

Задание1. Студент может зарегистрироваться

```
### POST create student
POST http://localhost:8080/api/students
Content-Type: application/json
```

```
{
  "fullName": "test",
  "email": "mail@mail.ru"
}
```

```
GET http://localhost:8080/api/students
```

Show Request

```
HTTP/1.1 200
```

```
> (Headers) ...Content-Type: application/json...
```

```
{
  "courses": {
    "1": "Java",
    "2": "C#",
    "3": "Python",
    "4": "PHP"
  },
  "students": [
    {
      "id": -174331071285075512,
      "fullName": "test",
      "email": "mail@mail.ru",
      "courses": []
    }
  ]
}
```

Response file saved.

Задание2. Студент может записаться на один или несколько курсов

```
### PUT subscribe student to courses
PUT http://localhost:8080/api/students
Content-Type: application/json
```

```
{
  "id": "-174331071285075512",
  "courses": ["C#", "PHP"]
}
```

```
{
  "courses": {
    "1": "Java",
    "2": "C#",
    "3": "Python",
    "4": "PHP"
  },
  "students": [
    {
      "id": -174331071285075512,
      "fullName": "test",
      "email": "mail@mail.ru",
      "courses": [
        "C#",
        "PHP"
      ]
    }
  ]
}
```

Задание 3. Система автоматически определяет студента по его сессии

```
### POST registartion student without courses
POST http://localhost:8080/api/students/register
Content-Type: application/json
```

```
{
  "fullName": "test2",
  "email": "mail2@mail.ru"
}
```

HTTP/1.1 200

(Headers) ...Content-Type: text/plain;charset=ISO-8859-1...

Data has been registered. Let to choose courses

```
{
  "courses": {
    "1": "Java",
    "2": "C#",
    "3": "Python",
    "4": "PHP"
  },
  "students": []
}
```

```
### POST continue registartion student
POST http://localhost:8080/api/students/complete-registration
Content-Type: application/json
💡
{
  "courses": ["PHP"]
}
```

```
HTTP/1.1 200
> (Headers) ...Content-Type: application/json...

{
  "courses": {
    "1": "Java",
    "2": "C#",
    "3": "Python",
    "4": "PHP"
  },
  "students": [
    {
      "id": -7588489164762460554,
      "fullName": "test2",
      "email": "mail2@mail.ru",
      "courses": [
        "PHP"
      ]
    }
  ]
}
```

Задание 4. Все входные параметры проверяются на корректность (фио, email)

```
### POST create student
POST http://localhost:8080/api/students
Content-Type: application/json

{
  "fullName": "test",
  "email": "@mail.ru"
}
```

```
HTTP/1.1 404
> (Headers) ...Content-Type: application/json...

{
  "message": "Invalid email"
}
```

Задание 5. Тесты должны проверять работу всех операций

```
StudentControllerTest.java x
class StudentControllerTest { new *
}

@Test new *
@DisplayName("Тест получения всех студентов")
void getAllStudentsAndCourses() {
    Map<Integer, String> courses = Map.of(1, "Java");
    List<Student> students = List.of(new Student(id: 1L, fullName: "John", email: "john@mail.ru", courses.values().stream().toList()));
    when(studentService.getAllStudents()).thenReturn(students);
    when(studentService.getAllCourses()).thenReturn(courses);

    StudentAndCoursesResponse response = studentController.getAllStudents();

    assertEquals(courses, response.courses());
    assertEquals(students, response.students());
    verify(studentService).getAllStudents();
}

@Test new *
@DisplayName("Тест регистрации студента")
void registerStudent() {
    StudentRequest request = new StudentRequest(fullName: "John", email: "john@mail.ru");
    when(userSessionService.getFullName()).thenReturn("John");
    when(userSessionService.getEmail()).thenReturn("john@mail.ru");

    ResponseEntity<?> response = studentController.registerStudent(request, Collections.emptyList());

    assertEquals(HttpStatus.OK, response.getStatusCode());
    assertEquals(expected: "Data has been registered. Let to choose courses", response.getBody());
}

@Test new *
@DisplayName("Тест завершения регистрации студента")
void completeRegistration() {
    SubscribeRequest subscribeRequest = new SubscribeRequest(id: null, List.of("PHP"));
    Student student = new Student(id: 1L, fullName: "John", email: "john@mail.ru");
}
```

```
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 2.690 s -- in ru.prj.controllers.StudentControllerTest
[INFO] Results:
[INFO] Tests run: 7, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] Total time: 8.050 s
[INFO] Finished at: 2025-03-13T16:42:12+03:00
[INFO]
D:\JavaEE\Course_JavaEE\homework12 [homeworks/homework13 ≡ +1 ~0 -0 | +4 ~3 -0 !]>
```