**Riphah International University Islamabad**

**LAB MANUAL**

**OBJET ORIENTED PROGRAMMING**

**Lab Exercise:**

**Task 1:**

**import java.util.\*;**

**// Student class**

**class Student {**

**private int studentId;**

**private String name;**

**public Student(int studentId, String name) {**

**this.studentId = studentId;**

**this.name = name;**

**}**

**public int getStudentId() {**

**return studentId;**

**}**

**public String getName() {**

**return name;**

**}**

**public void displayStudent() {**

**System.out.println("Student ID: " + studentId + ", Name: " + name);**

**}**

**}**

**// College class (Aggregation with Student)**

**class College {**

**private String collegeName;**

**private List<Student> students;**

**public College(String collegeName) {**

**this.collegeName = collegeName;**

**this.students = new ArrayList<>();**

**}**

**public void addStudent(Student student) {**

**students.add(student);**

**}**

**public void displayCollegeDetails() {**

**System.out.println("College: " + collegeName);**

**System.out.println("Students:");**

**for (Student s : students) {**

**s.displayStudent();**

**}**

**}**

**}**

**// Main class**

**public class AggregationExample {**

**public static void main(String[] args) {**

**// Creating Students**

**Student s1 = new Student(101, "Alice");**

**Student s2 = new Student(102, "Bob");**

**Student s3 = new Student(103, "Charlie");**

**// Creating College**

**College college = new College("Tech University");**

**// Associating Students with College**

**college.addStudent(s1);**

**college.addStudent(s2);**

**college.addStudent(s3);**

**// Display details**

**college.displayCollegeDetails();**

**}**

**}**

**Task 2:**

// Brain class (Dependent on Person)

class Brain {

public void think() {

System.out.println("Thinking... Deep thoughts in progress!");

}

}

// Person class (Composition with Brain)

class Person {

private Brain brain; // Brain is part of Person (Composition)

public Person() {

this.brain = new Brain(); // Brain is created within Person

}

public void think() {

brain.think();

}

}

// Main class

public class CompositionExample {

public static void main(String[] args) {

// Creating a Person object

Person person = new Person();

// Calling think() method using Brain object

person.think();

// Deleting the Person object (setting to null)

person = null;

System.gc(); // Request garbage collection (for demonstration)

System.out.println("Person object deleted. Brain no longer exists.");

}

}