

# Sandeep Sir: Assignment 5

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

→ CounterClass

```
package com.assignment.question1.counter;

public class CounterClass {
    private static int counter;

    public CounterClass() {
        // TODO Auto-generated constructor stub
        CounterClass.setCounter();
    }

    public static int getCounter() {
        return counter;
    }

    public static void setCounter() {
        CounterClass.counter += 1;
    }
}
```

MainClass

```
package com.assignment.question1.main;
import com.assignment.question1.counter.CounterClass;

public class MainClass {
```

```

public static void main(String[] args) {
    // TODO Auto-generated method stub
    CounterClass c1 = new CounterClass();
    CounterClass c2 = new CounterClass();
    CounterClass c3 = new CounterClass();

    System.out.println("Counter is: " + CounterClass.ge
tCounter());
}
}

```

```

<terminated> MainClass
Counter is: 3

```

2. 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.

→ LoggerClass

```

package com.assignment.question2.logger;

public class Logger {
    private static Logger referance;
    private String loggerMsg = "";

```

```

private Logger() {}

public static Logger getInstance(){
    if(Logger.reference == null) {
        Logger.reference = new Logger();
    }
    return Logger.reference;
}

public void log(String message) {
    this.loggerMsg += message;
}

public String getLog() {
    return "Your Log Messages are:" + this.loggerMsg;
}

public void clearLog() {
    this.loggerMsg = "";
    System.out.println("Logger is Cleared");
}
}

```

## MainClass

```

package com.assignment.question2.main;
import com.assignment.question2.logger.Logger;

public class MainClass {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Logger l1;
        l1 = Logger.getInstance();
    }
}

```

```

        l1.log("Hi ");
        System.out.println(l1.getLog());

        Logger l2 = Logger.getInstance();
        l2.log("I'm Sumant Reddy");
        System.out.println(l2.getLog());

        l2.clearLog();
        System.out.println(l1.getLog());
    }
}

```

```

<terminated> MainClass (1) [Java Application] C:\Users\reddy\p2
Your Log Messages are:Hi
Your Log Messages are:Hi I'm Sumant Reddy
Logger is Cleared
Your Log Messages are:

```

- 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.

→ EmployeeClass

```
package com.assignment.question3.employee;

import java.util.Scanner;

public class Employee {
    private String name;
    private int id;
    private double salary;
    private static int NumOfEmployee;
    private static double TotalExpenditure;
    private static double raise;

    public Employee(String name, int id, double salary) {
        this.name = name;
        this.id = id;
        this.salary = salary;

        Employee.NumOfEmployee += 1;
        Employee.TotalExpenditure += this.salary;
    }

    public int menuList(Scanner sc) {
        System.out.println("\n");
        System.out.println("0.Exit.");
        System.out.println("1.Get total employees.");
        System.out.println("2.Apply Raise Expenditure.");
        System.out.println("3.Get Total Expenditure.");
        System.out.println("4.Update Salary.");
        System.out.println("5.Print Record.");
    }
}
```

```

        System.out.print("Enter choice : ");
        int choice = sc.nextInt( );
        return choice;
    }

    public static String getTotalEmployees() {
        return "Total Number of Employees: " + Employee.Num
OfEmployee;
    }

    public static void applyRaise(Scanner sc) {
        Employee.raise = sc.nextDouble();
        System.out.println("Updated Raise!!");
    }

    public static void calculateTotalSalaryExpense() {
        double raiseAmount = Employee.TotalExpencc * (Emplo
yee.raise/100);
        Employee.TotalExpencc += raiseAmount;
    }

    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 this.salary + (this.salary*(Employee.raise/1
00));
        }

        public void updateSalary(Scanner sc) {
            this.salary = sc.nextDouble();
            System.out.println("Updated Salary!!");
        }
        public static String getTotalExpenditure() {
            Employee.calculateTotalSalaryExpenditure();
            return "Total Expenditure is: "+ Employee.TotalExpenditure;
        }

        @Override
        public String toString() {
            return "Employee [name=" + this.getName() + ", id="
+ this.getId() + ", salary=" + this.getSalary() + "]\n";
        }
    }
}

```

## MainClass

```

package com.assignment.question3.main;
import java.util.Scanner;

import com.assignment.question3.employee.Employee;

public class MainClass {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        Employee e1 = new Employee("Sumant", 101, 50000.5);
        Employee e2 = new Employee("Hemant", 101, 30000.7);
    }
}

```

```

Employee e3 = new Employee("xyz", 101, 20000.3);

int choice;
while ( ( choice = e1.menuList( sc ) ) != 0 ) {
    switch( choice ) {
        case 1:
            System.out.println(Employee.getTotalEmployee
es());
            break;
        case 2:
            Employee.applyRaise(sc);
            break;
        case 3:
            System.out.println(Employee.getTotalExpende
());
            break;
        case 4:
            e1.updateSalary(sc);
            break;
        case 5:
            System.out.println(e1.toString());
            break;
    }
}
}
}

```



<terminated> MainClass (2) [Java Application] C:\Users\reddy\p2\

```
0.Exit.
1.Get total employees.
2.Apply Raise Expense.
3.Get Total Expense.
4.Update Salary.
5.Print Record.
Enter choice      :      1
Total Number of Employees: 3
```

```
0.Exit.
1.Get total employees.
2.Apply Raise Expense.
3.Get Total Expense.
4.Update Salary.
5.Print Record.
Enter choice      :      2
10
Updated Raise!!
```

```
0.Exit.
1.Get total employees.
2.Apply Raise Expense.
3.Get Total Expense.
4.Update Salary.
5.Print Record.
Enter choice      :      3
Total Expense is: 110001.65
```

```
0.Exit.
1.Get total employees.
2.Apply Raise Expense.
3.Get Total Expense.
4.Update Salary.
5.Print Record.
Enter choice      :      4
60000
```

```
0.Exit.  
1.Get total employees.  
2.Apply Raise Expense.  
3.Get Total Expense.  
4.Update Salary.  
5.Print Record.  
Enter choice      :      5  
Employee [name=Sumant, id=101, salary=66000.0]
```

```
0.Exit.  
1.Get total employees.  
2.Apply Raise Expense.  
3.Get Total Expense.  
4.Update Salary.  
5.Print Record.  
Enter choice      :      0
```