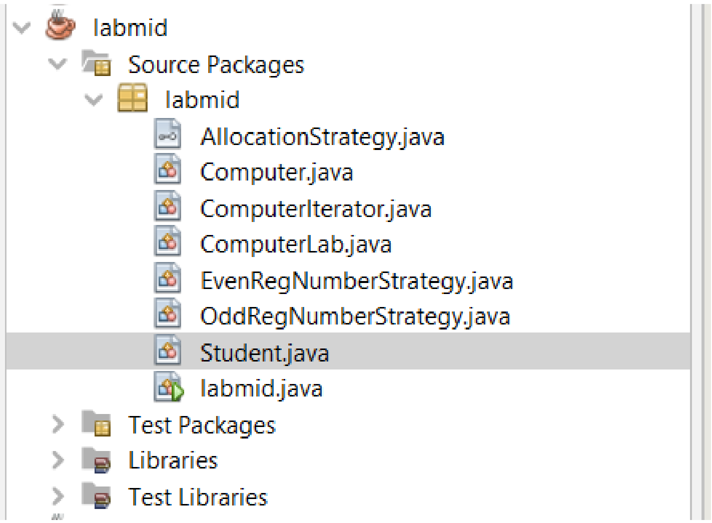
Name: Muhammad Awais

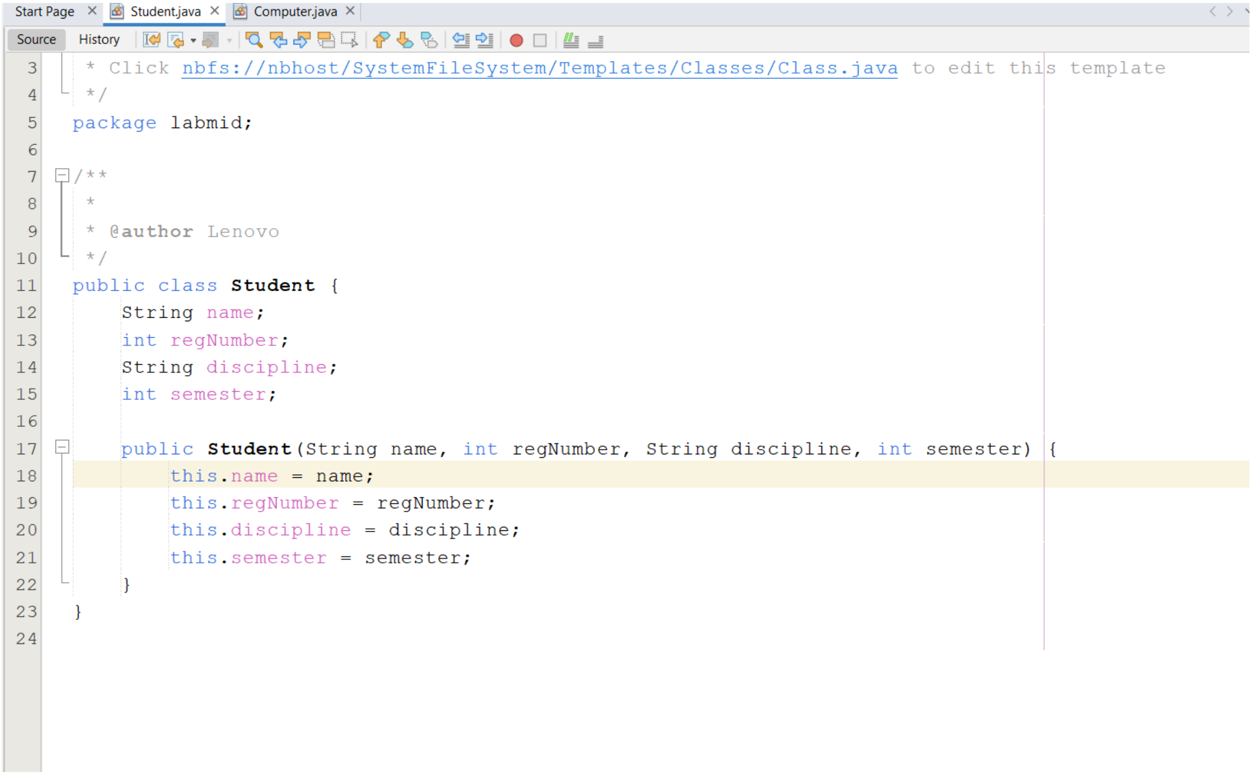
Reg number: FA20-BSE-021

Lab Mid

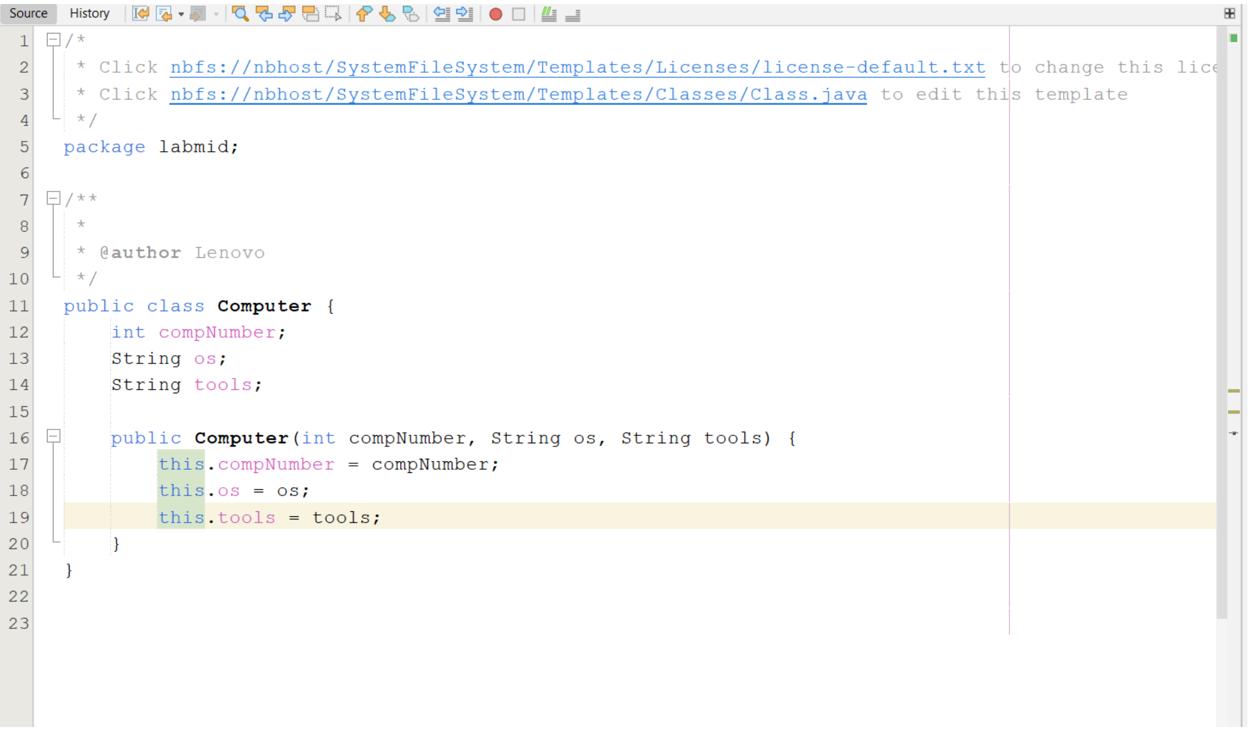
Design Pattern

ScreenShots:

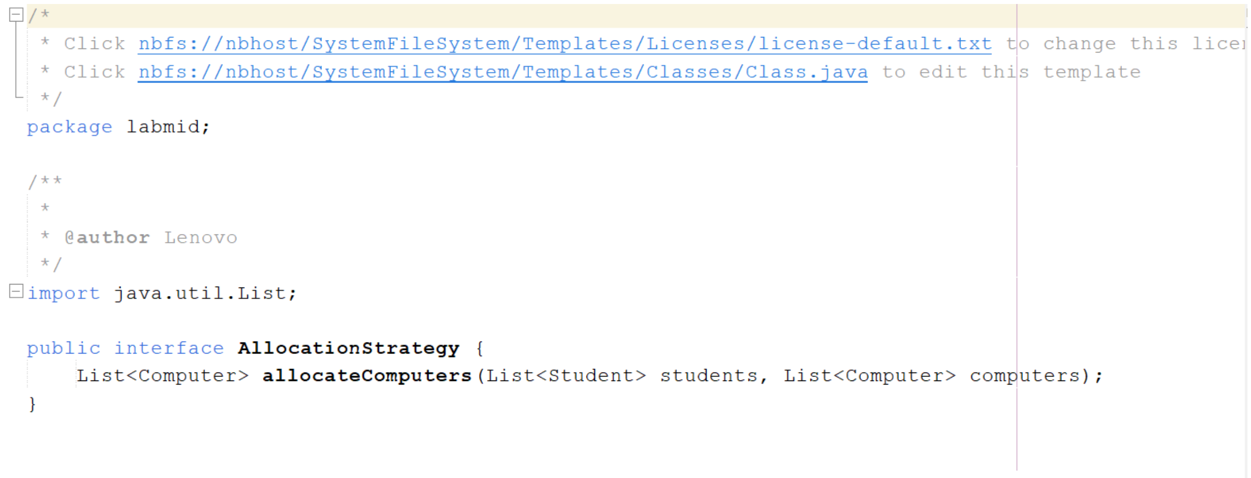




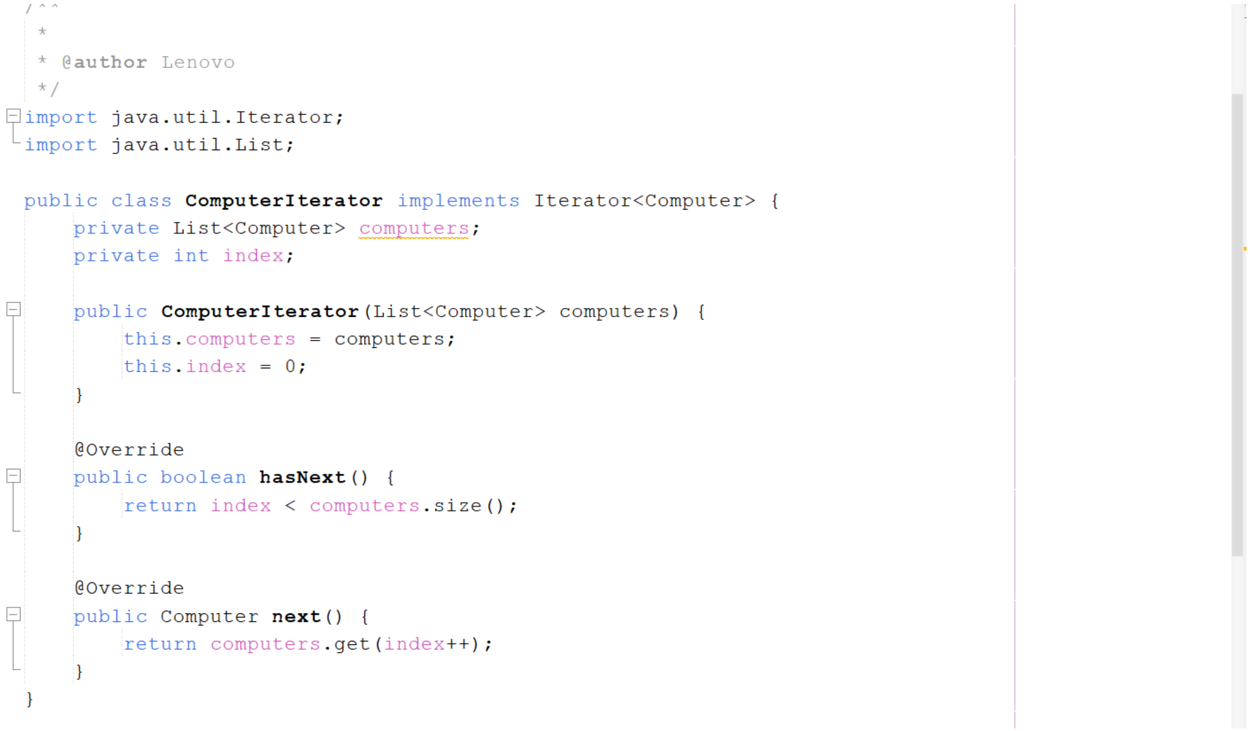
```
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package labmid;

/**
 *
 * @author Lenovo
 */
public class Student {
    String name;
    int regNumber;
    String discipline;
    int semester;

    public Student(String name, int regNumber, String discipline, int semester) {
        this.name = name;
        this.regNumber = regNumber;
        this.discipline = discipline;
        this.semester = semester;
    }
}
```



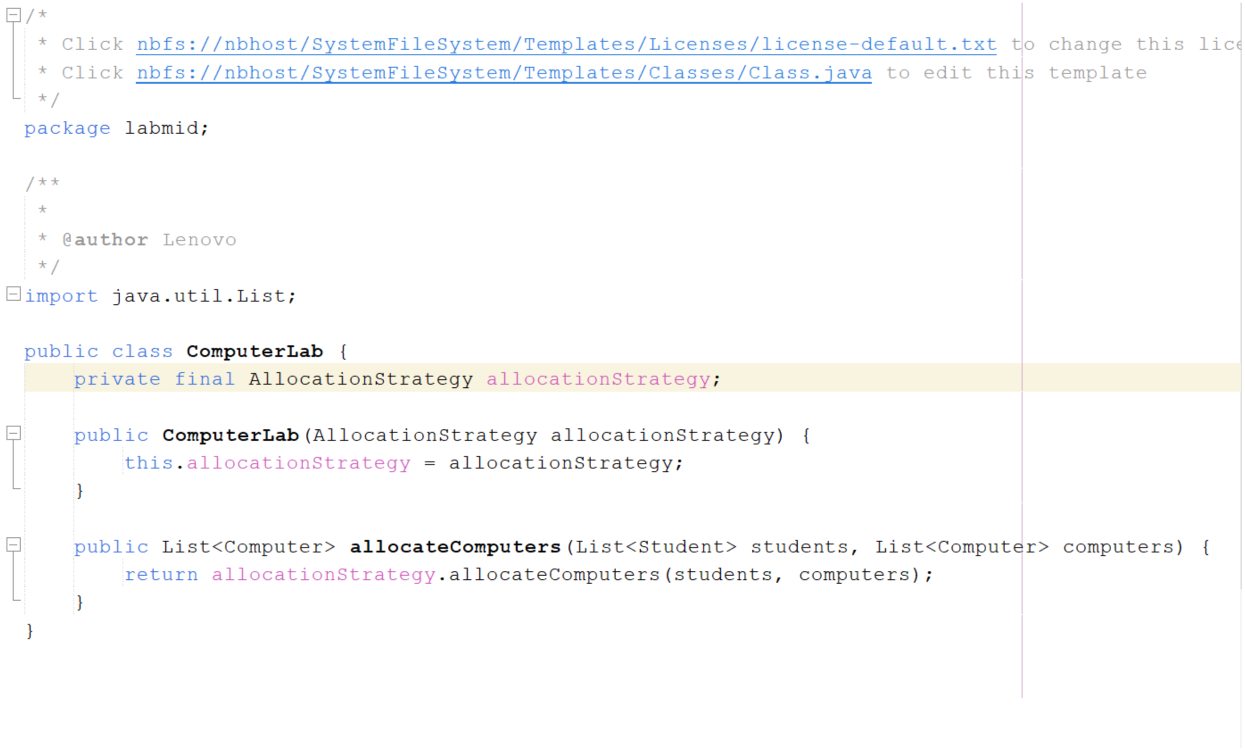
```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this lice
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package labmid;

/**
 *
 * @author Lenovo
 */
public class Computer {
    int compNumber;
    String os;
    String tools;

    public Computer(int compNumber, String os, String tools) {
        this.compNumber = compNumber;
        this.os = os;
        this.tools = tools;
    }
}
```



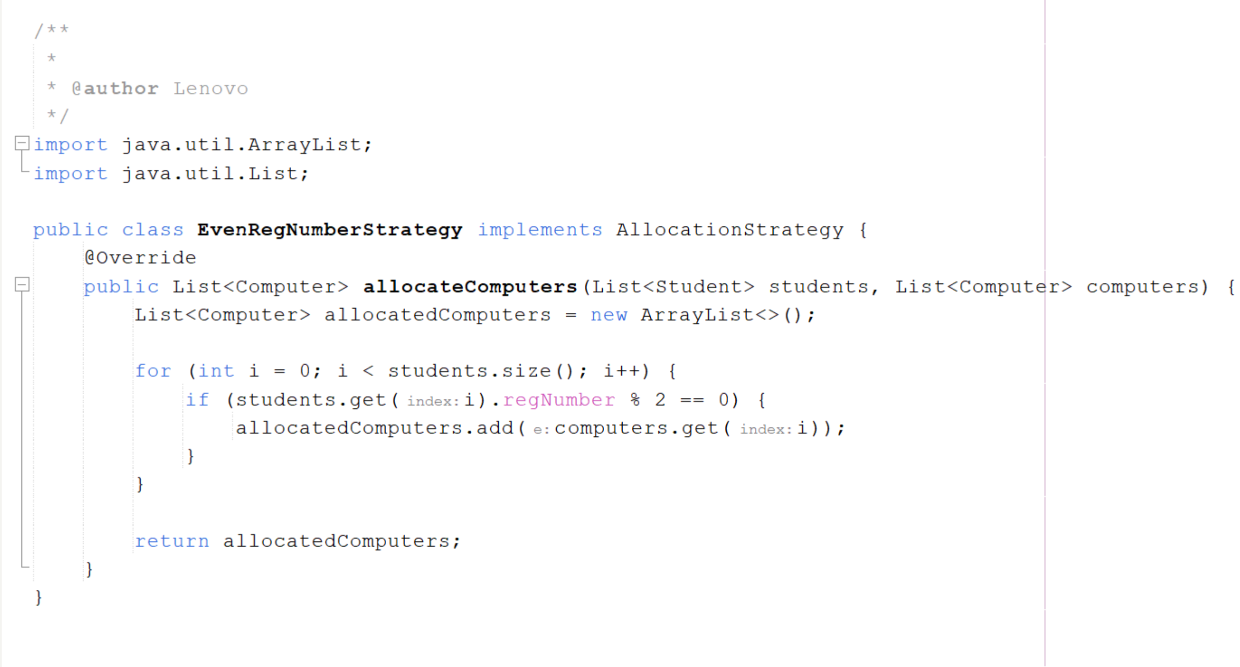
```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this licen
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package labmid;

/**
 *
 * @author Lenovo
 */
import java.util.List;

public interface AllocationStrategy {
    List<Computer> allocateComputers(List<Student> students, List<Computer> computers);
}
```



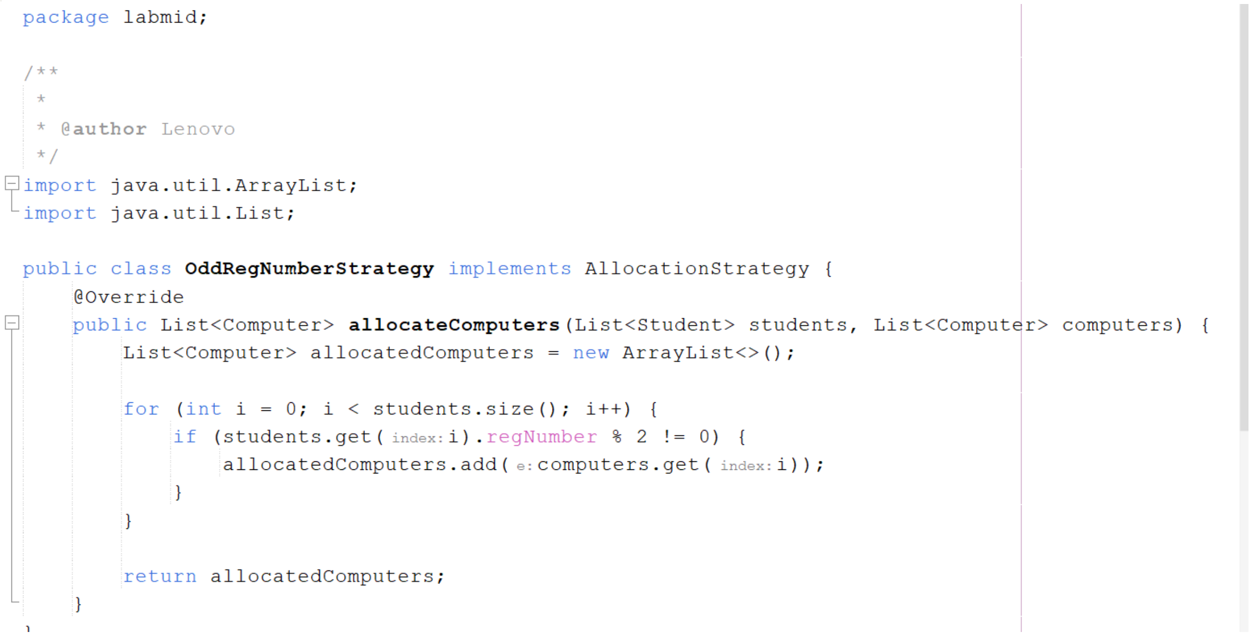
```java
/**
 *
 * @author Lenovo
 */
import java.util.Iterator;
import java.util.List;

public class ComputerIterator implements Iterator<Computer> {
    private List<Computer> computers;
    private int index;

    public ComputerIterator(List<Computer> computers) {
        this.computers = computers;
        this.index = 0;
    }

    @Override
    public boolean hasNext() {
        return index < computers.size();
    }

    @Override
    public Computer next() {
        return computers.get(index++);
    }
}
```



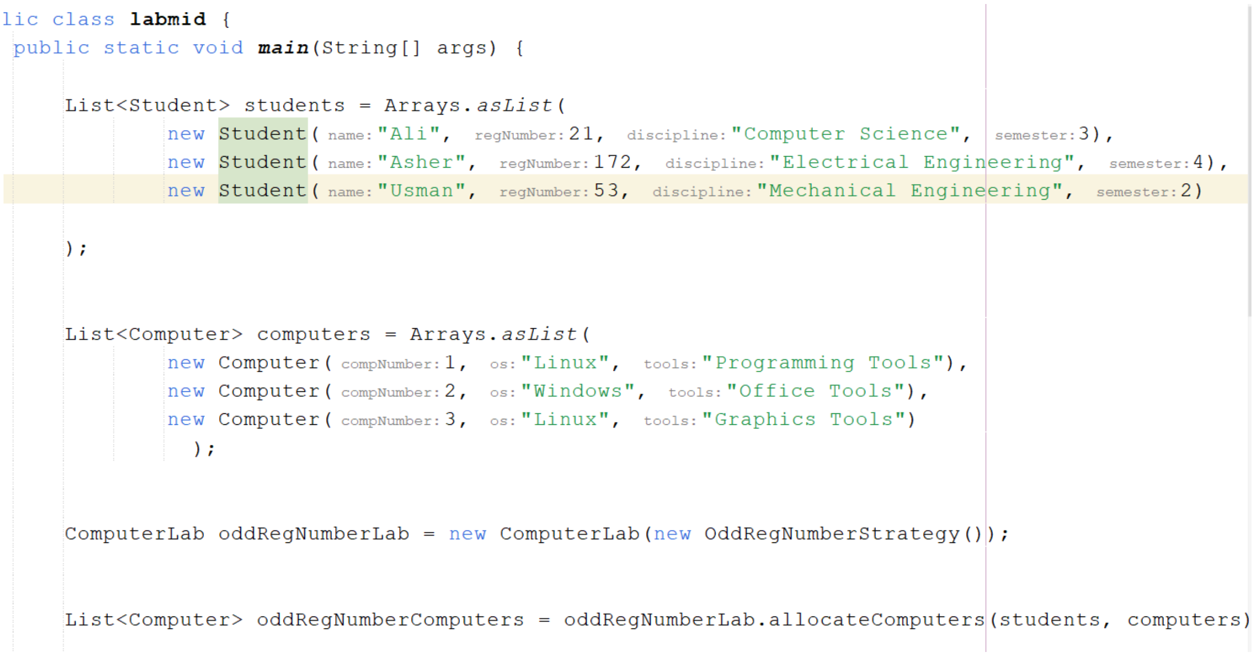
```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this lice
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package labmid;

/**
 *
 * @author Lenovo
 */
import java.util.List;

public class ComputerLab {
    private final AllocationStrategy allocationStrategy;

    public ComputerLab(AllocationStrategy allocationStrategy) {
        this.allocationStrategy = allocationStrategy;
    }

    public List<Computer> allocateComputers(List<Student> students, List<Computer> computers) {
        return allocationStrategy.allocateComputers(students, computers);
    }
}
```



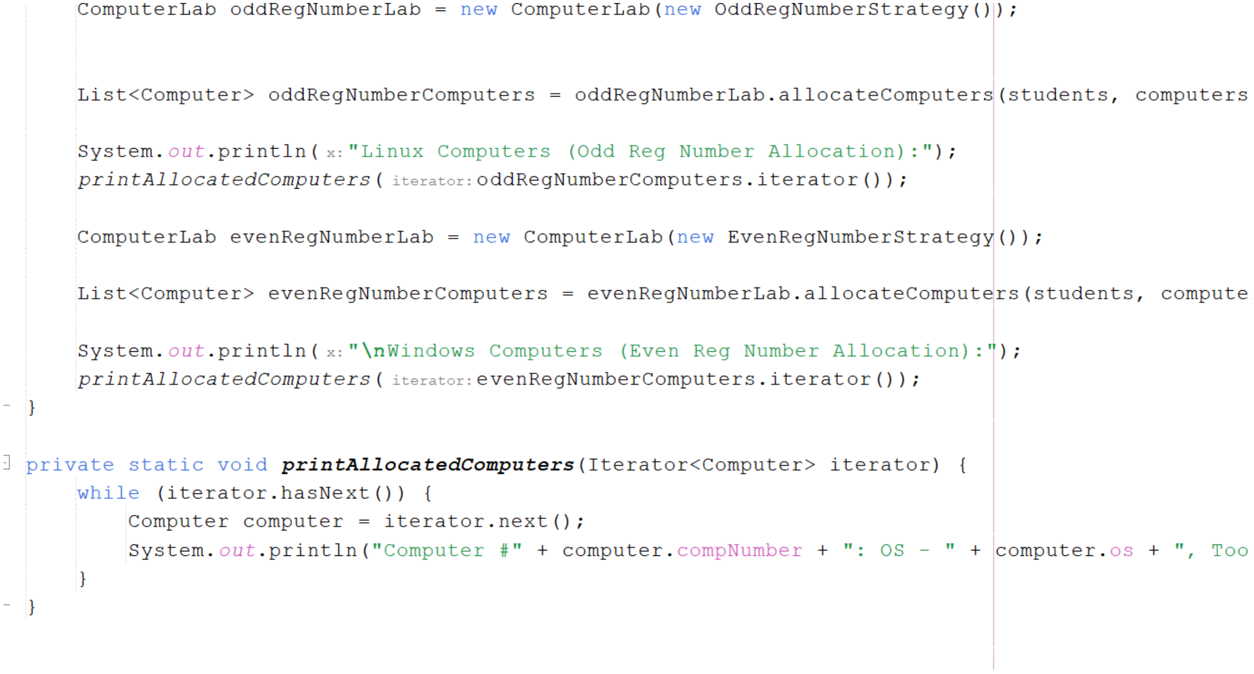
```java
/**
 *
 * @author Lenovo
 */
import java.util.ArrayList;
import java.util.List;

public class EvenRegNumberStrategy implements AllocationStrategy {
    @Override
    public List<Computer> allocateComputers(List<Student> students, List<Computer> computers) {
        List<Computer> allocatedComputers = new ArrayList<>();

        for (int i = 0; i < students.size(); i++) {
            if (students.get( index: i).regNumber % 2 == 0) {
                allocatedComputers.add( e: computers.get( index: i));
            }
        }

        return allocatedComputers;
    }
}
```



```java
package labmid;

/**
 *
 * @author Lenovo
 */
import java.util.ArrayList;
import java.util.List;

public class OddRegNumberStrategy implements AllocationStrategy {
    @Override
    public List<Computer> allocateComputers(List<Student> students, List<Computer> computers) {
        List<Computer> allocatedComputers = new ArrayList<>();

        for (int i = 0; i < students.size(); i++) {
            if (students.get( index: i).regNumber % 2 != 0) {
                allocatedComputers.add( e: computers.get( index: i));
            }
        }

        return allocatedComputers;
    }
```



```
lic class labmid {
 public static void main(String[] args) {

     List<Student> students = Arrays.asList(
             new Student( name: "Ali",  regNumber: 21,  discipline: "Computer Science",  semester: 3),
             new Student( name: "Asher",  regNumber: 172,  discipline: "Electrical Engineering",  semester: 4),
             new Student( name: "Usman",  regNumber: 53,  discipline: "Mechanical Engineering",  semester: 2)

     );


     List<Computer> computers = Arrays.asList(
             new Computer( compNumber: 1,  os: "Linux",  tools: "Programming Tools"),
             new Computer( compNumber: 2,  os: "Windows",  tools: "Office Tools"),
             new Computer( compNumber: 3,  os: "Linux",  tools: "Graphics Tools")
               );


     ComputerLab oddRegNumberLab = new ComputerLab(new OddRegNumberStrategy());


     List<Computer> oddRegNumberComputers = oddRegNumberLab.allocateComputers(students, computers)
```



```
     ComputerLab oddRegNumberLab = new ComputerLab(new OddRegNumberStrategy());


     List<Computer> oddRegNumberComputers = oddRegNumberLab.allocateComputers(students, computers

     System.out.println( x: "Linux Computers (Odd Reg Number Allocation):");
     printAllocatedComputers( iterator: oddRegNumberComputers.iterator());

     ComputerLab evenRegNumberLab = new ComputerLab(new EvenRegNumberStrategy());

     List<Computer> evenRegNumberComputers = evenRegNumberLab.allocateComputers(students, compute

     System.out.println( x: "\nWindows Computers (Even Reg Number Allocation):");
     printAllocatedComputers( iterator: evenRegNumberComputers.iterator());
 }

 private static void printAllocatedComputers(Iterator<Computer> iterator) {
     while (iterator.hasNext()) {
         Computer computer = iterator.next();
         System.out.println("Computer #" + computer.compNumber + ": OS - " + computer.os + ", Too
     }
 }
```