1. Design and implement a class named InstanceCounter to track and count the number of instances created from this class.

**InstanceCounter.java**

package in.cdac.ques1;

public class InstanceCounter {

private static int count=0; //declare count as a static field so that count will be shared for all instances

public InstanceCounter(){

InstanceCounter.count++; //increment count after an instance is created and constructor is called

}

public static int getCount() {

return InstanceCounter.count;

}

}

**Project.java**

package in.cdac.ques1;

public class Project {

public static void main(String[] args) {

InstanceCounter i1 = new InstanceCounter();

InstanceCounter i2 = new InstanceCounter();

InstanceCounter i3 = new InstanceCounter();

InstanceCounter i4 = new InstanceCounter();

System.out.println("Total instances of Class InstanceCounter: "+InstanceCounter.getCount());

}

}



1. Design and implement a class named Logger to manage logging messages for an application. The class should be implemented as a singleton to ensure that only one instance of the Logger exists throughout the application.

The class should include the following methods:

* **getInstance()**: Returns the unique instance of the Logger class.
* **log(String message)**: Adds a log message to the logger.
* **getLog()**: Returns the current log messages as a String.
* **clearLog()**: Clears all log messages.

**Logger.java**

package in.cdac.ques2;

public class Logger { //Singleton Class

private static Logger l1;

private static String message;

static {

Logger.l1 = null;

Logger.message="";

}

private Logger() { //private constructor to restrict creation of instance from outside the class

}

public static Logger getInstance() {

if(Logger.l1 == null) { //check if there is no instance inside reference l1

Logger.l1 = new Logger(); //only create new instance of Logger Class if there is no instance present in reference l1

}

return l1;

}

public void log(String message) {

Logger.message = message;

}

public String getLog(){

return Logger.message;

}

public void clearLog() {

Logger.message = "";

}

}

**Project.java**

package in.cdac.ques2;

import java.util.Scanner;

public class Project {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int choice;

Logger lg = Logger.getInstance();

System.out.println("\n1. Enter log\n2. Print log\n3. Delete log\n0.Exit");

while((choice = sc.nextInt())!=0) {

switch(choice){

case 1:

sc.nextLine();

System.out.println("\nEnter the log message you want to store: ");

String s1 = sc.nextLine();

lg.log(s1);

break;

case 2:

if(lg.getLog()=="")

System.out.println("No log entry is present.");

else

System.out.println("Current log: "+lg.getLog());

break;

case 3:

lg.clearLog();

System.out.println("Log entry has been deleted.");

break;

}

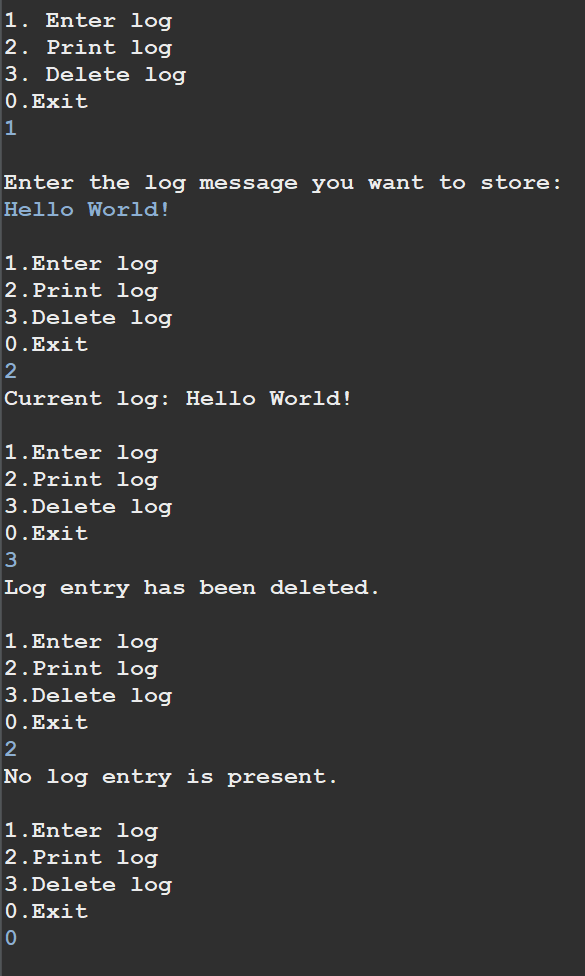
System.out.println("\n1.Enter log\n2.Print log\n3.Delete log\n0.Exit");

}

sc.close();

}

}



1. Design and implement a class named Employee to manage employee data for a company. The class should include fields to keep track of the total number of employees and the total salary expense, as well as individual employee details such as their ID, name, and salary.

The class should have methods to:

* Retrieve the total number of employees (getTotalEmployees())
* Apply a percentage raise to the salary of all employees (applyRaise(double percentage))
* Calculate the total salary expense, including any raises (calculateTotalSalaryExpense())
* Update the salary of an individual employee (updateSalary(double newSalary))

Understand the problem statement and use static and non-static fields and methods appropriately. Implement static and non-static initializers, constructors, getter and setter methods, and a toString() method to handle the initialization and representation of employee data.

Write a menu-driven program in the main method to test the functionalities.

**Employee.java**

package in.cdac.ques3;

import java.util.Scanner;

public class Employee {

private String name;

private int id;

private double salary;

private static int noOfEmployees;

private static double totalSalary;

static {

}

{

Employee.noOfEmployees++;

}

public Employee() {

this.name="";

}

public Employee(String name, int id, double salary) {

this.name = name;

this.id = id;

this.salary = salary;

totalSalary = totalSalary + salary;

}

// private static Scanner sc = new Scanner(System.in);

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public double getSalary() {

return salary;

}

public void setSalary(double salary) {

this.salary = salary;

}

public static int getNoOfEmployees() {

return noOfEmployees;

}

public static void setNoOfEmployees(int noOfEmployees) {

Employee.noOfEmployees = noOfEmployees;

}

public static double getTotalSalary() {

return totalSalary;

}

public static void setTotalSalary(double totalSalary) {

Employee.totalSalary = totalSalary;

}

public static int getTotalEmployees() {

return Employee.getNoOfEmployees();

}

public void applyRaise(double percentage) {

double raiseAmount = (percentage/100)\*this.getSalary();

this.setSalary(this.getSalary()+raiseAmount);

setTotalSalary(getTotalSalary()+raiseAmount);

}

public static double calculateTotalSalaryExpense() {

return getTotalSalary();

}

public void updateSalary(double newSalary) {

setTotalSalary(getTotalSalary()-this.getSalary());

this.setSalary(newSalary);

setTotalSalary(getTotalSalary()+this.getSalary());

}

public static int menuList(Scanner sc) {

System.out.println("1. Get total number of employees");

System.out.println("2. Apply percentage raise to the salary of all employees");

System.out.println("3. Get total salary expense of all employees");

System.out.println("4. Update salary of an employee");

System.out.println("5. Print salary of an employee");

System.out.println("0. Exit");

return sc.nextInt();

}

}

**Program.java**

package in.cdac.ques3;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

int choice;

int choiceNext;

double raise;

double salaryNext;

Scanner sc =new Scanner(System.in);

Employee emp1 = new Employee("Rahul",2080,60000);

Employee emp2 = new Employee("John",2081,55000);

// System.out.println(emp1.getSalary());

// System.out.println(emp2.getSalary());

// System.out.println("Total salary expense of all Employees: "+Employee.calculateTotalSalaryExpense());

//

// emp1.applyRaise(15);

// emp2.applyRaise(10);

// System.out.println("After Raise");

// System.out.println(emp1.getSalary());

// System.out.println(emp2.getSalary());

//

// System.out.println("Total salary expense of all Employees: "+Employee.calculateTotalSalaryExpense());

// System.out.println("Total no. of Employees: "+Employee.getTotalEmployees());

while ((choice=Employee.menuList(sc))!=0) {

switch(choice) {

case 1: System.out.println("Total no. of employees:"+Employee.getNoOfEmployees());

break;

case 2: raise = sc.nextDouble();

emp1.applyRaise(raise);

emp2.applyRaise(raise);

break;

case 3: System.out.println("Total salary expense for all employees:"+Employee.calculateTotalSalaryExpense());

break;

case 4: System.out.println("1. Name: "+emp1.getName()+" ID: "+emp1.getId());

System.out.println("2. Name: "+emp2.getName()+" ID: "+emp2.getId());

choiceNext = sc.nextInt();

switch(choiceNext) {

case 1: System.out.println("Enter the new salary for this employee: ");

salaryNext = sc.nextDouble();

emp1.updateSalary(salaryNext);

break;

case 2: System.out.println("Enter the new salary for this employee: ");

salaryNext = sc.nextDouble();

emp2.updateSalary(salaryNext);

break;

}

break;

case 5: System.out.println("1. Name: "+emp1.getName()+" ID: "+emp1.getId());

System.out.println("2. Name: "+emp2.getName()+" ID: "+emp2.getId());

choiceNext = sc.nextInt();

switch(choiceNext) {

case 1: System.out.println("Salary of "+emp1.getName()+":");

System.out.println(emp1.getSalary());

break;

case 2: System.out.println("Salary of "+emp2.getName()+":");

System.out.println(emp2.getSalary());

break;

}

}

}

sc.close();

}

}

