

Диаграмма классов

Visual Paradigm

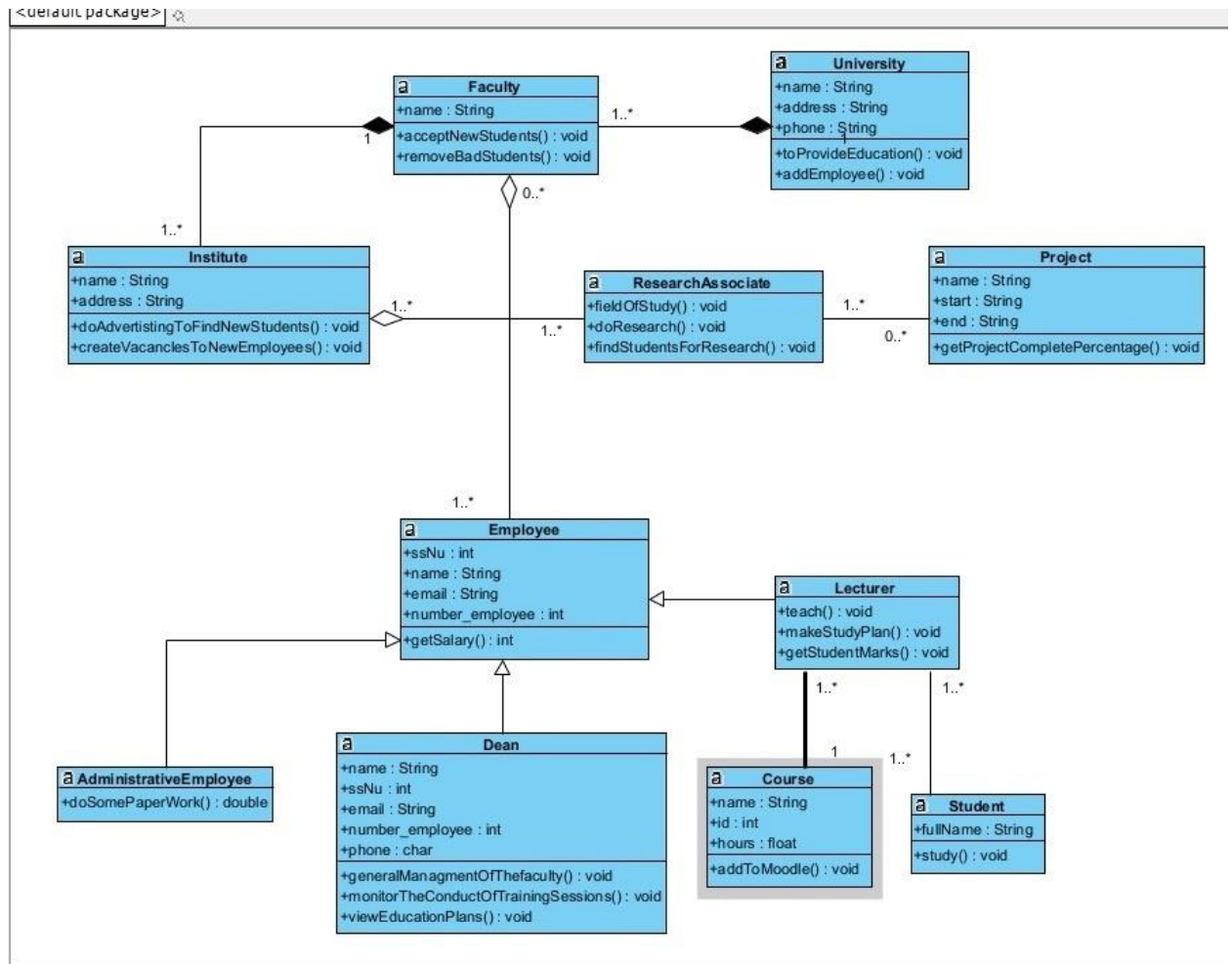
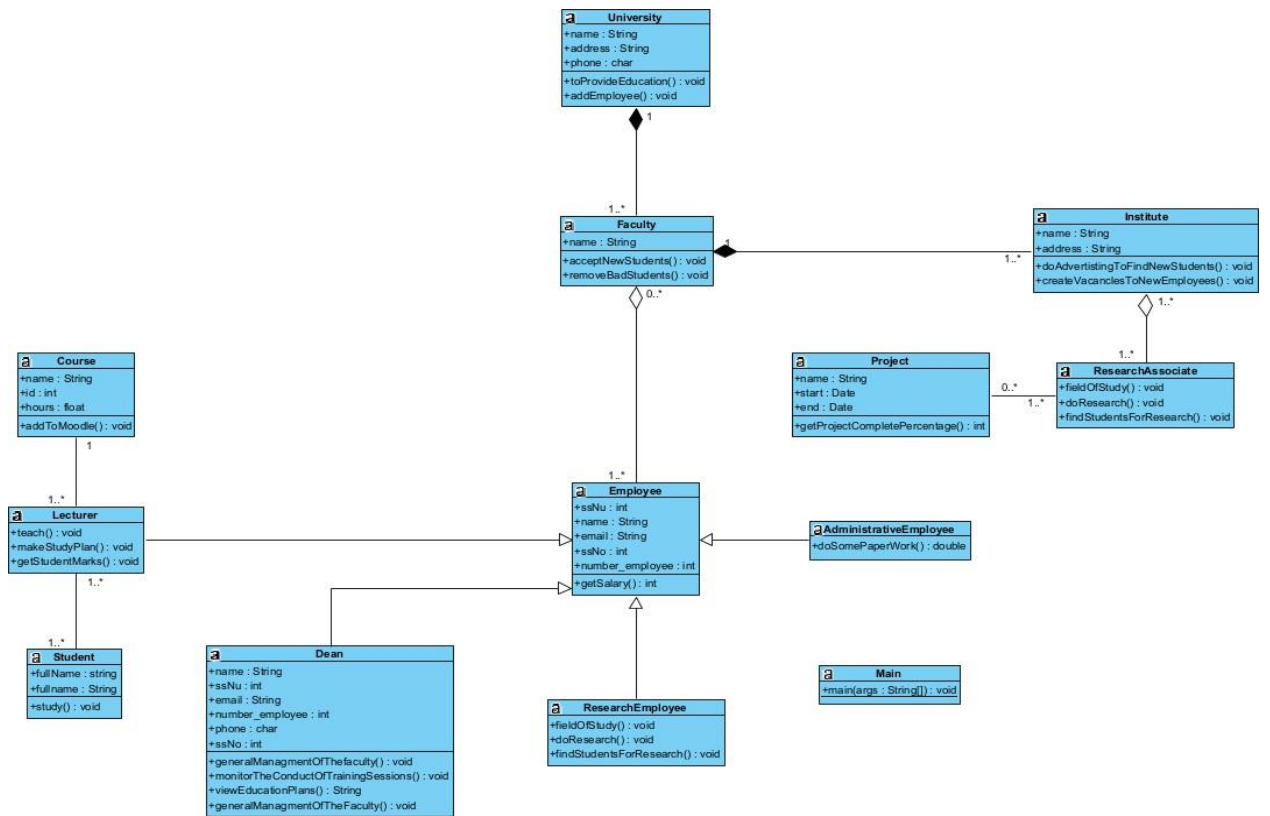


Диаграмма классов

Преобразование кода с помощью Visual Paradigm



Use Case

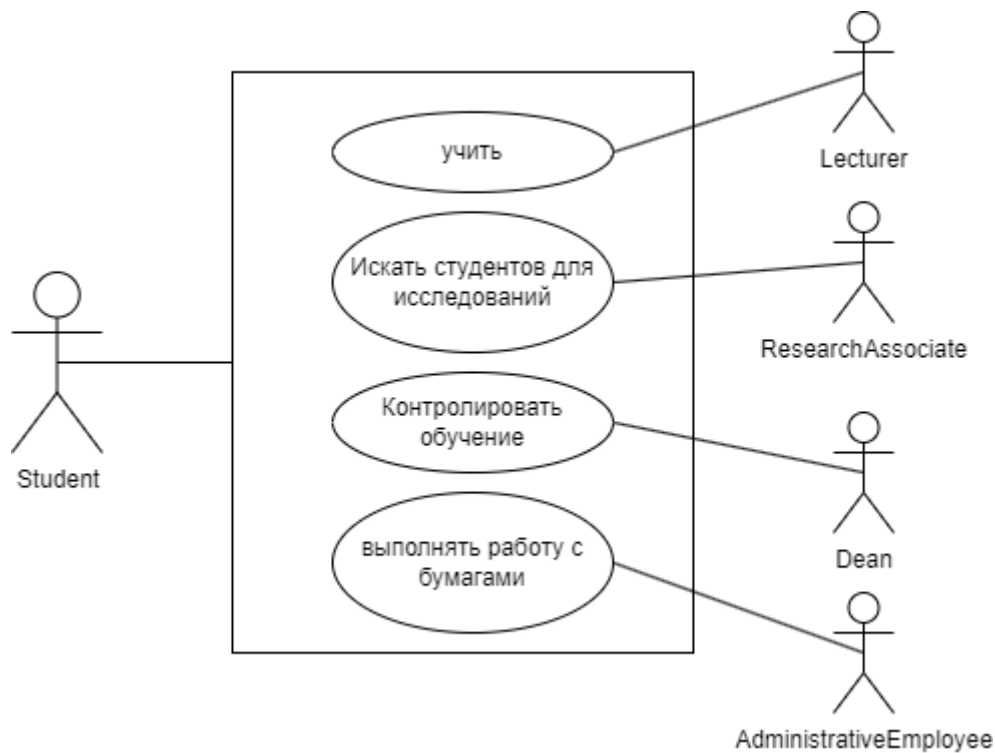


Диаграмма последовательности обучения

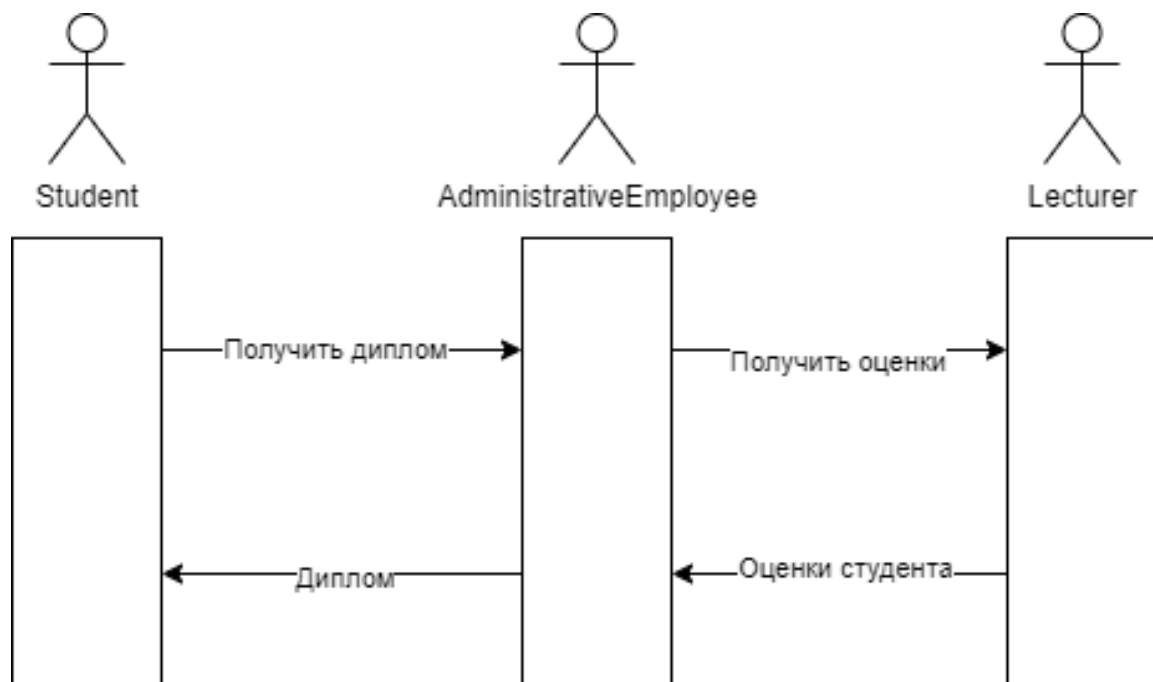
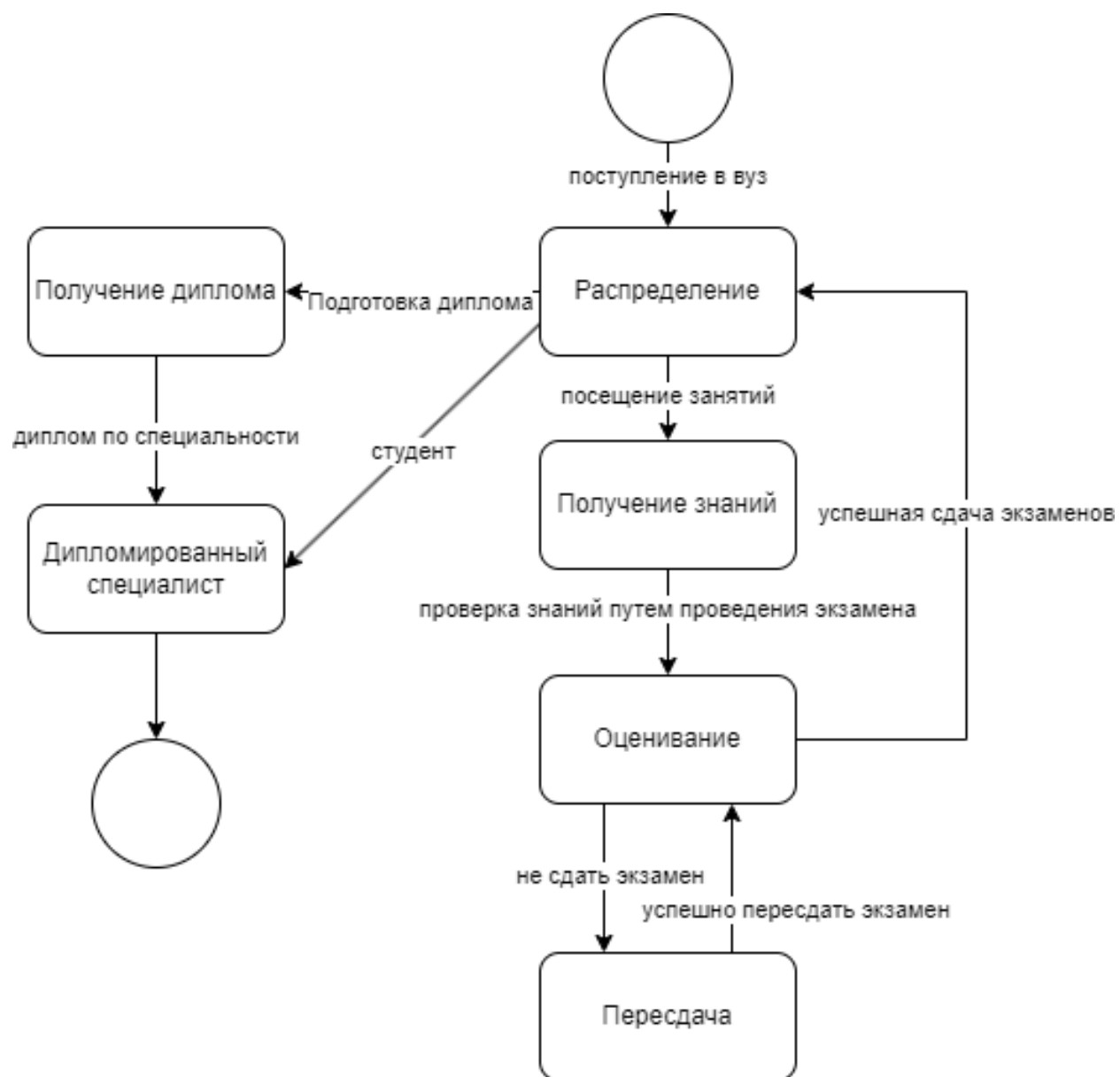


Диаграмма состояний



Код и результат компиляции

```
import java.util.Scanner;
```

no usages

```
public class AdministrativeEmployee extends Employee {
```

no usages

```
    public double doSomePaperWork() {
```

```
        // TODO - implement AdministrativeEmployee.doSomePaperWork
```

```
        throw new UnsupportedOperationException();
```

```
    }
```



```
}
```

no usages

```
public class Course {
```



no usages

```
    public String name;
```

no usages

```
    public int id;
```

no usages

```
    public float hours;
```

no usages

```
    public void addToMoodle() {
```

```
        // TODO - implement Course.addToMoodle
```

```
        throw new UnsupportedOperationException();
```

```
    }
```

```
}
```

```
public class Dean extends Employee {
```



no usages

```
public String name;
```

no usages

```
public int ssNu;
```

no usages

```
public String email;
```

no usages

```
public int number_employee;
```

no usages

```
public char phone;
```

no usages

```
public void generalManagmentOfThefaculty() {
```

```
    // TODO - implement Dean.generalManagmentOfThefaculty
```

```
    throw new UnsupportedOperationException();
```

```
}
```

no usages

```
public void monitorTheConductOfTrainingSessions() {
```

```
    // TODO - implement Dean.monitorTheConductOfTrainingSessions
```

```
    throw new UnsupportedOperationException();
```

```
}
```

no usages

```
public void viewEducationPlans() {
```

```
    // TODO - implement Dean.viewEducationPlans
```

```
    throw new UnsupportedOperationException();
```

```
}
```

```
}
```

3 usages 4 inheritors

```
public class Employee {
```



no usages

```
public int ssNu;
```

no usages

```
public String name;
```

no usages

```
public String email;
```

no usages

```
public void getSalary() {
```

```
    // TODO - implement Employee.getSalary
```

```
    throw new UnsupportedOperationException();
```

```
}
```

```
}
```

no usages

```
public class Faculty {
```



no usages

```
public String name;
```

no usages

```
public void acceptNewStudents() {
```

```
    // TODO - implement Faculty.acceptNewStudents
```

```
    throw new UnsupportedOperationException();
```

```
}
```

no usages

```
public void removeBadStudents() {
```

```
    // TODO - implement Faculty.removeBadStudents
```

```
    throw new UnsupportedOperationException();
```

```
}
```

```
}
```

no usages

```
public class Institute {
```



no usages

```
public String name;
```

no usages

```
public String address;
```

no usages

```
public void doAdvertisingToFindNewStudents() {  
    // TODO - implement Institute.doAdvertisingToFindNewStudents  
    throw new UnsupportedOperationException();  
}
```

no usages

```
public void createVacanciesToNewEmployees() {  
    // TODO - implement Institute.createVacanciesToNewEmployees  
    throw new UnsupportedOperationException();  
}
```

```
}
```


no usages

```
public class Lecturer extends ResearchAssociate {
```



no usages

```
public void teach() {  
    // TODO - implement Lecturer.teach  
    throw new UnsupportedOperationException();  
}
```

no usages

```
public void makeStudyPlan() {  
    // TODO - implement Lecturer.makeStudyPlan  
    throw new UnsupportedOperationException();  
}
```

no usages

```
public void getStudentMarks() {  
    // TODO - implement Lecturer.getStudentMarks  
    throw new UnsupportedOperationException();  
}
```

```
}
```

no usages

```
public class Project {
```



no usages

```
public String name;
```

no usages

```
public String start;
```

no usages

```
public String end;
```

no usages

```
public void getProjectCompletePercentage() {
```

```
    // TODO - implement Project.getProjectCompletePercentage
```

```
    throw new UnsupportedOperationException();
```

```
}
```

```
}
```

```

import java.util.Scanner;

3 usages 1 inheritor
public class ResearchAssociate extends Employee {

    no usages
    public void fieldOfStudy() {
        // TODO - implement ResearchAssociate.fieldOfStudy
        throw new UnsupportedOperationException();
    }

    no usages
    public void doResearch() {
        // TODO - implement ResearchAssociate.doResearch
        throw new UnsupportedOperationException();
    }

    1 usage
    public double findStudentsForResearch() {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Введите количество студентов для исследования - ");
        int list = scanner.nextInt();

        System.out.println("Введите количество дней для исследования одного студента - ");
        int speed = scanner.nextInt();


        return (double) list * speed;
    }
}


```

```

Введите количество студентов для исследования -
10
Введите количество дней для исследования одного студента -
2
Уоличество дней на исследование студентов 20.0 дней.

```

```
public class Student {  
     no usages  
    public String fullName;  
  
    no usages  
    public void study() {  
        // TODO - implement Student.study  
        throw new UnsupportedOperationException();  
    }  
}
```

```
no usages  
public class University {  
     no usages  
    public String name;  
    no usages  
    public String address;  
    no usages  
    public String phone;  
  
    no usages  
    public void toProvideEducation() {  
        // TODO - implement University.toProvideEducation  
        throw new UnsupportedOperationException();  
    }  
  
    no usages  
    public void addEmployee() {  
        // TODO - implement University.addEmployee  
        throw new UnsupportedOperationException();  
    }  
}
```