**Note:**

1. This assignment is designed to practice static fields, static initializers, and static methods.
2. Understand the problem statement and use static and non-static wisely to solve the problem.
3. Use constructors, proper getter/setter methods, and toString() wherever required.
4. Design and implement a class named InstanceCounter to track and count the number of instances created from this class.

*InstanceCounter.java*

package com.example.countinstance;

public class InstanceCounter {

int a,b,c;

public static int *count*;

public InstanceCounter() {

++*count*;

}

public void setA(int a) {

this.a = a;

}

public void setB(int b) {

this.b = b;

}

public void setC(int c) {

this.c = c;

}

public int getC() {

return c != 0 ? c : (this.a + this.b);

}

*@Override*

public String toString() {

return String.*valueOf*(*count*);

}

}

*Program.java*

package com.example.countinstance;

public class Program {

public static void main(String[] args) {

InstanceCounter ic1 = new InstanceCounter();

InstanceCounter ic2 = new InstanceCounter();

InstanceCounter ic3 = new InstanceCounter();

ic2.setA(7);

ic2.setB(9);

System.***out***.println(ic2.getC());

ic3.setC(43);

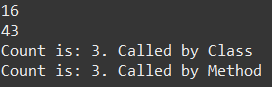
System.***out***.println(ic3.getC());

System.***out***.println("Count is: " + InstanceCounter.*count* + ". Called by Class");

System.***out***.println("Count is: " + ic1.toString() + ". Called by Method");

}

}



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.

*Program.java*

package com.example.logger;

import java.util.Scanner;

public class Program {

public static void main(String[] args) {

Logger lg1 = Logger.*getInstance*();

Scanner sc = new Scanner(System.***in***);

Logger lg2 = Logger.*getInstance*();

System.***out***.print("Enter log message for Logger1: ");

lg1.log(sc.next());

System.***out***.println(lg1.getLog());

System.***out***.print("Enter log message for Logger2: ");

//lg2.log(sc.next());

System.***out***.println(lg2.getLog());

lg1.clearLog();

System.***out***.println(lg1.getLog());

sc.close();

}

}

*Logger.java*

package com.example.logger;

public class Logger {

String message;

private Logger() {

this.message = "Currently no log exist!";

}

private static Logger *reference* = null;

public static Logger getInstance() {

if( *reference* == null )

*reference* = new Logger();

return *reference*;

}

public void log(String message) {

this.message = message;

}

public String getLog() {

return message;

}

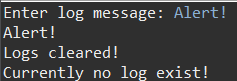
public void clearLog() {

this.message = "Currently no log exist!";

System.*out*.println("Logs cleared!");

}

}

****

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 com.example.employee;

public class Employee {

private String name;

private int id;

private float salary;

private static int *totalEmp* = 0;

private static float *salaryExp*;

public Employee() {

this("", 0, 0.0f);

}

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

++*totalEmp*;

this.name = name;

this.id = id;

this.salary = salary;

}

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 float getSalary() {

return salary;

}

public void setSalary(float salary) {

if(this.salary != salary)

Employee.*salaryExp* += salary - this.salary;

this.salary = salary;

}

public static int getTotalEmp() {

return *totalEmp*;

}

public static void setTotalEmp(int totalEmp) {

Employee.*totalEmp* = totalEmp;

}

public static float getSalaryExp() {

return *salaryExp*;

}

public static void setSalaryExp(float salaryExp) {

Employee.*salaryExp* = salaryExp;

}

public void applyRaise(float percentage){

float appraisal = (float) (1 + (percentage/100));

setSalary(this.salary \* appraisal);

}

}

*EmpUtils.java*

package com.example.employee;

import java.util.Scanner;

public class EmpUtils {

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

Employee emp = new Employee();

public void acceptRecord() {

System.***out***.print("Enter employee name: ");

emp.setName(*sc*.next());

System.***out***.print("Enter employee ID: ");

emp.setId(*sc*.nextInt());

System.***out***.print("Enter employee salary: ");

emp.setSalary(*sc*.nextFloat());

}

public void setRaise() {

System.***out***.print("Enter raise percentage: ");

emp.applyRaise(*sc*.nextInt());

}

public void showRecord() {

System.***out***.println("No of total employee is: " + Employee.*getTotalEmp*());

System.***out***.println("Total salary expense is: " + Employee.*getSalaryExp*());

}

public void closeResource() {

*sc*.close();

}

}

*Program.java*

package com.example.employee;

public class Program {

public static void main(String[] args) {

EmpUtils eu = new EmpUtils();

eu.acceptRecord();

eu.setRaise();

eu.showRecord();

eu.closeResource();

}

}

