Name:

Noor-ul-Ain-Khan

Registration number:

FA25-BDS-039

Assignment:

Programming Fundamentals

Date:

04-October-2025

Submitted to:

Mr.Khurram Iqbal

Programming paradigms

- A programming paradigm refers to a style, way, or classification of programming

Imperative programming paradigm:

- The programming in which the programmer instructs the machine *how to change its state.*

Examples: C, C++, Java,etc.

It includes

- Procedural programming paradigm
- Object oriented programming
- Parallel processing approach

Declarative programming paradigm:

- The programmer states only what the result should look like, not how to obtain it.

Examples: SQL, Prolog etc.

- Here's a brief explaination of declarative programming paradigm
- Logic programming paradigm
- Functional programming
- Database processing approach
- **Programming languages** are used in order to solve problems
    - Structured/Procedural programming
    - Functional programming
    - Object oriented programming

Here's a brief description of languages

Structured Programming

- Kind of imperative programming
- Control flow is defined by nested loops, conditionals, and subroutines
- Sequence , Selection and Repetition

Procedural Programming

- A language derived from structure programming and based on the use of call procedures
- Procedures are the functions, routines, or subroutines that specify the computational steps that need to be performed
- A program is designed using a top-down approach in procedural programming
- C/C++ starts execution from main() top function
- Examples: COBOL, BASIC etc.

Object Oriented Programming (OOP)

- Object-oriented programming is about creating objects that contain both data and methods
- Classes and objects are the two main aspects of object-oriented programming
  - Examples: C++, C#, Java etc.

Functional Programming (FP)

- Program is composed of series of short functions.
- All code is within the functions
- All variables are scoped to function

- Examples: C++, Python, Java Script etc.
- **it includes**
- First-class functions
- Higher-order function
- Pure functions and side-effects

Syntax errors

The mistakes in java syntax or grammer preventing java code from compiling.

1. **Missing semicolon**

   Example:

   ```
   int x = 5
   ```
2. **Missing closing braces**

   Example

   ```
   : public class main
           public static void main(String[] args)
   ```
3. **Wrong spelling**

   Example:

   ```
   improt java.uitl.scanwen;
   ```

   Logical errors

   These errors cause incorrect result.

1. **Wrong formula**

   Examples:

   ```
   double area = 2 * 3.14 * radius;
   ```
2. **Assigning not comparing**

   Examples:

   ```
   if (x=5) {
   ```
3. **Off by one error in loop**

   Examples:

   ```
   for (int i = 0; i <= 10; i++) {
   sum += i;
   ```

Runtime errors

1. **Division by zero**

   Examples:

   ```
   int result = 3/0;
   ```

2. **Null pointer access**

   Examples:

   ```
   String text = null;
   ```

3. **Array out of bounds**

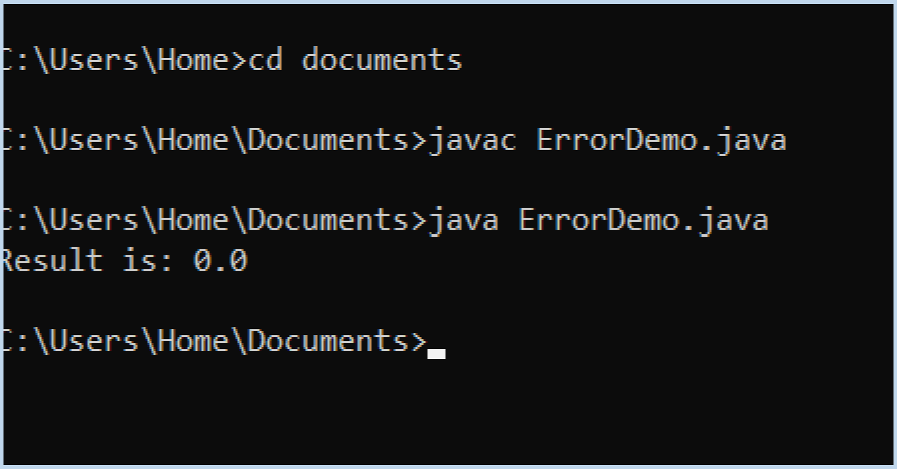
   Examples:

   ```
           int[] numbers = {1, 2, 3};
   System.out.println(numbers[5]);
   ```

Finding errors

Program 1 errors

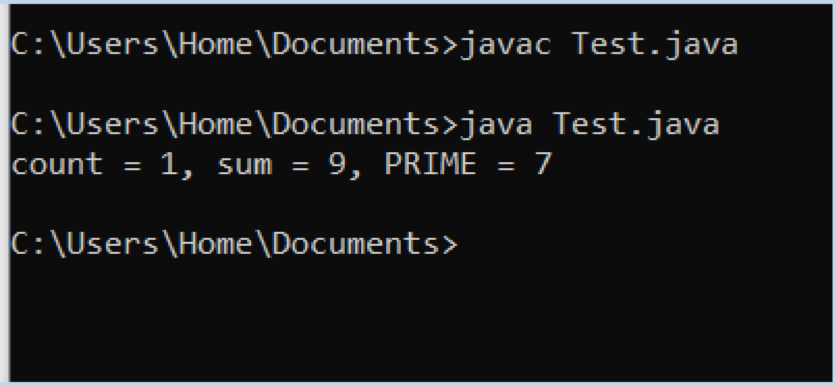
Error	**Error type**	**Correction**
Number = “ten”;	Syntax error	Number = 10;
float pi = 3,1416;	Syntax error	float pi = 3.1416;
double result == 0;	Syntax error	double result = 0;
if (number = 5)	Syntax error	if (number == 5)
(“Result is:” + result;	Syntax error	(“Result is:” + result);



```
C:\Users\Home>cd documents

C:\Users\Home\Documents>javac ErrorDemo.java

C:\Users\Home\Documents>java ErrorDemo.java
Result is: 0.0

C:\Users\Home\Documents>
```

Program 2 errors

Errors	Type	Correction
count = 1;	Syntax errors	int count = 1;
x := 25.67;	Syntax errors	x = 25.67;
newNum = count * ONE + 2;	Syntax errors	Declaration
sum + count = sum;	Syntax errors	sum = count + sum;
PRIME = “ + Prime	Syntax errors	prime



```
C:\Users\Home\Documents>javac Test.java

C:\Users\Home\Documents>java Test.java
count = 1, sum = 9, PRIME = 7

C:\Users\Home\Documents>
```

Comments

Single line	Multi line
// first number	/*This program will calculate product of three numbers */
// second number	
// third number	
// product of numbers	

Special symbols

Symbol	Purpose
{ }	Denotes the beginning and end of a block

;	Terminates a statement
()	Used in method declarations and method calls

Reserved words

Reserved words	purposes
public	Access modifier
class	Used to declare a class
int	Data type for integers

Identifiers

Identifier	**purpose**
“ “	Predefined class in Java API
out	Predefined output stream object of System
println	Predefined method to print output

User-defined Identifiers

Identifiers	**Purposes**
Product	Name of the class
num1, num2, num3	Variable names defined by user
result	User-defined variable for storing product

Standard input output stream object

Name	**purposes**
System.in	Used for input
System.out	Used for output

Java statements

1.Java statement 1

```
        String name = "Java";
```

2.Java statement 2

```
        boolean isPassed = true;
```

3.Java statement 3

```
        int a = 4, b = 5;
        int product = a * b;
```

4.Java statement 4

```
        float temperature = 36.6f;
```

5.Java statement 5

```
        int num1 = 10, num2 = 3;
        int remainder = num1 % num2;
```

6.Java statement 6

```
        double p = 4.0, q = 5.0;
System.out.println("p: " + p + ", q: " + q + ", result: " + ((p * q) / 2));
```

7.Java statement 7

```
        char symbol = '#';
```

8.Java statement 8

```
        int mark1, mark2, mark3;
```

9.Java statement 9

```
        int score = 87;
        double resultScore = score;
```

10.Java statement 10

```
        double radius = 7.0;
        double area = Math.PI * radius * radius;
```

so the above values declare and initiliaze were for examples now we will trace the program.

Tracing variables

Statement 1

```
        a = (++b) * 2 + (c--);
// ++b = 7 (b becomes 7)
// (c--) = 7, then c becomes 6
// a = 7 * 2 + 7 = 14 + 7 = 21
```

Solution

a = 21 , b = 7 and c = 6 and d = 2

statement 2

```
c = (a++) - (--d) + b;
// a++ = 21 (use 21, then a becomes 22)
// --d = 1 (d becomes 1)
// c = 21 - 1 + 7 = 27
```

Solution

a = 22, b = 7, c = 27, d = 1

statement 3

```
b = (d--) + (c++) * ++a;
// d-- = 1 (use 1, then d becomes 0)
// c++ = 27 (use 27, then c becomes 28)
// ++a = 23 (a becomes 23)
// b = 1 + 27 * 23 = 1 + 621 = 622
```

Solution

a = 23, b = 622, c = 28, d = 0

statement 4

```
d = (--a) + (b++) - (c--);
// --a = 22
// b++ = 622 (then b becomes 623)
// c-- = 28 (then c becomes 27)
// d = 22 + 622 - 28 = 616
```

Solution

a = 22, b = 623, c = 27, d = 616

Statements	**a**	**b**	**c**	**d**
a = (++b) * 2 + (c--);	21	7	6	2
c = (a++) - (--d) + b;	22	7	27	1

b = (d--) + (c++) * ++a;	23	622	28	0
d = (--a) + (b++) - (c--);	22	623	27	616

Type conversion

Statement 1

```
sum = a + b + (int)c;
// (int)c = 14
// sum = 3 + 5 + 14 = 22
```

Solution

a = 3, b = 5, c = 14.1, sum = 22

Statement 2

```
c /= a;
// c = 14.1 / 3 = 4.7
```

Solution

c = 4.7

Statement 3

```
b += (int)c - a;

// (int)c = 4

// b += 4 - 3 => b = 5 + 1 = 6
```

Solution

b = 6

Statement 4

a *= 2 * b + (int)c;
// 2 * b = 12
// (int)c = 4
// a *= 12 + 4 = 16 => a = 3 * 16 = 48

Solution

48

Statement	**a**	**b**	**c**	**Sum**
sum = a + b + (int) c;	3	5	14.1	22
c /= a;	3	5	4.7	22
b += (int) c - a;	3	6	4.7	22
a *= 2 * b + (int) c;	48	6	4.7	22

Final values:

a = 48

b = 6

c = 4.7

sum = 22

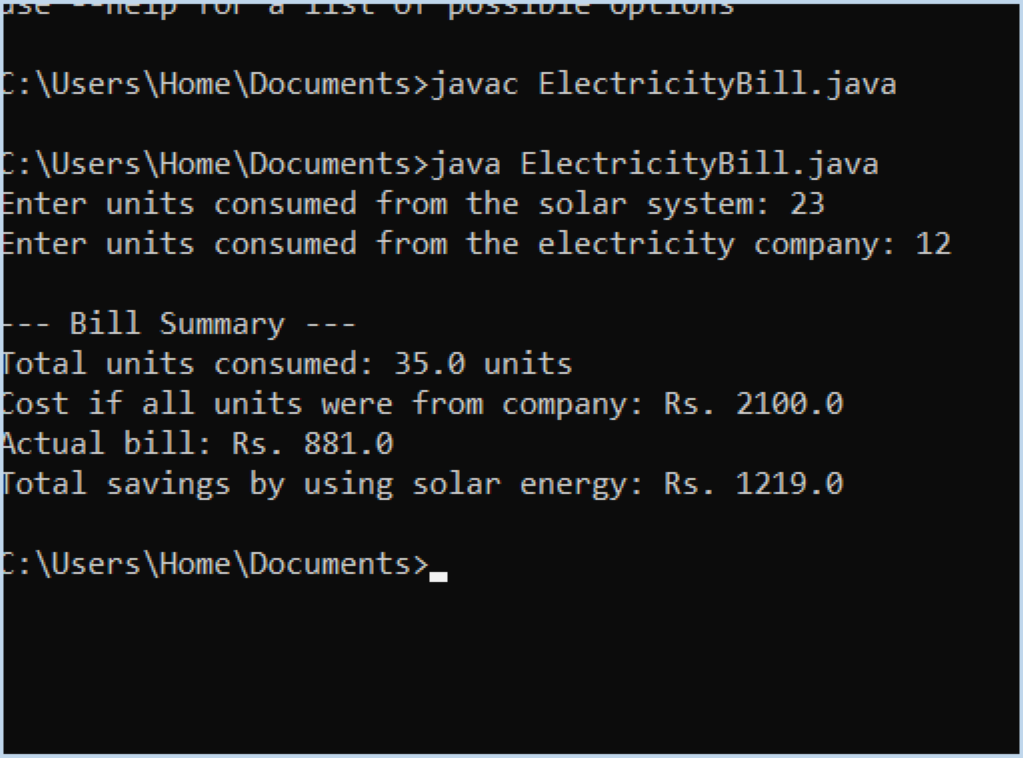
Question 5

Ali lives in a village where he uses both solar energy and electricity from the company. During the day, he consumes electricity generated from his solar system, which costs him only Rs. 7 per unit. After 5 PM, he uses electricity from the company, which charges Rs. 60 per unit. Ali wants to calculate how much money he is saving by using solar energy. Create a program that asks for the number of units consumed from the solar system and the number of units consumed from the electricity company. The program should then calculate the total bill if all units were taken from the company, the actual bill using both solar and company electricity, and the total savings Ali makes by using solar power.

java code

```
import java.util.Scanner;

public class ElectricityBill {
  public static void main(String[] args) {
    // Rates per unit
    final double SOLAR_RATE = 7.0;
    final double COMPANY_RATE = 60.0;

    // Scanner to get user input
    Scanner scanner = new Scanner(System.in);

    // Input from user
    System.out.print("Enter units consumed from the solar system: ");
    double solarUnits = scanner.nextDouble();

    System.out.print("Enter units consumed from the electricity company: ");
    double companyUnits = scanner.nextDouble();

    // Total units consumed
    double totalUnits = solarUnits + companyUnits;
```

```
        // Calculations
        double totalCompanyCost = totalUnits * COMPANY_RATE;
        double actualCost = (solarUnits * SOLAR_RATE) + (companyUnits * COMPANY_RATE);
        double savings = totalCompanyCost - actualCost;

        // Output
        System.out.println("\n--- Bill Summary ---");
        System.out.println("Total units consumed: " + totalUnits + " units");
        System.out.println("Cost if all units were from company: Rs. " + totalCompanyCost);
        System.out.println("Actual bill: Rs. " + actualCost);
        System.out.println("Total savings by using solar energy: Rs. " + savings);

        // Close the scanner
        scanner.close();
    }
}
```



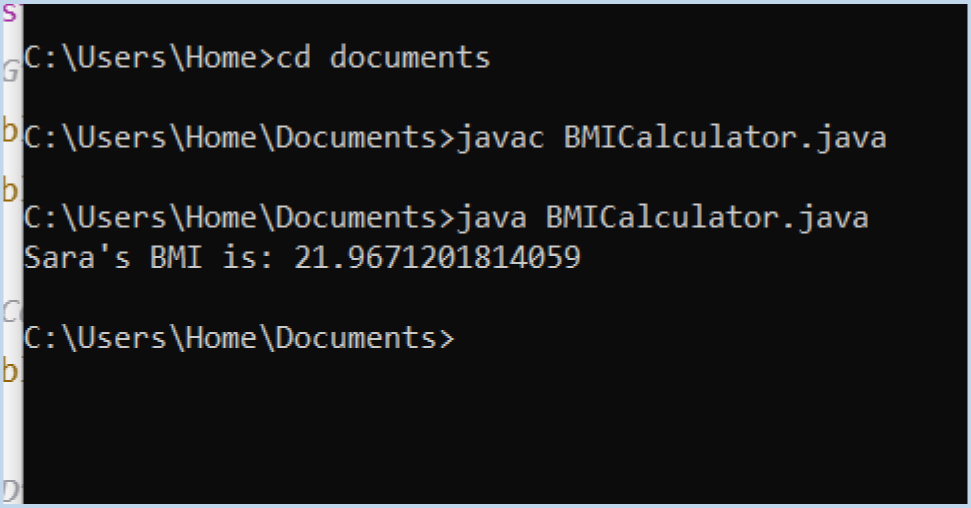
```
use --help for a list of possible options

C:\Users\Home\Documents>javac ElectricityBill.java

C:\Users\Home\Documents>java ElectricityBill.java
Enter units consumed from the solar system: 23
Enter units consumed from the electricity company: 12

--- Bill Summary ---
Total units consumed: 35.0 units
Cost if all units were from company: Rs. 2100.0
Actual bill: Rs. 881.0
Total savings by using solar energy: Rs. 1219.0

C:\Users\Home\Documents>
```

Question 6

Sara visits her doctor for a regular health check-up, and the doctor asks her to calculate her Body Mass Index (BMI) to monitor her fitness. Sara's weight is 62 kilograms, and her height is 1.68 meters. The formula for BMI is the weight divided by the square of the height. Create a program that calculates Sara's BMI and displays the result

Java code

```
public class BMICalculator {

    public static void main(String[] args) {

        // Given data

        double weight = 62.0;     // in kilograms

        double height = 1.68;     // in meters



        // Calculate BMI

        double bmi = weight / (height * height);
        // Display result

        System.out.println("Sara's BMI is: " + bmi);

    }

}
```



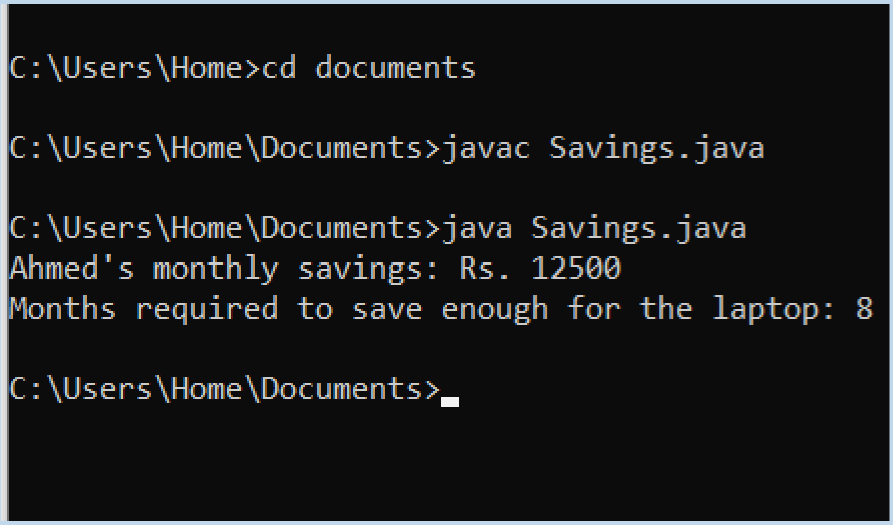
```
C:\Users\Home>cd documents

C:\Users\Home\Documents>javac BMICalculator.java

C:\Users\Home\Documents>java BMICalculator.java
Sara's BMI is: 21.9671201814059

C:\Users\Home\Documents>
```

Question 7

Ahmed works in a private company and earns Rs. 50,000 each month. His monthly expenses, however, are Rs. 37,500. Ahmed plans to purchase a new laptop that costs Rs. 100,000. He wants to calculate how long it will take to save enough money. Create a program that calculates Ahmed’s monthly savings and determines how many months are required for him to save enough money to buy the laptop

Java code

```
public class Savings {

    public static void main(String[] args) {

        // Given values

        int monthlyIncome = 50000;

        int monthlyExpenses = 37500;

        int laptopPrice = 100000;
        // Calculate monthly savings
        int monthlySavings = monthlyIncome - monthlyExpenses;

        // Calculate required months
        int monthsRequired = (int) Math.ceil((double) laptopPrice / monthlySavings);

        // Output results
        System.out.println("Ahmed's monthly savings: Rs. " + monthlySavings);
        System.out.println("Months required to save enough for the laptop: " + monthsRequired);
    }
}
```



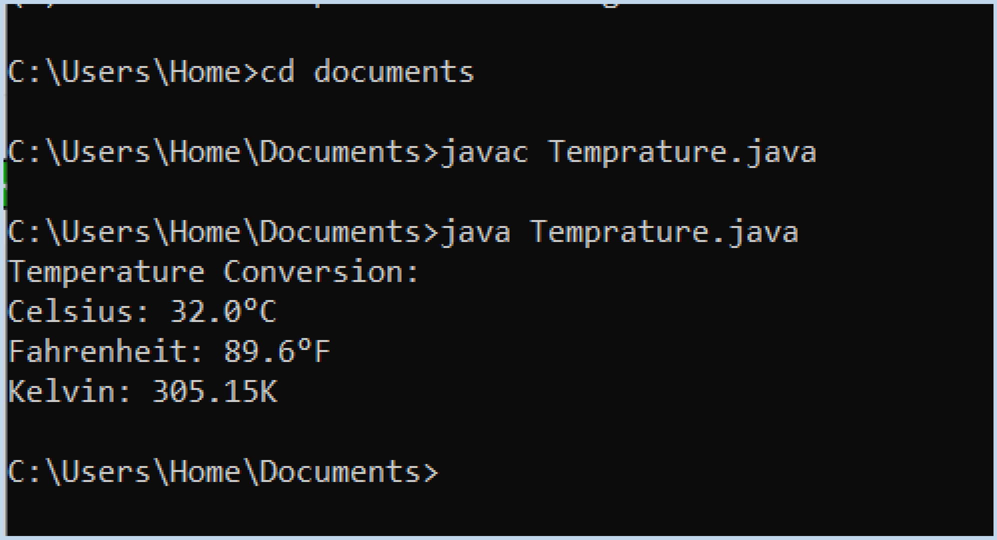
```
C:\Users\Home>cd documents

C:\Users\Home\Documents>javac Savings.java

C:\Users\Home\Documents>java Savings.java
Ahmed's monthly savings: Rs. 12500
Months required to save enough for the laptop: 8

C:\Users\Home\Documents>_
```

Question 8

The weather station in Lahore has reported today’s temperature as 32 degrees Celsius. For preparing an international weather report, the same temperature also needs to be expressed in Fahrenheit and Kelvin. The formulas to use are Fahrenheit = (9/5 × Celsius) + 32 and Kelvin = Celsius + 273.15. Create a program that converts 32 degrees Celsius into Fahrenheit and Kelvin and displays the converted values

Java code

```
public class Temprature {

   public static void main(String[] args) {

      // Temperature in Celsius

      double celsius = 32;


      // Convert to Fahrenheit

      double fahrenheit = (9.0 / 5.0 * celsius) + 32;


      // Convert to Kelvin

      double kelvin = celsius + 273.15;


      // Display the results

      System.out.println("Temperature Conversion:");

      System.out.println("Celsius: " + celsius + "°C");

      System.out.println("Fahrenheit: " + fahrenheit + "°F");

      System.out.println("Kelvin: " + kelvin + "K");

   }

}
```



```
C:\Users\Home>cd documents

C:\Users\Home\Documents>javac Temprature.java

C:\Users\Home\Documents>java Temprature.java
Temperature Conversion:
Celsius: 32.0°C
Fahrenheit: 89.6°F
Kelvin: 305.15K

C:\Users\Home\Documents>
```

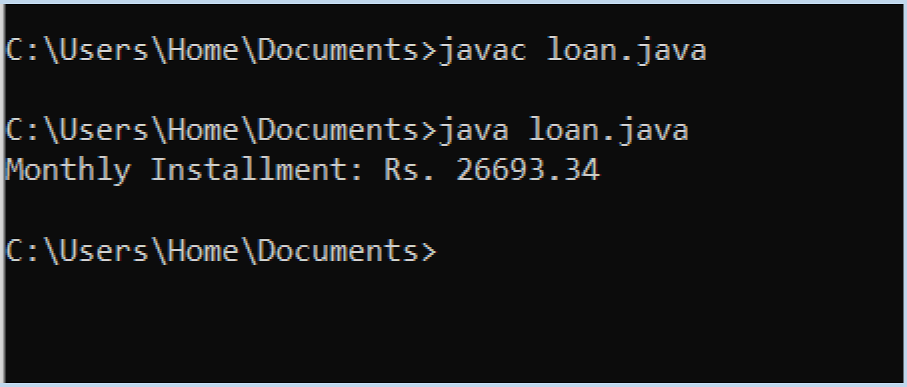
Question 9

. Hassan has purchased a car by taking a loan of Rs. 1,200,000 from the bank. The loan is to be repaid in 5 years with an annual interest rate of 12%. Hassan wants to know his monthly installment amount so he can plan his expenses. The formula for calculating the monthly installment is Payment = (P × r) / (1 – (1 + r)^(-n)), where P is the loan amount, r is the monthly interest rate (annual rate divided by 12), and n is the total number of months. Create a program that calculates Hassan’s monthly car loan installment.

Java code

```
public class LoanCalculator {

  public static void main(String[] args) {

    // Given values

    double loanAmount = 1200000; // P

    double annualInterestRate = 12; // in percent

    int loanTermYears = 5;



    // Calculated values

    double monthlyInterestRate = annualInterestRate / 12 / 100; // r

    int totalMonths = loanTermYears * 12; // n
```

```
        // Monthly installment calculation

        double monthlyPayment = (loanAmount * monthlyInterestRate) /

                (1 - Math.pow(1 + monthlyInterestRate, -totalMonths));


        // Display result

        System.out.printf("Monthly Installment: Rs. %.2f\n", monthlyPayment);

    }

}
```



```
C:\Users\Home\Documents>javac loan.java

C:\Users\Home\Documents>java loan.java
Monthly Installment: Rs. 26693.34

C:\Users\Home\Documents>
```

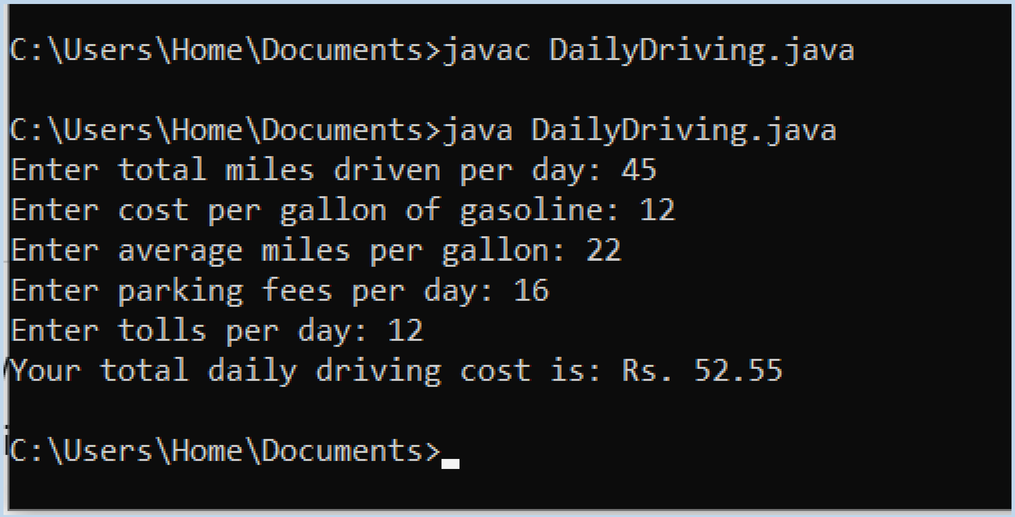
Question – 10:

Research several car-pooling websites. Create an application that calculates your daily driving cost, so that you can estimate how much money could be saved by carpooling, which also has other advantages such as reducing carbon emissions and reducing traffic congestion. The application should input the following information and display the user's cost per day of driving to work: a) Total miles driven per day. b) Cost per gallon of gasoline. c) Average miles per gallon. d) Parking fees per day. e) Tolls per day.

Java code

```
import java.util.Scanner;
```

```
public class DailyDrivingCostCalculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);


        // Input values
        System.out.print("Enter total miles driven per day: ");
        double milesPerDay = scanner.nextDouble();


        System.out.print("Enter cost per gallon of gasoline: ");
        double costPerGallon = scanner.nextDouble();


        System.out.print("Enter average miles per gallon: ");
        double milesPerGallon = scanner.nextDouble();


        System.out.print("Enter parking fees per day: ");
        double parkingFees = scanner.nextDouble();


        System.out.print("Enter tolls per day: ");
        double tolls = scanner.nextDouble();


        // Calculate cost of gas used
        double fuelCost = (milesPerDay / milesPerGallon) * costPerGallon;


        // Calculate total daily cost
        double totalDailyCost = fuelCost + parkingFees + tolls;


        // Output result

        System.out.printf("Your total daily driving cost is: Rs. %.2f\n", totalDailyCost);


        scanner.close();

    }

}
```



```
C:\Users\Home\Documents>javac DailyDriving.java

C:\Users\Home\Documents>java DailyDriving.java
Enter total miles driven per day: 45
Enter cost per gallon of gasoline: 12
Enter average miles per gallon: 22
Enter parking fees per day: 16
Enter tolls per day: 12
Your total daily driving cost is: Rs. 52.55

C:\Users\Home\Documents>_
```

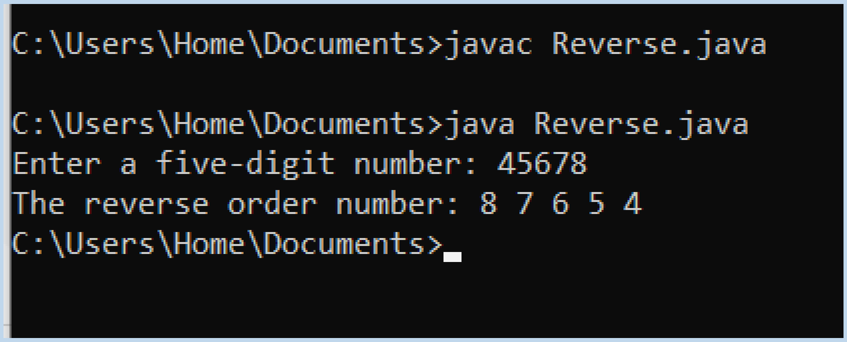
Question – 11:

Write an application that inputs one number consisting of five digits from the user and show the number in reverse order. Note the reverse order must also be a number. E.g. 93324

The reverse order number: 4 2 3 3 9

Java code

```
import java.util.Scanner;

public class Reverse {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);
        // Input (assumes valid five-digit number)
        System.out.print("Enter a five-digit number: ");
        int number = input.nextInt();

        // Manually extract digits
        int d1 = number % 10;
        int d2 = (number / 10) % 10;
        int d3 = (number / 100) % 10;
        int d4 = (number / 1000) % 10;
        int d5 = number / 10000;

        // Output reversed digits with spaces
        System.out.println("The reverse order number: " + d1 + " " + d2 + " " + d3 + " " + d4 + " " + d5);

        input.close();
    }
}
```



```
C:\Users\Home\Documents>javac Reverse.java

C:\Users\Home\Documents>java Reverse.java
Enter a five-digit number: 45678
The reverse order number: 8 7 6 5 4
C:\Users\Home\Documents>_
```

Completed