



## 2.2. QUẢN LÝ SINH VIÊN CHO TRUNG TÂM TIN HỌC.

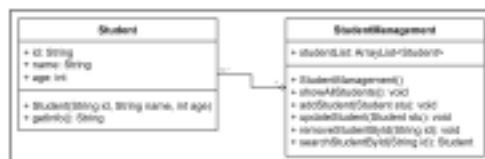


## 2.2. QUẢN LÝ SINH VIÊN CHO TRUNG TÂM TIN HỌC.

### ĐỀ BÀI.

Thực hiện các yêu cầu.

Cài đặt theo sơ đồ lớp sau:



Chi tiết về các lớp:

Lớp Student (Sinh viên):

Trường dữ liệu (field):

`public String id;` Mã số sinh viên.

`public String name;` Tên sinh viên.

`public int age;` Tuổi.

Phương thức khởi tạo (constructor):

`Student(String id, String name, int age)`: Phương thức khởi tạo 3 trường dữ liệu `id`, `name` và `age` từ 3 tham số.

Hành vi (method):

`String getInfo()`: Phương thức trả về chuỗi thông tin của sinh viên, chuỗi thông tin có dạng `{id=id?, name=name?, age=age?}`.

Lớp StudentManagement (Quản lý sinh viên):

Trường dữ liệu (field):

`public ArrayList<Student> studentList;` Danh sách sinh viên.

Phương thức khởi tạo (constructor):

`StudentManagement()`: Phương thức khởi tạo mặc định, khởi tạo trường dữ liệu `studentList`.

Hành vi (method):

`void showAllStudents()`: Phương thức in ra thông tin của tất cả sinh viên trong danh sách sinh viên `studentList`.

`public void addStudent(Student stu)`: Phương thức thêm đối tượng sinh viên `stu` vào danh sách sinh viên `studentList` nếu mã số sinh viên `id` của sinh viên `stu` hợp lệ. `void updateStudent(Student stu)`: Phương thức cập nhật lại các thông tin của đối tượng sinh viên được tìm thấy trong danh sách sinh viên `studentList` dựa trên mã số sinh viên `id` của `stu`, thông tin cập nhật cho đối tượng sinh viên được tìm thấy trong danh sách sinh viên `studentList` ứng với các thông tin của `stu`.

`void removeStudentById(String id)`: Phương thức xóa đi đối tượng sinh viên được tìm thấy trong danh sách sinh viên `studentList` dựa trên mã số sinh viên `id` ứng với tham số `id`.

## 2.2. QUẢN LÝ SINH VIÊN CHO TRUNG TÂM TIN HỌC. 1

`Student searchStudentByid(String id)`: Phương thức trả về đối tượng sinh viên được tìm thấy trong danh sách sinh viên `studentList` dựa trên mã số sinh viên `id` ứng với tham số `id`.

### TEST CASE.

#### Test case 1.

```
package main;

public class Main {

    public static void main(String[] args) {

        StudentManagement sm = new StudentManagement();

        System.out.println("StudentManagement after initing.");
        sm.showAllStudents();

        System.out.println();
    }
}
```

```

        sm.addStudent(new Student("S01", "A", 21));
        sm.addStudent(new Student("S02", "B", 22));
        sm.addStudent(new Student("S03", "C", 23));
        sm.addStudent(new Student("S03", "D", 19));
        System.out.println("StudentManagement after adding 3 students:");
        sm.showAllStudents();

        System.out.println();

        System.out.println("Searching student with id by S01:");
        System.out.println(sm.searchStudentById("S01").getInfo());

        System.out.println();

        Student student = sm.searchStudentById("S02");
        student.setName("X");
        student.setAge(100);
        sm.updateStudent(student);
        System.out.println("StudentManagement after updating student has id by S02:");
        sm.showAllStudents();

        System.out.println();

        sm.removeStudentById("S02");
        System.out.println("StudentManagement after removing student has id by S02:");
        sm.showAllStudents();
    }
}

StudentManagement after initing:
StudentManagement after adding 3 students:
Student [id=S01, name=A, age=21]
Student [id=S02, name=B, age=22]
Student [id=S03, name=C, age=23]

Searching student with id by S01:
Student [id=S01, name=A, age=21]

StudentManagement after updating student has id by S02:
Student [id=S01, name=A, age=21]
Student [id=S02, name=X, age=100]
Student [id=S03, name=C, age=23]

StudentManagement after removing student has id by S02:
Student [id=S01, name=A, age=21]
Student [id=S03, name=C, age=23]

```

## 2.2. QUẢN LÝ SINH VIÊN CHO TRUNG TÂM TIN HỌC. 2