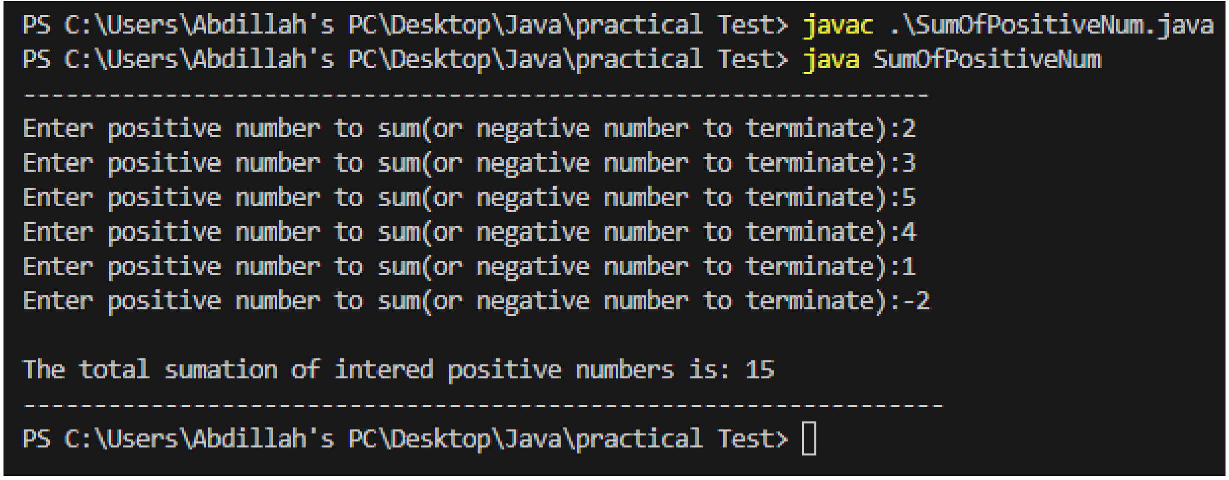
Program:



SumOfPositiveNum.java

```java
import java.util.Scanner;
public class SumOfPositiveNum{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("---------------------------------------------------------------");

        int number=0;
        int sum = 0;

        while(number>=0){
            System.out.print("Enter positive number to sum(or negative number to terminate):");
            number = sc.nextInt();
            if(number>=0){
                sum += number;
            }
        }
        System.out.println("\nThe total sumation of intered positive numbers is: "+sum);

        System.out.println("----------------------------------------------------------------");
    }
}

```

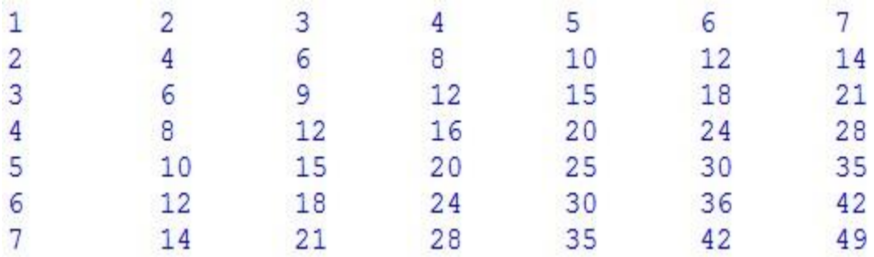
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\SumOfPositiveNum.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java SumOfPositiveNum
---------------------------------------------------------------
Enter positive number to sum(or negative number to terminate):2
Enter positive number to sum(or negative number to terminate):3
Enter positive number to sum(or negative number to terminate):5
Enter positive number to sum(or negative number to terminate):4
Enter positive number to sum(or negative number to terminate):1
Enter positive number to sum(or negative number to terminate):-2

The total sumation of intered positive numbers is: 15
----------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```

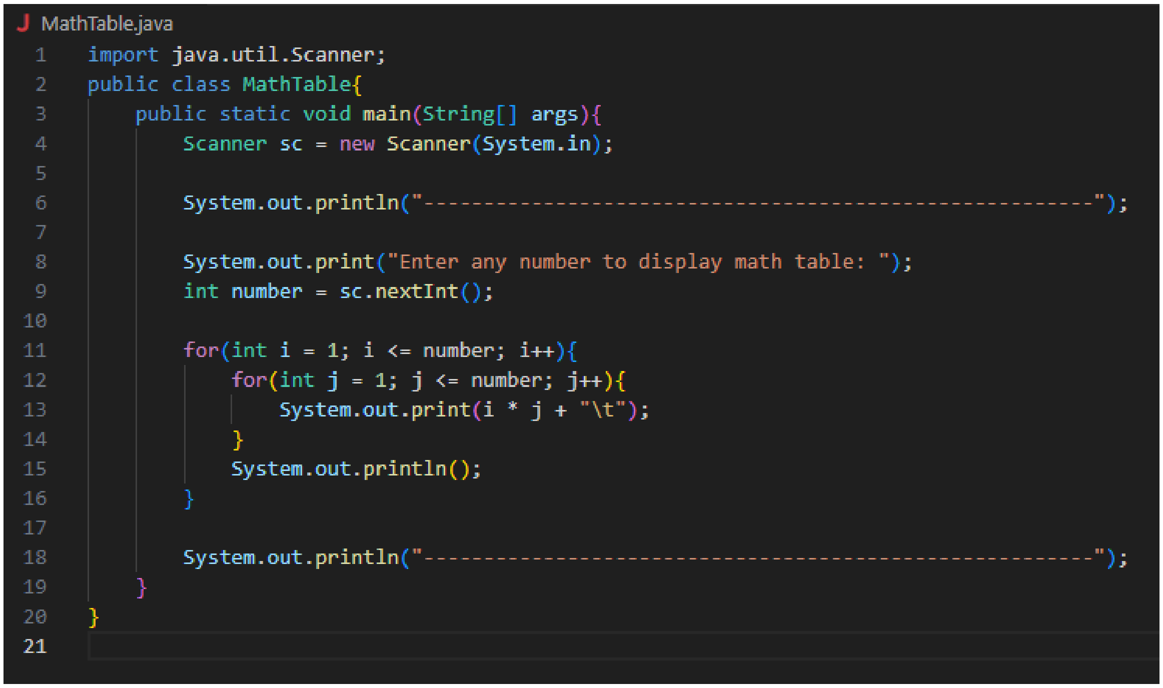
14. Write a program that gets a value from the user and then prints mathematical table as show below.

For example user input 7, it should print like this below



```
1       2       3       4       5       6       7
2       4       6       8       10      12      14
3       6       9       12      15      18      21
4       8       12      16      20      24      28
5       10      15      20      25      30      35
6       12      18      24      30      36      42
7       14      21      28      35      42      49
```

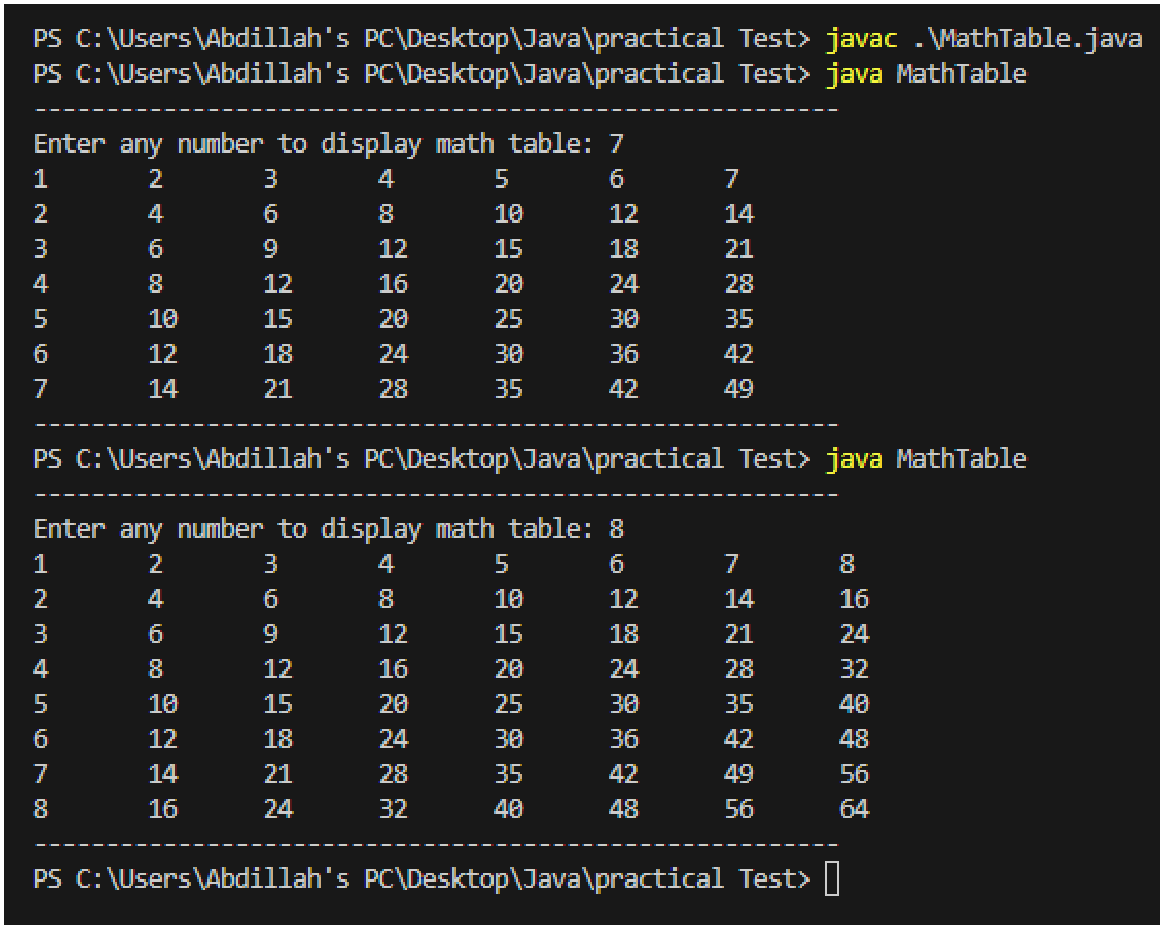
Program:



MathTable.java

```java
import java.util.Scanner;
public class MathTable{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("--------------------------------------------------------");

        System.out.print("Enter any number to display math table: ");
        int number = sc.nextInt();

        for(int i = 1; i <= number; i++){
            for(int j = 1; j <= number; j++){
                System.out.print(i * j + "\t");
            }
            System.out.println();
        }

        System.out.println("--------------------------------------------------------");
    }
}

```

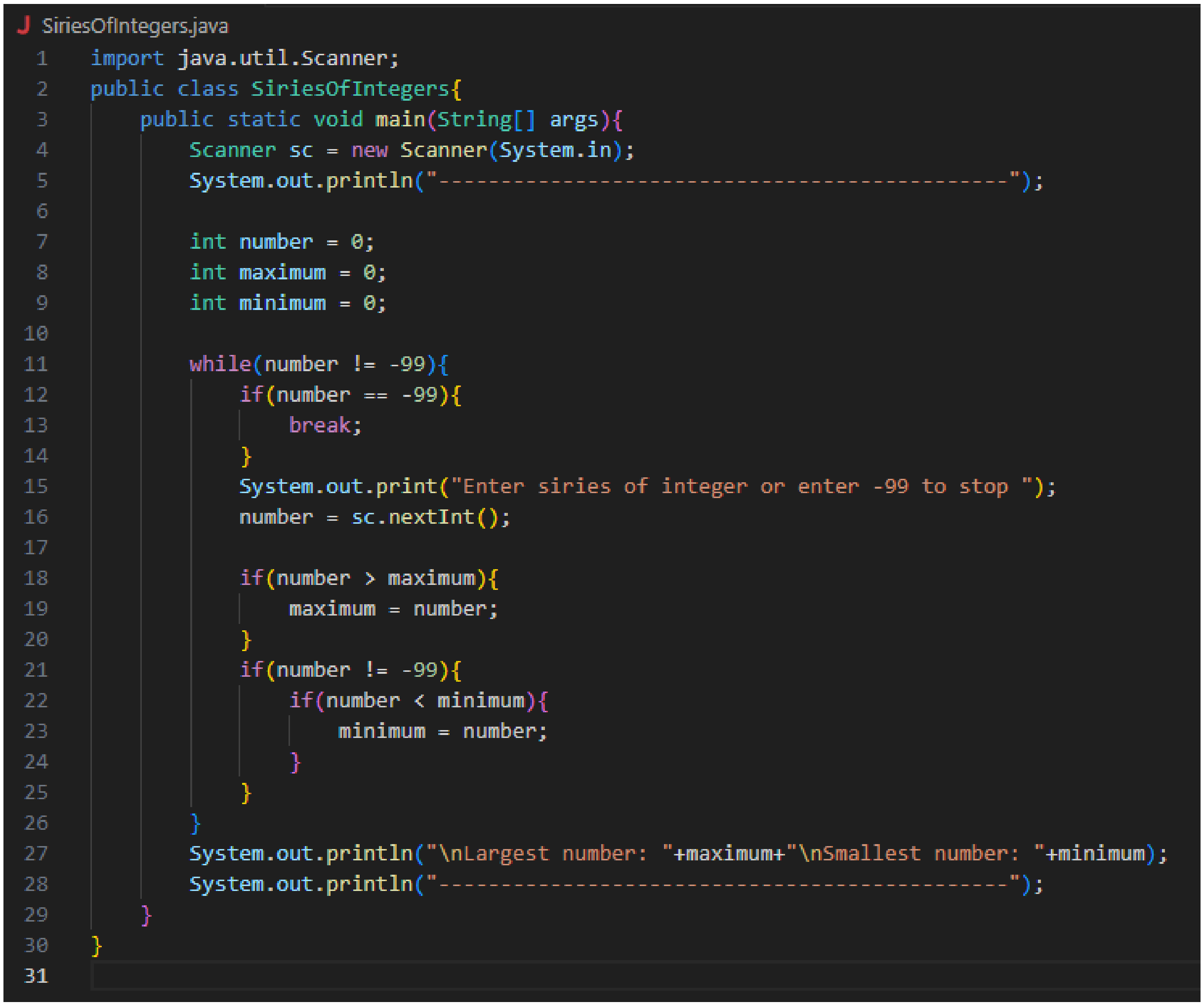
Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\MathTable.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java MathTable
--------------------------------------------------------
Enter any number to display math table: 7
1       2       3       4       5       6       7
2       4       6       8       10      12      14
3       6       9       12      15      18      21
4       8       12      16      20      24      28
5       10      15      20      25      30      35
6       12      18      24      30      36      42
7       14      21      28      35      42      49
--------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java MathTable
--------------------------------------------------------
Enter any number to display math table: 8
1       2       3       4       5       6       7       8
2       4       6       8       10      12      14      16
3       6       9       12      15      18      21      24
4       8       12      16      20      24      28      32
5       10      15      20      25      30      35      40
6       12      18      24      30      36      42      48
7       14      21      28      35      42      49      56
8       16      24      32      40      48      56      64
--------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test>
```

15. Write a program with a loop that lets the user enter a series of integers, followed by -99 to signal the end of the series. After all the numbers have been entered, the program should display the largest and smallest numbers entered. Do NOT use any build-in functions.

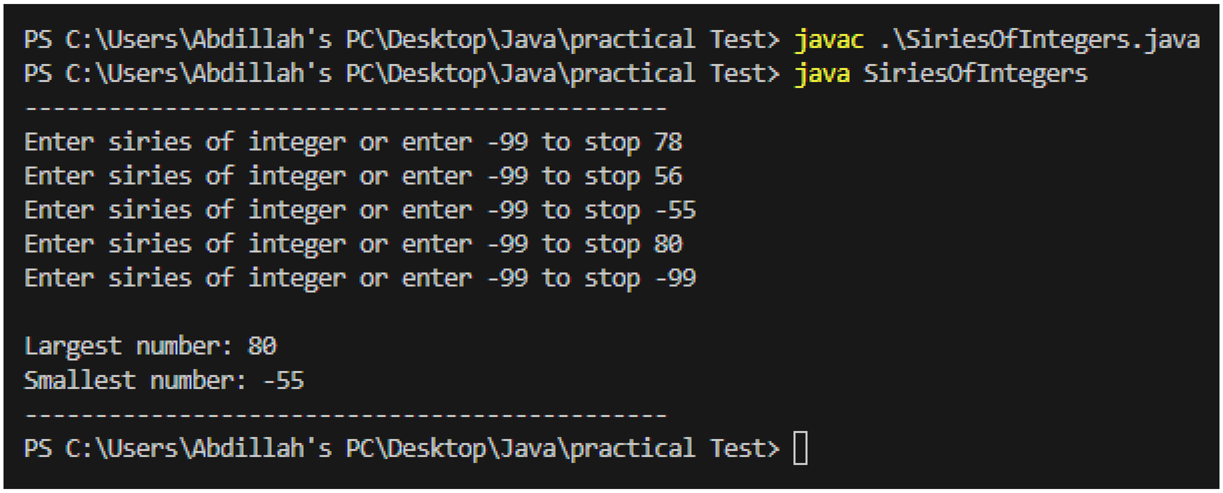
Program:



SiriesOfIntegers.java

```java
import java.util.Scanner;
public class SiriesOfIntegers{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("----------------------------------------------");

        int number = 0;
        int maximum = 0;
        int minimum = 0;

        while(number != -99){
            if(number == -99){
                break;
            }
            System.out.print("Enter siries of integer or enter -99 to stop ");
            number = sc.nextInt();

            if(number > maximum){
                maximum = number;
            }
            if(number != -99){
                if(number < minimum){
                    minimum = number;
                }
            }
        }
        System.out.println("\nLargest number: "+maximum+"\nSmallest number: "+minimum);
        System.out.println("----------------------------------------------");
    }
}

```

Output:



```
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> javac .\SiriesOfIntegers.java
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> java SiriesOfIntegers
----------------------------------------------
Enter siries of integer or enter -99 to stop 78
Enter siries of integer or enter -99 to stop 56
Enter siries of integer or enter -99 to stop -55
Enter siries of integer or enter -99 to stop 80
Enter siries of integer or enter -99 to stop -99

Largest number: 80
Smallest number: -55
----------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\practical Test> 
```