UE19CS353:Object Oriented Analysis and Design using Java

Assignment-2: Design Patterns

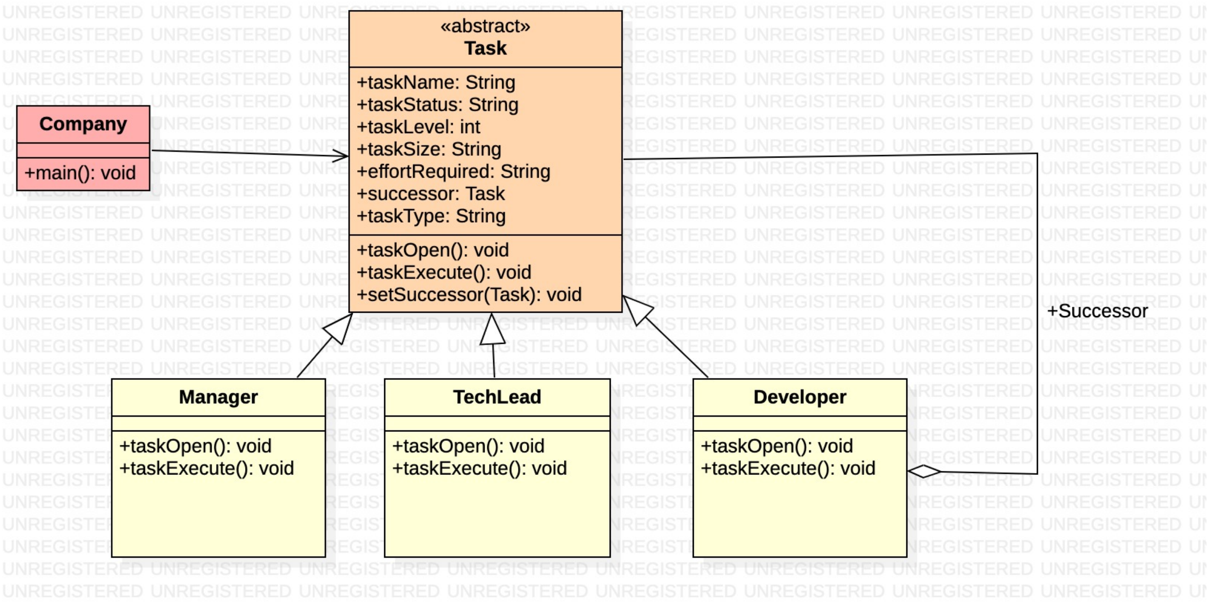
NAME : PRIYA MOHATA
SRN : PES2UG19CS301
SECTION : E

PROBLEM STATEMENT :

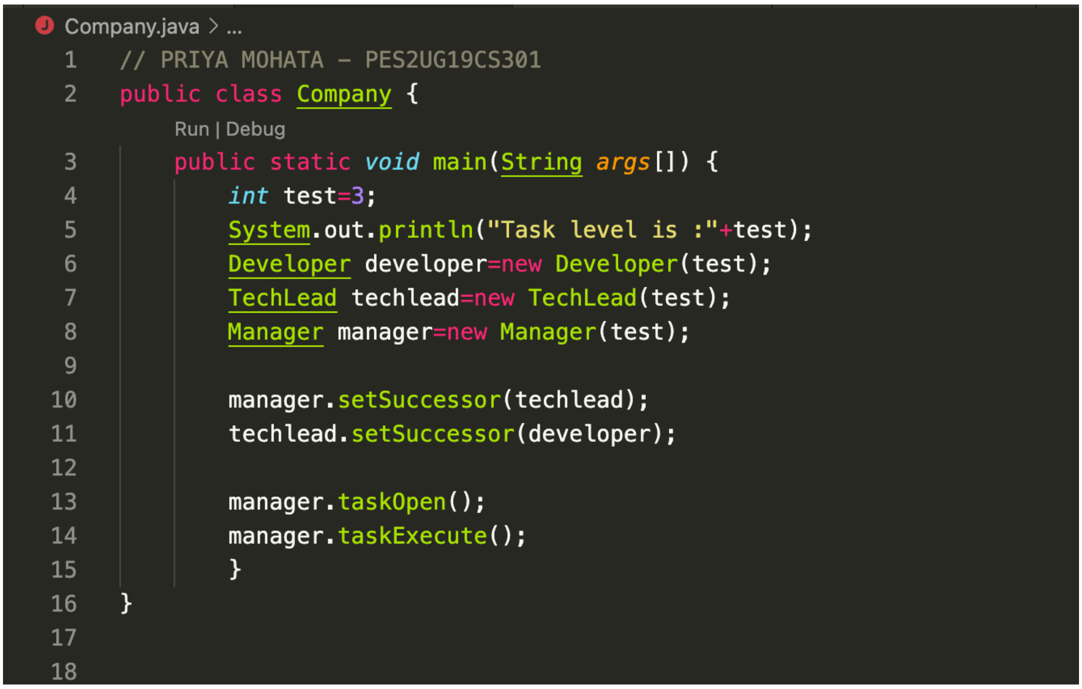
A Company has a Manager, TechLead and Developer as its Employee. The Company creates Task for its Employee. This Task contains attributes such as taskName, taskType, taskStatus, taskLevel, taskSize, effortRequired. After the Task gets created by the Company it will be given to its Employee. Task can be Opened and Executed by the Employee. If the taskLevel is 3, then it will be Opened and Executed by Manager, if it is 2, it will be Opened by the Manager (meaning manager has a task clearance level of 3 and 2) and Executed by the TechLead, and if 1 will be Opened by the TechLead and Executed by the Developer. Design the UML and implement the same using appropriate design patterns. Note: Design the application in such a way that extensibility is easy. It should be easy to add new types of Employee, new actions that can be performed on the task (Opened, Executed, etc.).

DESIGN PATTERN USED : CHAIN OF RESPONSIBILITY

UML DIAGRAM :

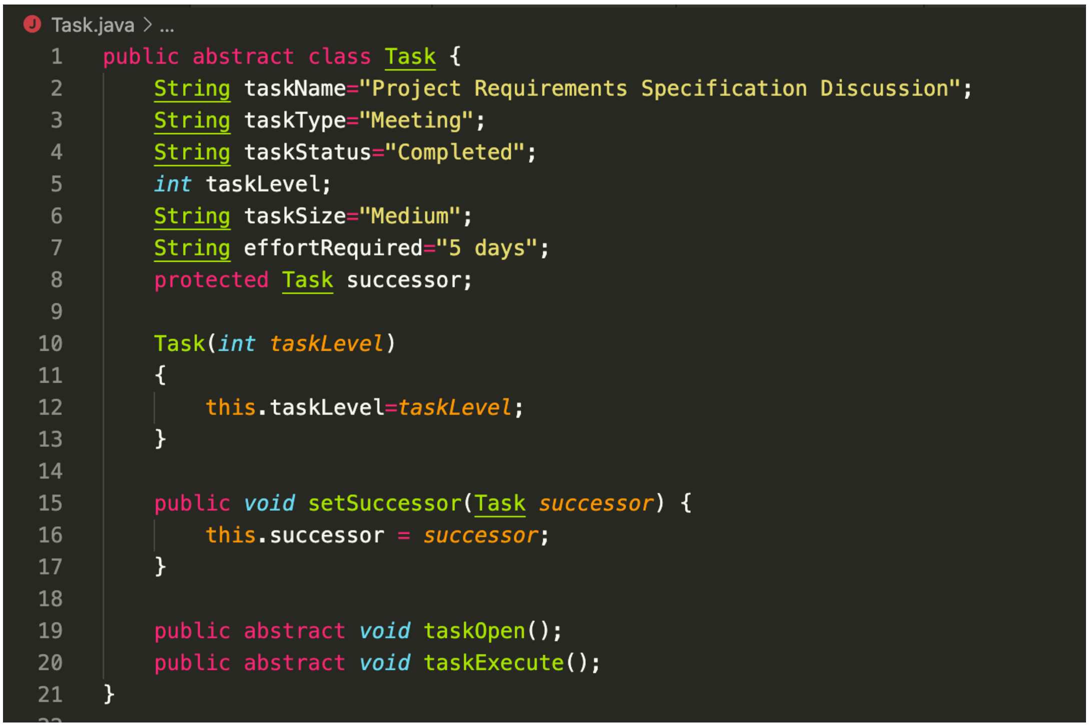


CODE SCREENSHOTS : COMPANY.JAVA



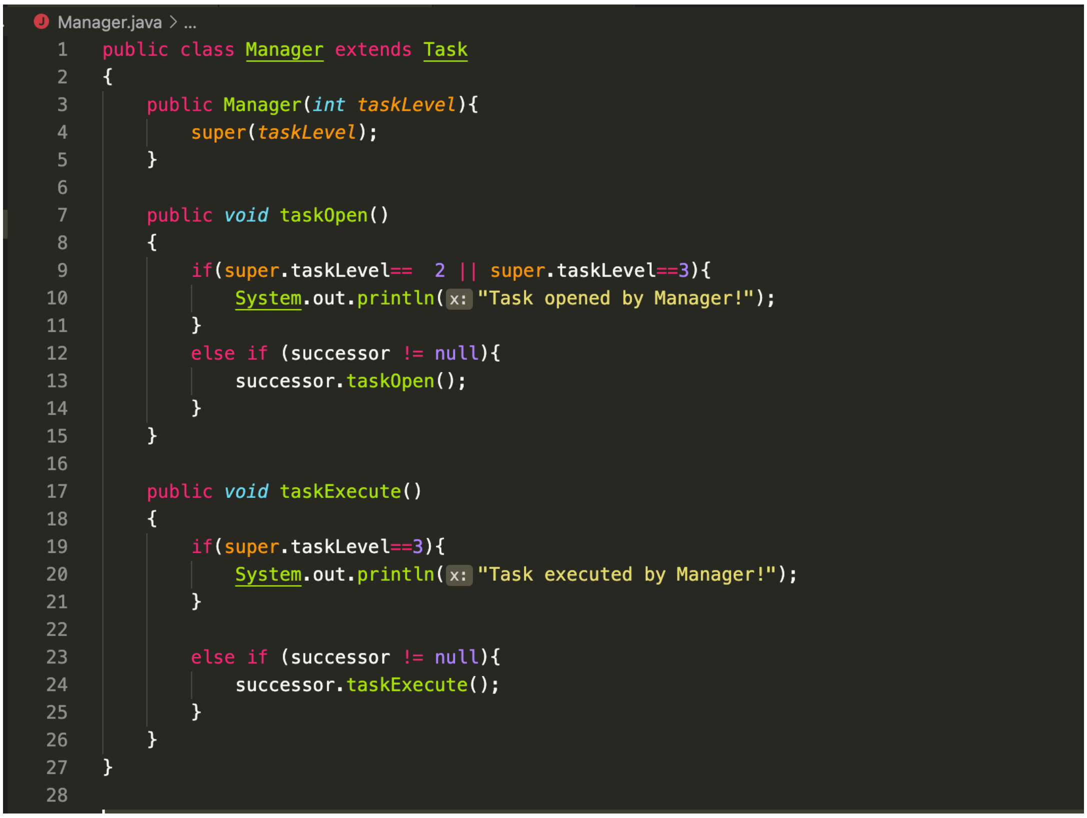
```java
// PRIYA MOHATA - PES2UG19CS301
public class Company {
    public static void main(String args[]) {
        int test=3;
        System.out.println("Task level is :"+test);
        Developer developer=new Developer(test);
        TechLead techlead=new TechLead(test);
        Manager manager=new Manager(test);

        manager.setSuccessor(techlead);
        techlead.setSuccessor(developer);

        manager.taskOpen();
        manager.taskExecute();
        }
}


```

TASK.JAVA



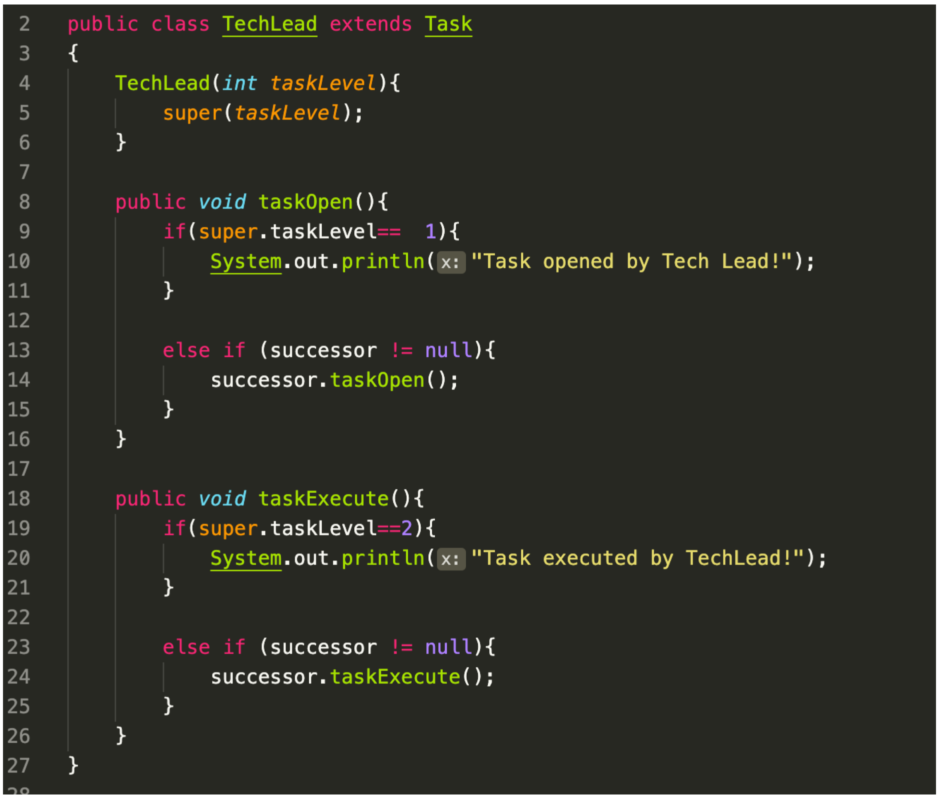
```java
public abstract class Task {
    String taskName="Project Requirements Specification Discussion";
    String taskType="Meeting";
    String taskStatus="Completed";
    int taskLevel;
    String taskSize="Medium";
    String effortRequired="5 days";
    protected Task successor;

    Task(int taskLevel)
    {
        this.taskLevel=taskLevel;
    }

    public void setSuccessor(Task successor) {
        this.successor = successor;
    }

    public abstract void taskOpen();
    public abstract void taskExecute();
}
```

MANAGER.JAVA



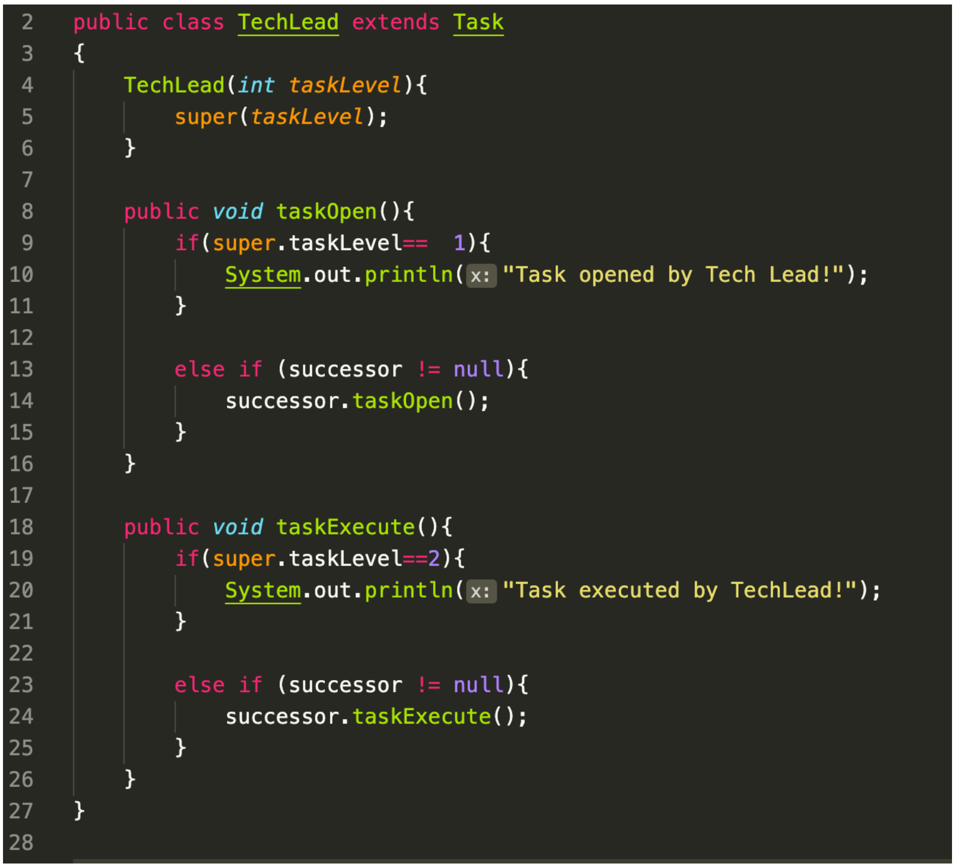
```java
public class Manager extends Task
{
    public Manager(int taskLevel){
        super(taskLevel);
    }

    public void taskOpen()
    {
        if(super.taskLevel==  2 || super.taskLevel==3){
            System.out.println(x: "Task opened by Manager!");
        }
        else if (successor != null){
            successor.taskOpen();
        }
    }

    public void taskExecute()
    {
        if(super.taskLevel==3){
            System.out.println(x: "Task executed by Manager!");
        }

        else if (successor != null){
            successor.taskExecute();
        }
    }
}
```

TECHLEAD.JAVA



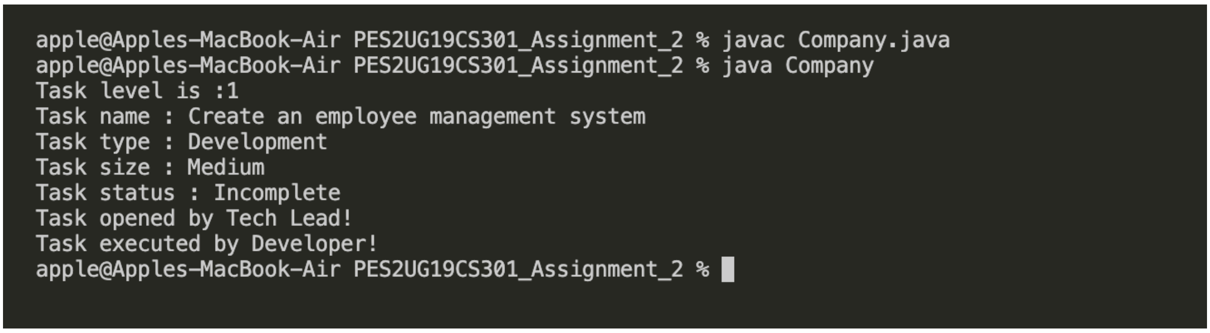
```java
public class TechLead extends Task
{
    TechLead(int taskLevel){
        super(taskLevel);
    }

    public void taskOpen(){
        if(super.taskLevel==  1){
            System.out.println(x: "Task opened by Tech Lead!");
        }

        else if (successor != null){
            successor.taskOpen();
        }
    }

    public void taskExecute(){
        if(super.taskLevel==2){
            System.out.println(x: "Task executed by TechLead!");
        }

        else if (successor != null){
            successor.taskExecute();
        }
    }
}
```

DEVELOPER.JAVA

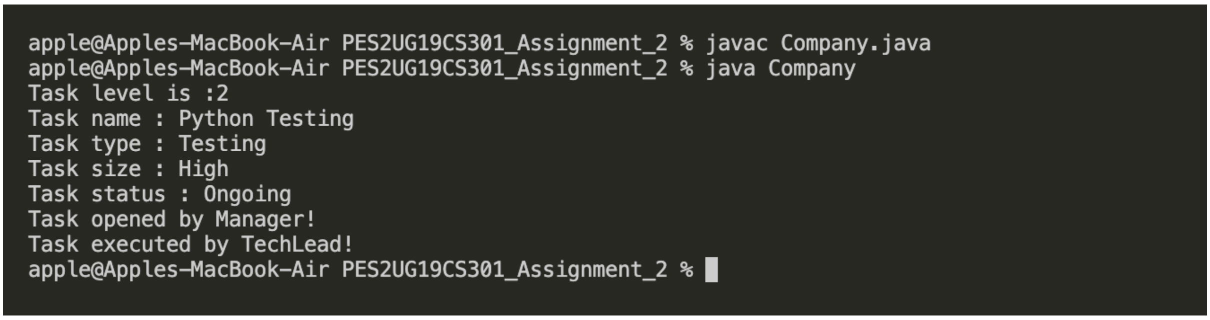


```java
public class TechLead extends Task
{
    TechLead(int taskLevel){
        super(taskLevel);
    }

    public void taskOpen(){
        if(super.taskLevel==  1){
            System.out.println("Task opened by Tech Lead!");
        }

        else if (successor != null){
            successor.taskOpen();
        }
    }

    public void taskExecute(){
        if(super.taskLevel==2){
            System.out.println("Task executed by TechLead!");
        }

        else if (successor != null){
            successor.taskExecute();
        }
    }
}
```

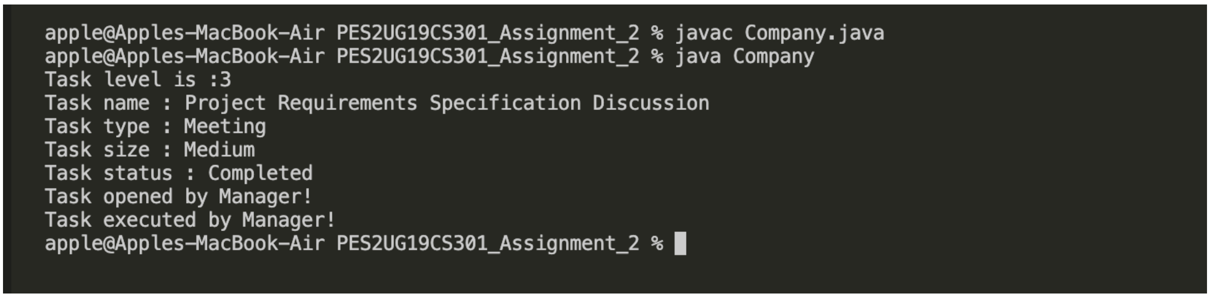
OUTPUT SCREENSHOTS :



```
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % javac Company.java
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % java Company
Task level is :1
Task name : Create an employee management system
Task type : Development
Task size : Medium
Task status : Incomplete
Task opened by Tech Lead!
Task executed by Developer!
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 %
```



```
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % javac Company.java
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % java Company
Task level is :2
Task name : Python Testing
Task type : Testing
Task size : High
Task status : Ongoing
Task opened by Manager!
Task executed by TechLead!
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 %
```



```
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % javac Company.java
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 % java Company
Task level is :3
Task name : Project Requirements Specification Discussion
Task type : Meeting
Task size : Medium
Task status : Completed
Task opened by Manager!
Task executed by Manager!
apple@Apples-MacBook-Air PES2UG19CS301_Assignment_2 %
```