Task:1

Write a Java program where the current year and the year in which an employee joined the organization are entered through the keyboard. If the number of years for which the employee has served the organization is greater than 5, a bonus of Rs. 5000/- is given to the employee. If the years of service are between 3 and 5 (inclusive), a bonus of Rs. 3000/- is given. If the years of service are less than 3, then the program should print a message indicating that no bonus is awarded.

```
import java.util.Scanner;
public class EmployeeBonus {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter the current year: ");
      int currentYear = scanner.nextInt();
      System.out.print("Enter the year the employee joined: ");
      int joinYear = scanner.nextInt();
      int yearsOfService = currentYear - joinYear
      if (yearsOfService > 5) {
         System.out.println("Congratulations! You have been awarded a bonus of Rs. 5000/-");
      } else if (yearsOfService >= 3 && yearsOfService <= 5) {
         System.out.println("Congratulations! You have been awarded a bonus of Rs. 3000/-");
      } else {
         System.out.println("Sorry, no bonus is awarded.");
      } scanner.close();
   }
}
```

Task:2

A library charges a fine for every book returned late. For the first 7 days, the fine is 50 paise, for 8-14 days the fine is one rupee, and above 14 days fine is 5 rupees. If you return the book after 21 days, your membership will be canceled. Write a program to accept the number of days the member is late to return the book and display the fine or the appropriate message.

```
import java.util.Scanner;
public class LibraryFine {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of days late: ");
        int daysLate = scanner.nextInt();
        if (daysLate <= 7) {
            System.out.println("Fine: 50 paise");
        } else if (daysLate <= 14) {
            System.out.println("Fine: Rs. 1");
        } else if (daysLate <= 21) {
            System.out.println("Fine: Rs. 5");
        } else {
            System.out.println("Your membership is canceled.");
        } scanner.close();
    }
}
```

Task:3

Write a Java program to check if a vowel is present in a string. Additionally, the program should count the total number of vowels present and display the count along with whether any vowels are present or not.

```
import java.util.Scanner;
public class VowelChecker {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a string: ");
      String input = scanner.nextLine().toLowerCase();
      int vowelCount = 0;
      boolean vowelPresent = false;
      for (int i = 0; i < input.length(); i++) {
         char ch = input.charAt(i);
         if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {
            vowelPresent = true;
            vowelCount++;
         }
  }
      if (vowelPresent) {
         System.out.println("Vowels are present in the string.");
         System.out.println("Total number of vowels: " + vowelCount);
      } else {
         System.out.println("No vowels are present in the string.");
      }
scanner.close();
   }
}
```

Task: 4

Write a Java program that accepts a list of student projects andreturns the number of projects that were completed on time, late,or early, and the average time it took for each student to completetheir projects

```
Import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

class Project {
   private String studentName;
   private int completionTime; // in days
public Project(String studentName, int completionTime) {
      this.studentName = studentName;
      this.completionTime = completionTime;
   }public String getStudentName() {
      return studentName;
   }public int getCompletionTime() {
      return completionTime;
   }
}public class StudentProjects {
   public static void main(String[] args) {
      List<Project> projects = new ArrayList<>();
      projects.add(new Project("John", 5));
      projects.add(new Project("Alice", 7));
      projects.add(new Project("Bob", 3));
      projects.add(new Project("Carol", 6));
      projects.add(new Project("David", 8));
Map<String, Integer> onTime = new HashMap<>();
      Map<String, Integer> late = new HashMap<>();
```

```
        Map<String, Integer> early = new HashMap<>();
        Map<String, Integer> projectCounts = new HashMap<>();
        Map<String, Integer> totalTime = new HashMap<>();

        for (Project project : projects) {
            String studentName = project.getStudentName();
            int completionTime = project.getCompletionTime();

            projectCounts.put(studentName, projectCounts.getOrDefault(studentName, 0) + 1);
            totalTime.put(studentName, totalTime.getOrDefault(studentName, 0) +
completionTime);

            if (completionTime == 5) { // Assuming 5 days is the on-time threshold
                onTime.put(studentName, onTime.getOrDefault(studentName, 0) + 1);
            } else if (completionTime > 5) {
                late.put(studentName, late.getOrDefault(studentName, 0) + 1);
            } else {
                early.put(studentName, early.getOrDefault(studentName, 0) + 1);
            }
        }

        System.out.println("Number of projects completed on time:");
        printMap(onTime);
        System.out.println("\nNumber of projects completed late:");
        printMap(late);
        System.out.println("\nNumber of projects completed early:");
        printMap(early);
        System.out.println("\nAverage completion time for each student:");
```

```
        for (String studentName : projectCounts.keySet()) {
            int averageTime = totalTime.get(studentName) / projectCounts.get(studentName);
            System.out.println(studentName + ": " + averageTime + " days");
        }
    }

    private static void printMap(Map<String, Integer> map) {
        for (Map.Entry<String, Integer> entry : map.entrySet()) {
            System.out.println(entry.getKey() + ": " + entry.getValue());
        }
    }
}
```

Task: 5

Write a Java program that accepts a list of movie ratings andreturns the number of movies rated in various categories (e.g.,PG, PG-13, R, etc.), and the average rating for each category.

```
import java.util.ArrayList;

import java.util.HashMap;
import java.util.List;
import java.util.Map;

class Movie {
    private String category;
    private double rating;

    public Movie(String category, double rating) {
        this.category = category;
        this.rating = rating;
    }

    public String getCategory() {
        return category;
    }

    public double getRating() {
        return rating;
    }
}

public class MovieRatings {
    public static void main(String[] args) {
        List<Movie> movies = new ArrayList<>();
        movies.add(new Movie("PG", 4.5));
        movies.add(new Movie("PG-13", 3.8));
        movies.add(new Movie("R", 4.2));
        movies.add(new Movie("PG", 4.1));
        movies.add(new Movie("PG-13", 3.9));
        movies.add(new Movie("R", 4.5));
        movies.add(new Movie("PG", 3.7));

        Map<String, Integer> movieCounts = new HashMap<>();
        Map<String, Double> totalRatings = new HashMap<>();
```

```
        for (Movie movie : movies) {
            String category = movie.getCategory();
            double rating = movie.getRating();

            movieCounts.put(category, movieCounts.getOrDefault(category, 0) + 1);
            totalRatings.put(category, totalRatings.getOrDefault(category, 0.0) + rating);
        }

        System.out.println("Number of movies rated in each category:");
        for (String category : movieCounts.keySet()) {
            System.out.println(category + ": " + movieCounts.get(category));
        }

        System.out.println("\nAverage rating for each category:");
        for (String category : totalRatings.keySet()) {
            double averageRating = totalRatings.get(category) /
movieCounts.get(category);
            System.out.println(category + ": " + averageRating);
        }
    }
}
```