1.Employee Management System We have a list of employees with (name, age, salary).Your task is to write a program to find the average salary of employees older than 30.

```
package task;
public class Employe {

	public static void main(String[] args) {

		String[] names = {"Vignesh","Dinesh", "Pushparaj", "Santhosh", "Ilakkiya"};
		int[] ages = {31,35, 28, 32, 40};
		double[] salaries = {8000,50000, 60000, 70000, 55000};


		double totalSalary = 0;
		int count = 0;

		for (int i = 0; i < names.length; i++) {
		   if (ages[i] > 30) {
		      totalSalary += salaries[i];
		      count++;
		   }
		}

		double averageSalary = count > 0 ? totalSalary / count : 0;

		System.out.println("Total Employees: " + names.length);
		System.out.println("Employees older than 30: " + count);
		System.out.println("Total Salary of Employees older than 30: " + totalSalary);
		System.out.println("Average Salary of Employees older than 30: " + averageSalary);
	    }
	}
```

Output:

Total Employees: 5
Employees older than 30: 4
Total Salary of Employees older than 30: 183000.0
Average Salary of Employees older than 30: 45750.0

2.Student Grades System There is a list of students with their (name, grades). We have to find the name of the student who has the highest grade.

```
package task1;

class Student {
    public void student() {
        String[] names = {"Ilakkiya", "Nes", "Sadhana", "Lavanya",
"Viji", "Raj", "Keerathana"};
        int[] grades = {100, 90, 80, 100, 60, 90, 70};

        // Grade values
        int highestGrade = findHighestGrade(grades);

        System.out.println("Students with the highest grade (" +
highestGrade + "):");
        for (int i = 0; i < grades.length; ++i) {
            if (grades[i] == highestGrade) {
                System.out.println(names[i]);
            }
        }
    }

    private int findHighestGrade(int[] grades) {
        int highestGrade = grades[0];

        for (int i = 1; i < grades.length; ++i) {
            if (grades[i] > highestGrade) {
```

```
                highestGrade = grades[i];
            }
        }

        return highestGrade;
    }
}

public class Students {
    public static void main(String[] args) {
        Student s = new Student();
        s.student();
    }
}
```

Output:

```
Students with the highest grade (100):
Ilakkiya
Lavanya
```

3. Product Inventory Management Suppose in a Product Inventory We have a list of products with (name,price, quantity). You have to calculate the total cost of all products which are in stock.

```
package task;

class Product1 {
    String name;
    double price;
    int quantity;

    public Product1(String name, double price, int quantity) {
        this.name = name;
        this.price = price;
        this.quantity = quantity;
    }
```

```
    public double getTotalCost() {
        return price * quantity;
    }
}

public class Product {
    public static void main(String[] args) {

        String[] productNames = {"Product1", "Product2", "Product3"};
        double[] productPrices = {10.0, 5.0, 8.0};
        int[] productQuantities = {5, 10, 7};

        double totalCost = 0;

        for (int i = 0; i < productNames.length; i++) {
            Product1 product = new Product1(productNames[i],
productPrices[i], productQuantities[i]);
            totalCost += product.getTotalCost();
        }

        System.out.println("Total cost of products in stock: $" +
totalCost);
    }
}
```

Output:

Total cost of products in stock: $156.0

Part 2:

4. Java Program to generate Nth Fibonacci Number using Iteration.

```
package task;

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

```
        int n = 10;

        for (int i = 0; i < n; i++) {
            System.out.print(fib(i) + " ");
        }
    }

    public static int fib(int n) {
        if (n <= 1)
            return n;

        int fib1 = 0;
        int fib2 = 1;

        for (int i = 2; i <= n; i++) {
            int fibonacci = fib1 + fib2;
            fib1 = fib2;
            fib2 = fibonacci;
        }

        return fib2;
    }
}
```

Output:
0 1 1 2 3 5 8 13 21 34

5. Java Program to Count the Number of Digits in a Number.

```
package task;
import java.util.Scanner;
public class Count {

        public static void main(String[] args) {
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter a number: ");
```

```
            long number = scanner.nextLong();
            scanner.close();

            int count = count(number);
            System.out.println("Number of digits in the given number:
" + count);
        }

        public static int count(long number) {
            int count = 0;

            number = Math.abs(number);

            while (number != 0) {
                number /= 10;
                count++;
            }

            return count;
        }
    }
```

Output:

Enter a number: 6876068858686
Number of digits in the given number: 13

6. Java Code for Calculating Frequency of a Digit D in a Number N.

```
package task;
import java.util.Scanner;

public class Frequent {

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

```
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        long number = scanner.nextLong();
        System.out.print("Enter the digit to count: ");
        int digit = scanner.nextInt();
        scanner.close();

        int frequency = frequency(number, digit);
        System.out.println("Frequency of digit " + digit + " in the
given number: " + frequency);
    }

    public static int frequency(long number, int digit) {
        int frequency = 0;

        number = Math.abs(number);

        while (number > 0) {

            int lastDigit = (int) (number % 10);

            if (lastDigit == digit) {
                frequency++;
            }

            number /= 10;
        }

        return frequency;
    }
}
```

Output:

```
Enter a number: 46768609
Enter the digit to count: 8
Frequency of digit 8 in the given number: 1
```

7. Write a program in Java to Toggle the case of every character of a string. For instance, if the input

string is “ApPLe”, the output should be “aPplE”.

```
package task;

public class Arrange {

        public static void main(String[] args) {
            String inputString = "ApPLe";
            String result = arrange(inputString);
            System.out.println("Toggled case: " + result);
        }

        public static String arrange(String str) {
            StringBuffer re = new StringBuffer();

            for (char c : str.toCharArray()) {
                if (Character.isUpperCase(c)) {
                    re.append(Character.toLowerCase(c));
                } else if (Character.isLowerCase(c)) {
                    re.append(Character.toUpperCase(c));
                } else {
                    re.append(c);
                }
            }

            return re.toString();
        }
    }
```

Output:

Toggled case: aPplE

8. Write a program in Java to prove that the strings are immutable in Java.

```
package str;

public class Blank {

	public static void main(String[] args) {
		String s1 = new String("ILAKKIYA");

		String s2=s1.toLowerCase();
		System.out.println(s2);
		System.out.println(s1);

	}

}
```

Output:
ilakkiya
ILAKKIYA

String is a immutable because s1 cannot change only edit.

9. Write a program in Java to Reverse an Array without using extra space.

```
package str;

public class Reverse1 {
	public static void main(String[] args) {
	   int[] arr = {1, 2, 3, 4, 5};
	   System.out.print("Original array: ");
	   print(arr);
	   reverse(arr);
```

```
        System.out.print("Reversed array: ");
        print(arr);
    }

    public static void reverse(int[] arr) {
        int length = arr.length;
        for (int i = 0; i < length / 2; i++) {

            int temp = arr[i];
            arr[i] = arr[length - 1 - i];
            arr[length - 1 - i] = temp;
        }
    }

    public static void print(int[] arr) {
        for (int value : arr) {
            System.out.print(value + " ");
        }
        System.out.println();
    }
}
```

Output:
Original array: 1 2 3 4 5
Reversed array: 5 4 3 2 1

10. Write a program in Java to count the total number of vowels and consonants in a String. The

string can contain all the alphanumeric and other special characters as well. However, only the

lowercase English alphabets are allowed in the String.

```
package task;

public class CountVC {

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

```
        int vCount = 0, cCount = 0;
        String str = "This is a simple sentence";


        str = str.toLowerCase();

        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);

            if (Character.isLetter(ch)) {

                if (vowel(ch)) {
                    vCount++;
                } else {
                    cCount++;
                }
            }
        }

        System.out.println("Number of vowels: " + vCount);
        System.out.println("Number of consonants: " + cCount);
    }

    private static boolean vowel(char ch) {
        return ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u';
    }
}
```

Output:
Number of vowels: 8
Number of consonants: 13