Lab-Report

Exp. Name : Association and Aggregation UML Code

Course Code : **CSE-326**

Course Title : **System Analysis and Design Lab**

Submitted To: Sputa Richard Philip

Senior Lecture

City University

Department of computer Science and Engineering

Faculty of Science and Engineering

Submitted By:

ID :171442560

Name :Md. Rasel Miah

Program :CSE(Eve)

Batch : 44th

Semester :Spring

Association

Association is a relationship between two separate classes that establishes through their objects. Each object have their own life-cycle and there is no owner. Association can be one-to-one, one-to-many, many-to-one, many-to-many.

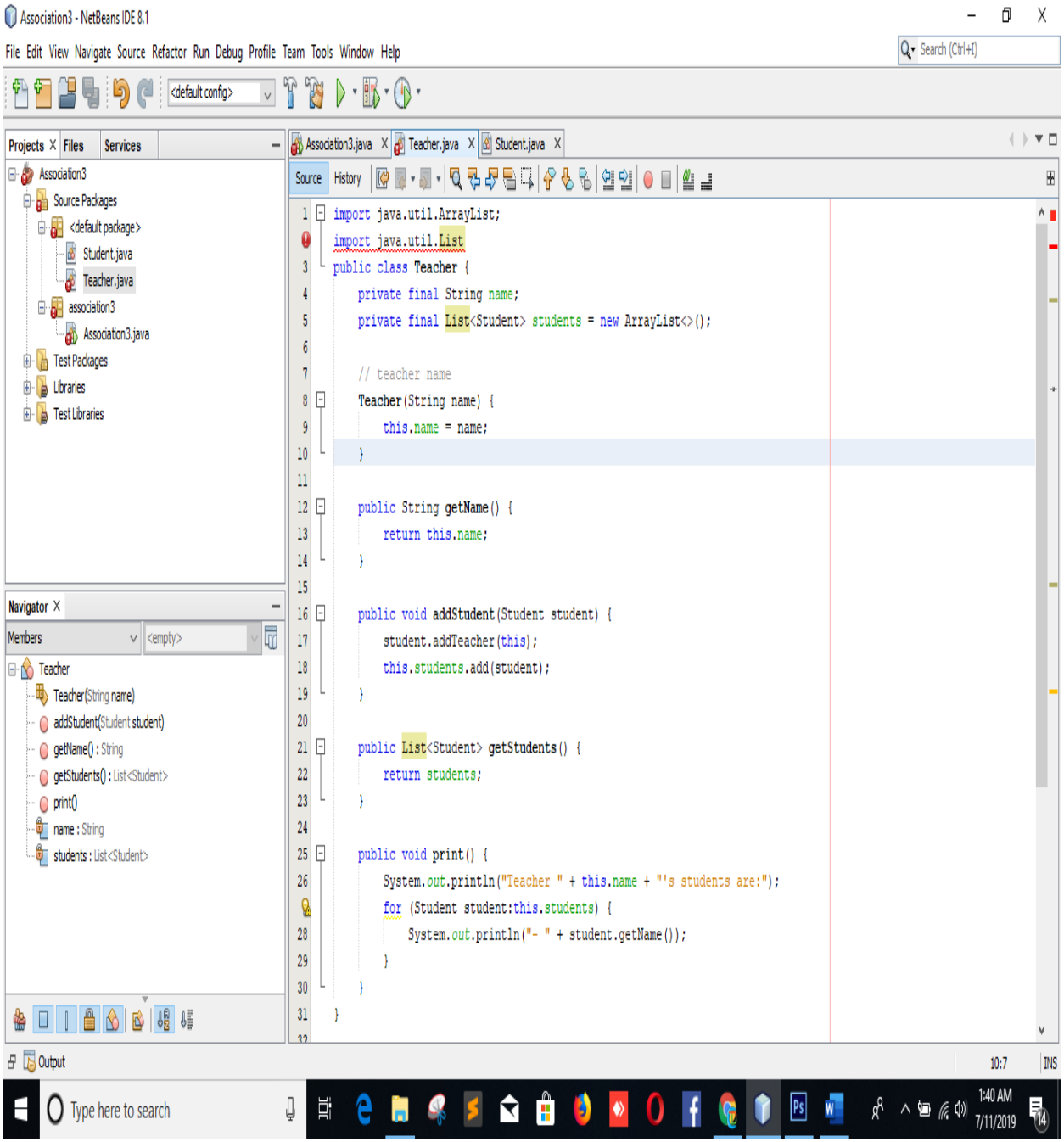
Let's take an example of Teacher and Student. Multiple students can associate with single teacher and single student can associate with multiple teachers, but there is no ownership between the objects and both have their own life-cycle. Both can be created and deleted independently.

Teacher.java:

```
import java.util.ArrayList;

import java.util.List

public class Teacher {

    private final String name;

    private final List<Student> students = new
ArrayList<>();
```

```
// teacher name
Teacher(String name) {
    this.name = name;
}

public String getName() {
    return this.name;
}

public void addStudent(Student student) {
    student.addTeacher(this);
    this.students.add(student);
}

public List<Student> getStudents() {
```

```
        return students;
    }

    public void print() {
        System.out.println("Teacher " + this.name + "'s
students are:");
        for (Student student:this.students) {
            System.out.println("- " + student.getName());
        }
    }
}
```

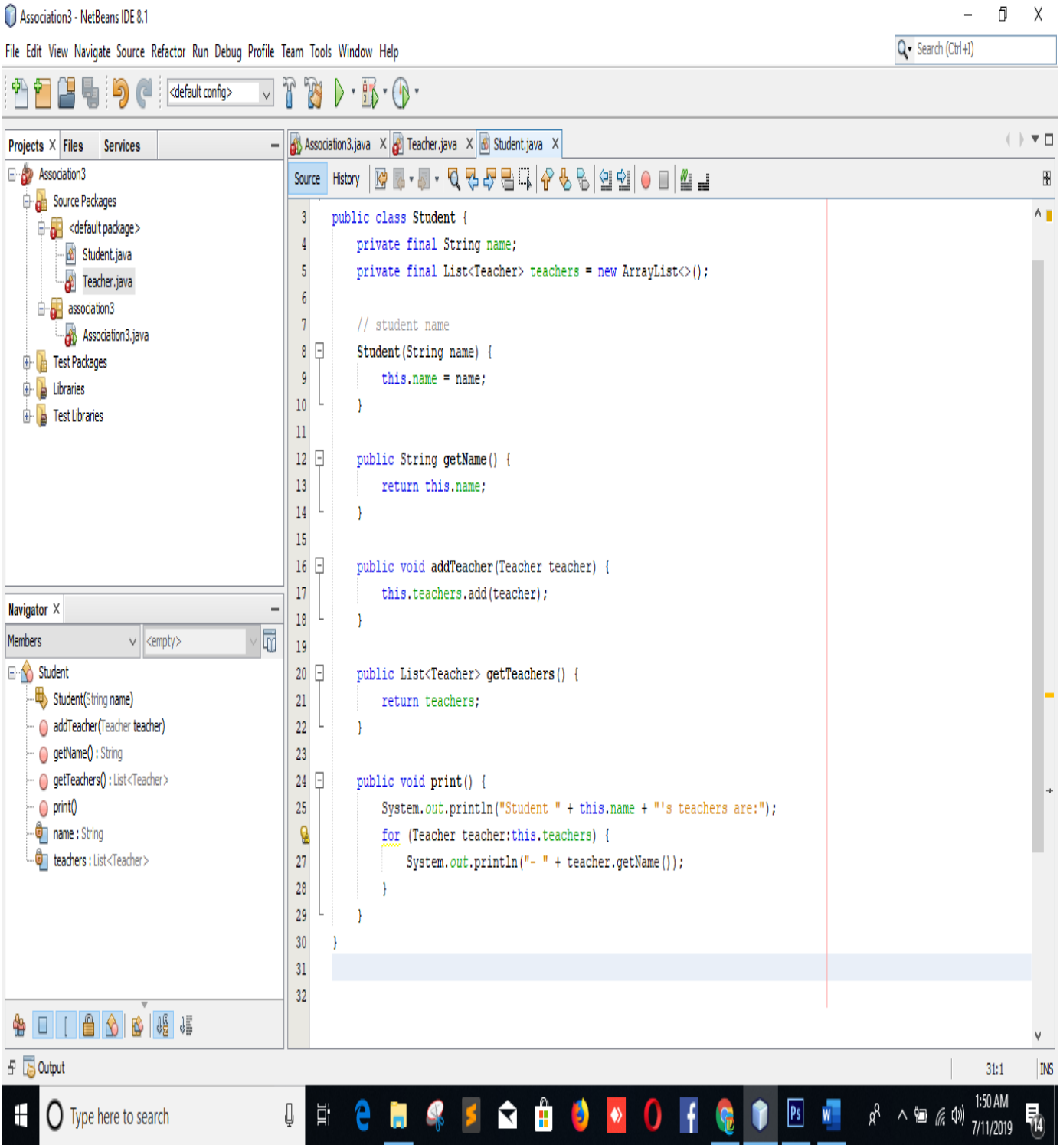

Association3 - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects
Files
Services
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Libraries
Test Libraries
Navigator
Members
<empty>
Teacher
Teacher(String name)
addStudent(Student student)
getName() : String
getStudents() : List<Student>
print()
name : String
students : List<Student>
Association3.java
Teacher.java
Student.java
Source
History
1 import java.util.ArrayList;
2 import java.util.List
3 public class Teacher {
4 private final String name;
5 private final List<Student> students = new ArrayList<>();
6
7 // teacher name
8 Teacher(String name) {
9 this.name = name;
10 }
11
12 public String getName() {
13 return this.name;
14 }
15
16 public void addStudent(Student student) {
17 student.addTeacher(this);
18 this.students.add(student);
19 }
20
21 public List<Student> getStudents() {
22 return students;
23 }
24
25 public void print() {
26 System.out.println("Teacher " + this.name + "'s students are:");
27 for (Student student:this.students) {
28 System.out.println("- " + student.getName());
29 }
30 }
31 }
Output
10:7
INS
Type here to search
1:40 AM
7/11/2019

Student.java:

```
import java.util.ArrayList;

import java.util.List;

public class Student {

    private final String name;

    private final List<Teacher> teachers = new
ArrayList<>();


    // student name

    Student(String name) {

        this.name = name;

    }


    public String getName() {

        return this.name;

    }
```

```
    public void addTeacher(Teacher teacher) {
        this.teachers.add(teacher);
    }

    public List<Teacher> getTeachers() {
        return teachers;
    }

    public void print() {
        System.out.println("Student " + this.name + "'s
teachers are:");
        for (Teacher teacher:this.teachers) {
            System.out.println("- " + teacher.getName());
        }
    }
```

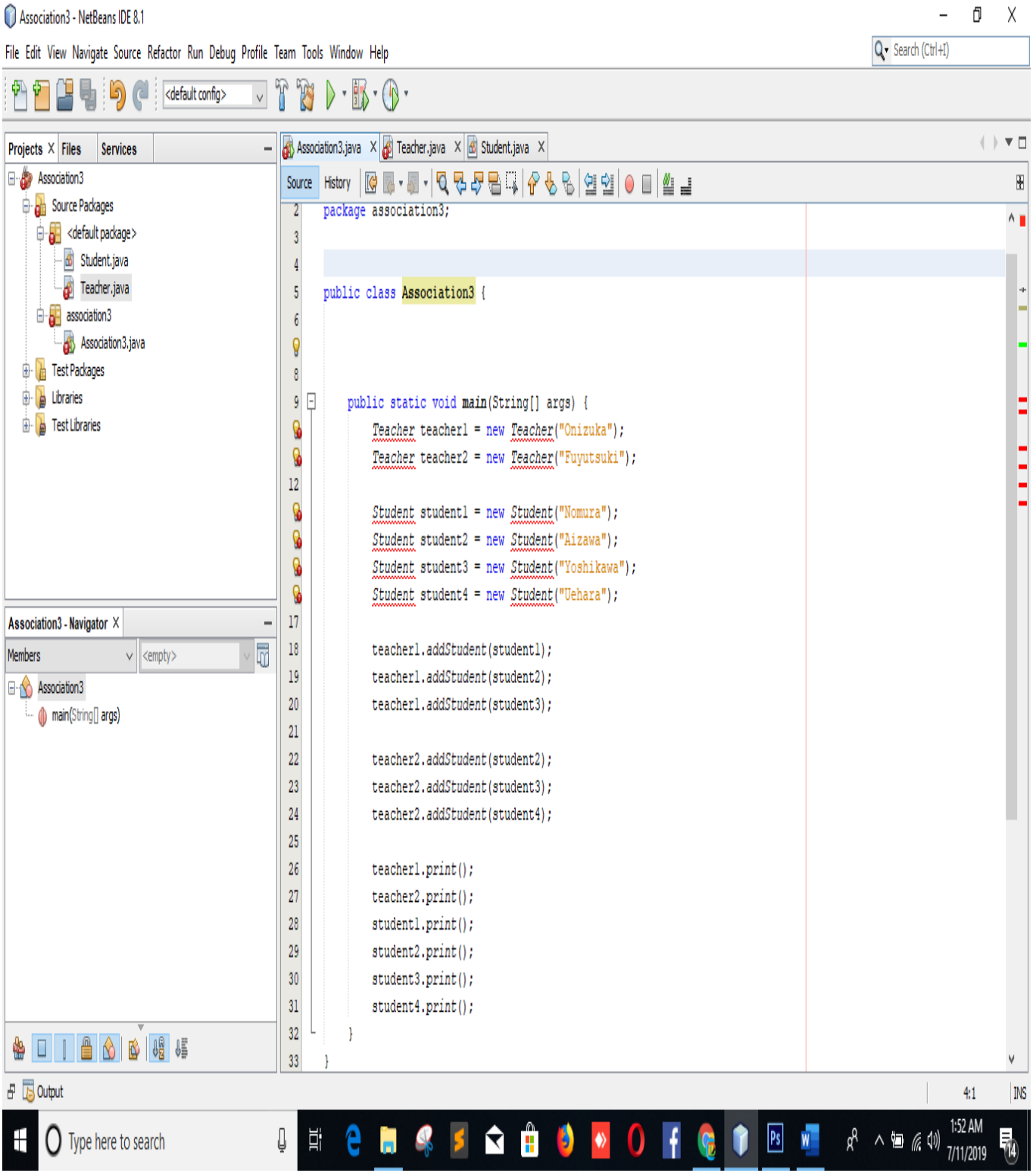
```
}
```


Association3 - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects
Files
Services
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Libraries
Test Libraries
Navigator
Members
<empty>
Student
Student(String name)
addTeacher(Teacher teacher)
getName() : String
getTeachers() : List<Teacher>
print()
name : String
teachers : List<Teacher>
Association3.java
Teacher.java
Student.java
Source
History
3 public class Student {
4 private final String name;
5 private final List<Teacher> teachers = new ArrayList<>();
6
7 // student name
8 Student(String name) {
9 this.name = name;
10 }
11
12 public String getName() {
13 return this.name;
14 }
15
16 public void addTeacher(Teacher teacher) {
17 this.teachers.add(teacher);
18 }
19
20 public List<Teacher> getTeachers() {
21 return teachers;
22 }
23
24 public void print() {
25 System.out.println("Student " + this.name + "'s teachers are:");
for (Teacher teacher:this.teachers) {
27 System.out.println("- " + teacher.getName());
28 }
29 }
30 }
31
32
Output
31:1
INS
Type here to search
1:50 AM
7/11/2019

Association.java

```
package association3;


public class Association3 {


    public static void main(String[] args) {
        Teacher teacher1 = new Teacher("Onizuka");
        Teacher teacher2 = new Teacher("Fuyutsuki");

        Student student1 = new Student("Nomura");
        Student student2 = new Student("Aizawa");
        Student student3 = new Student("Yoshikawa");
        Student student4 = new Student("Uehara");
```

```
        teacher1.addStudent(student1);
        teacher1.addStudent(student2);
        teacher1.addStudent(student3);

        teacher2.addStudent(student2);
        teacher2.addStudent(student3);
        teacher2.addStudent(student4);

        teacher1.print();
        teacher2.print();
        student1.print();
        student2.print();
        student3.print();
        student4.print();
    }
}
```


Association3 - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects
Files
Services
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Libraries
Test Libraries
Association3.java
Teacher.java
Student.java
Source
History
package association3;
public class Association3 {
public static void main(String[] args) {
Teacher teacher1 = new Teacher("Onizuka");
Teacher teacher2 = new Teacher("Fuyutsuki");
Student student1 = new Student("Nomura");
Student student2 = new Student("Aizawa");
Student student3 = new Student("Yoshikawa");
Student student4 = new Student("Uehara");
teacher1.addStudent(student1);
teacher1.addStudent(student2);
teacher1.addStudent(student3);
teacher2.addStudent(student2);
teacher2.addStudent(student3);
teacher2.addStudent(student4);
teacher1.print();
teacher2.print();
student1.print();
student2.print();
student3.print();
student4.print();
}
}
Association3 - Navigator
Members
<empty>
Association3
main(String[] args)
Output
4:1
INS
Type here to search
1:52 AM
7/11/2019

Aggregation:

Aggregation is a specialized form of Association where all objects have their own life cycle, where the child can exist independently of the parent. Aggregation is also called a "**Has-a**" relationship.

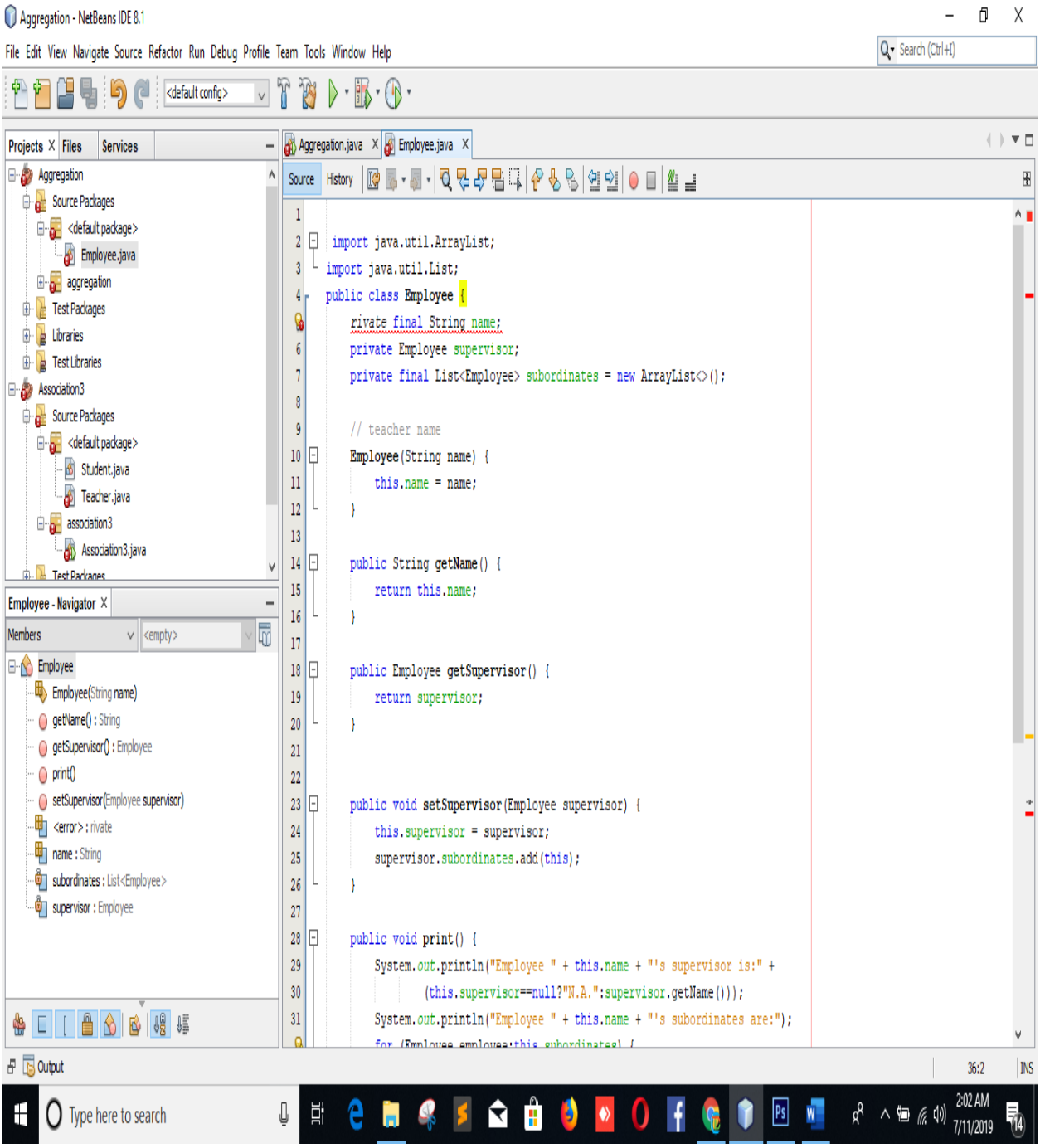
Let's take an example of Supervisor and Subordinate. An employee (as a subordinate) can not belong to multiple supervisors, but if we delete the supervisor, the employee object (subordinate) will *not* be destroyed. We can think about it as a "**has-a**" relationship.

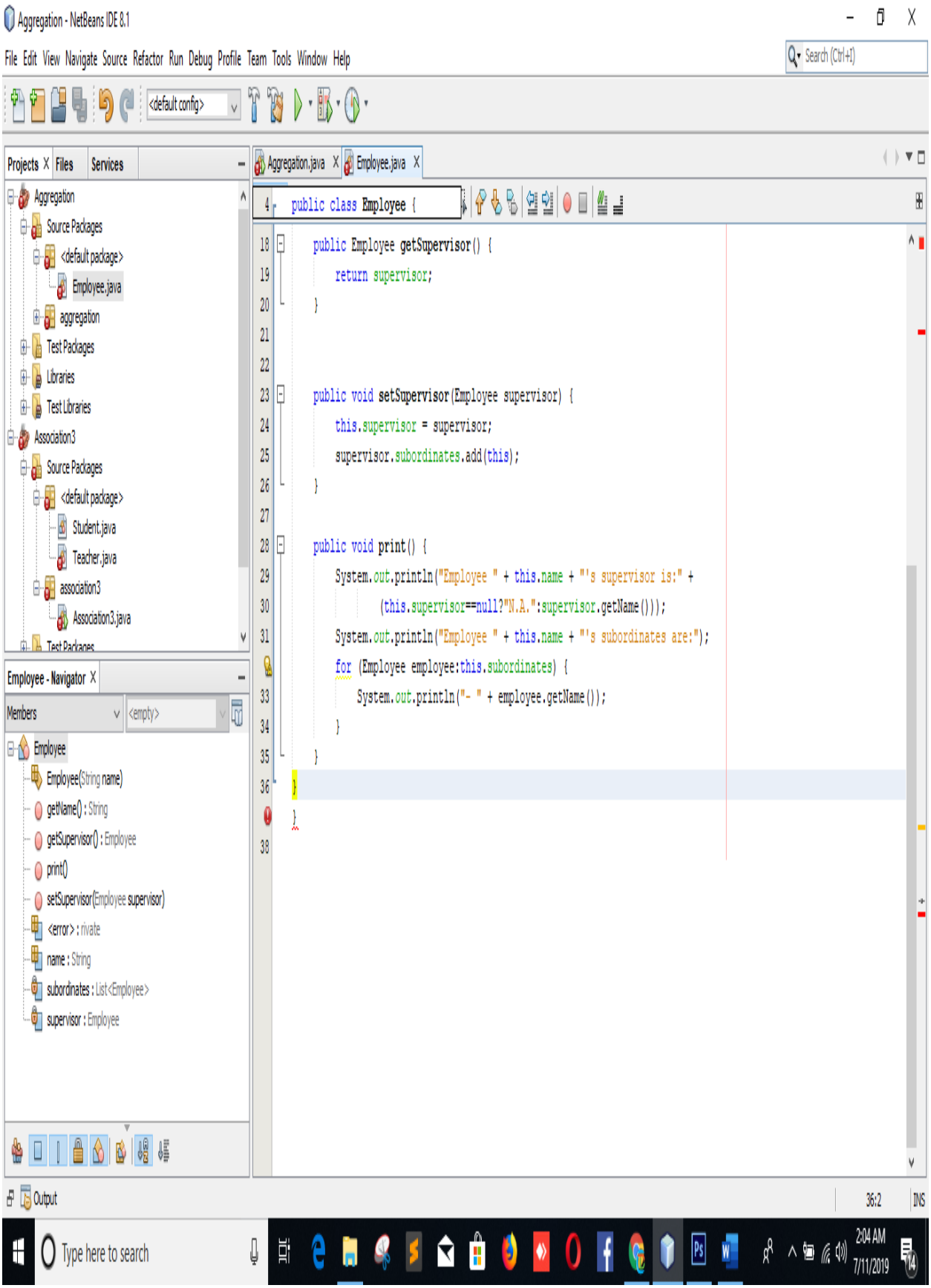
Employee.java

```
 import java.util.ArrayList;

import java.util.List;

public class Employee {

    rivate final String name;

    private Employee supervisor;

    private final List<Employee> subordinates = new
ArrayList<>();


    // teacher name

    Employee(String name) {

        this.name = name;
```

```
    }

    public String getName() {
        return this.name;
    }

    public Employee getSupervisor() {
        return supervisor;
    }


    public void setSupervisor(Employee supervisor) {
        this.supervisor = supervisor;
        supervisor.subordinates.add(this);
    }
```

```
    public void print() {

        System.out.println("Employee " + this.name +
"'s supervisor is:" +


(this.supervisor==null?"N.A.":supervisor.getName()))
;

        System.out.println("Employee " + this.name +
"'s subordinates are:");

        for (Employee employee:this.subordinates) {

            System.out.println("- " +
employee.getName());

        }

    }

}

}
```


Aggregation - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects X Files Services
Aggregation
Source Packages
<default package>
Employee.java
aggregation
Test Packages
Libraries
Test Libraries
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Employee - Navigator X
Members
<empty>
Employee
Employee(String name)
getName() : String
getSupervisor() : Employee
print()
setSupervisor(Employee supervisor)
<error> : rivate
name : String
subordinates : List<Employee>
supervisor : Employee
Aggregation.java X
Employee.java X
Source History
1 import java.util.ArrayList;
2 import java.util.List;
3 public class Employee {
4 rivate final String name;
5 private Employee supervisor;
6 private final List<Employee> subordinates = new ArrayList<>();
7
8 // teacher name
9 Employee(String name) {
10 this.name = name;
11 }
12
13 public String getName() {
14 return this.name;
15 }
16
17 public Employee getSupervisor() {
18 return supervisor;
19 }
20
21
22 public void setSupervisor(Employee supervisor) {
23 this.supervisor = supervisor;
24 supervisor.subordinates.add(this);
25 }
26
27 public void print() {
28 System.out.println("Employee " + this.name + "'s supervisor is:" +
29 (this.supervisor==null?"N.A.":supervisor.getName()));
30 System.out.println("Employee " + this.name + "'s subordinates are:");
31 for (Employee employee:this.subordinates) {
Output
36:2
INS
Type here to search
2:02 AM
7/11/2019

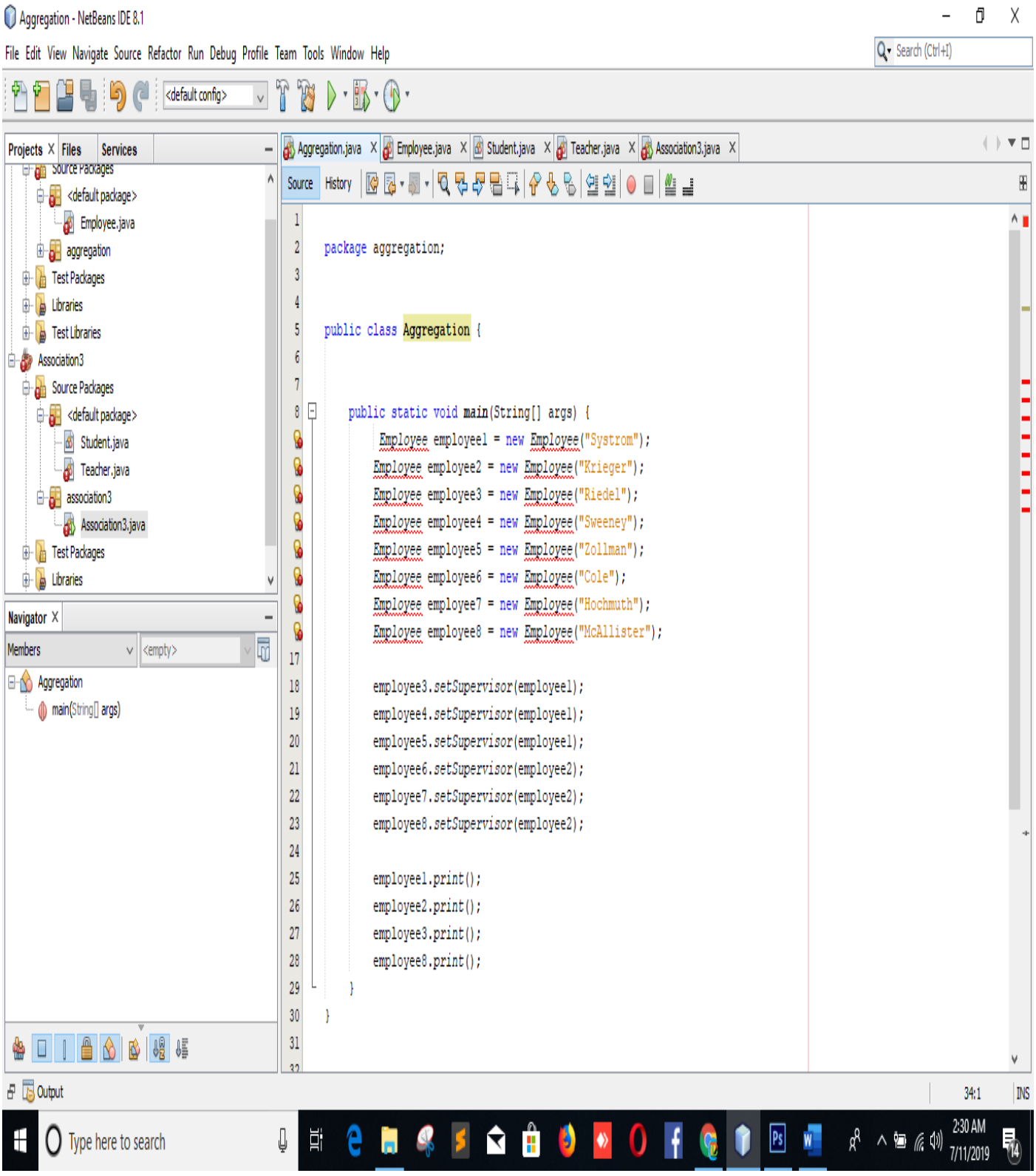

Aggregation - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects
Files
Services
Aggregation
Source Packages
<default package>
Employee.java
aggregation
Test Packages
Libraries
Test Libraries
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Employee - Navigator
Members
<empty>
Employee
Employee(String name)
getName() : String
getSupervisor() : Employee
print()
setSupervisor(Employee supervisor)
<error> : rivate
name : String
subordinates : List<Employee>
supervisor : Employee
Aggregation.java
Employee.java
public class Employee {
public Employee getSupervisor() {
return supervisor;
}
public void setSupervisor(Employee supervisor) {
this.supervisor = supervisor;
supervisor.subordinates.add(this);
}
public void print() {
System.out.println("Employee " + this.name + "'s supervisor is:" +
(this.supervisor==null?"N.A.":supervisor.getName()));
System.out.println("Employee " + this.name + "'s subordinates are:");
for (Employee employee:this.subordinates) {
System.out.println("- " + employee.getName());
}
}
}
Output
36:2
INS
Type here to search
2:04 AM
7/11/2019

Aggregation

```
package aggregation;



public class Aggregation {



    public static void main(String[] args) {

         Employee employee1 = new
Employee("Systrom");

        Employee employee2 = new
Employee("Krieger");

        Employee employee3 = new
Employee("Riedel");

        Employee employee4 = new
Employee("Sweeney");
```

```
        Employee employee5 = new
Employee("Zollman");

        Employee employee6 = new Employee("Cole");

        Employee employee7 = new
Employee("Hochmuth");

        Employee employee8 = new
Employee("McAllister");



        employee3.setSupervisor(employee1);

        employee4.setSupervisor(employee1);

        employee5.setSupervisor(employee1);

        employee6.setSupervisor(employee2);

        employee7.setSupervisor(employee2);

        employee8.setSupervisor(employee2);



        employee1.print();

        employee2.print();
```

```
        employee3.print();
        employee8.print();
    }
}
```


Aggregation - NetBeans IDE 8.1
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
<default config>
Projects
Files
Services
Source Packages
<default package>
Employee.java
aggregation
Test Packages
Libraries
Test Libraries
Association3
Source Packages
<default package>
Student.java
Teacher.java
association3
Association3.java
Test Packages
Libraries
Navigator
Members
<empty>
Aggregation
main(String[] args)
Aggregation.java
Employee.java
Student.java
Teacher.java
Association3.java
Source
History
1 package aggregation;
3 public class Aggregation {
5 public static void main(String[] args) {
6 Employee employee1 = new Employee("Systrom");
7 Employee employee2 = new Employee("Krieger");
8 Employee employee3 = new Employee("Riedel");
9 Employee employee4 = new Employee("Sweeney");
10 Employee employee5 = new Employee("Zollman");
11 Employee employee6 = new Employee("Cole");
12 Employee employee7 = new Employee("Hochmuth");
13 Employee employee8 = new Employee("McAllister");
15 employee3.setSupervisor(employee1);
16 employee4.setSupervisor(employee1);
17 employee5.setSupervisor(employee1);
18 employee6.setSupervisor(employee2);
19 employee7.setSupervisor(employee2);
20 employee8.setSupervisor(employee2);
22 employee1.print();
23 employee2.print();
24 employee3.print();
25 employee8.print();
26 }
27 }
Output
34:1
INS
Type here to search
2:30 AM
7/11/2019