============================= ManyToMany =================================

Post Man result OneToOne ki and OneToMany ki chesinatte vuntadi…….!!!

===================================================================================

ManyToMany Mapping main aim :

// Get Address details by User ID

// Get User details by Address ID

================ Unidirectional ManyToMany coding ===================

**// Get Course details by Student ID -- only Parent ----- > Child only Possible**

**@GetMapping("/coursesByStudent/{studentId}")  
public List<Unidirectional\_Course> getCoursesByStudentId(@PathVariable Long studentId) {  
 // Fetch the student by ID  
 Unidirectional\_Student student = unidirectionalStudentRepo.findById(studentId)  
 .orElseThrow(() -> new RuntimeException("Student not found"));  
  
 // Return the list of courses associated with the student  
 return student.getCourses();  
}**

Out put : http://localhost:8080/Course/coursesByStudent/1

[

    {

        "id": 111,

        "title": "java"

    },

    {

        "id": 111,

        "title": "java"

    },

    {

        "id": 222,

        "title": "spring"

    },

================ Bidirectional ManyToMany coding ===================

**// Get Course details by Student ID**

**Bidirectional\_Student\_Controller**

**@GetMapping("/coursesByStudent/{studentId}")  
public List<Bidirectional\_Course> getCoursesByStudentId(@PathVariable Long studentId) {  
 // Fetch the student by ID  
 Bidirectional\_Student student = bidirectionalStudentRepo.findById(studentId)  
 .orElseThrow(() -> new RuntimeException("Student not found"));  
  
 // Return the list of courses associated with the student  
 return student.getCourses();  
}**

**Out Put :** http://localhost:8080/BIStudent/coursesByStudent/1

[

    {

        "id": 222,

        "title": "spring",

        "students": [

            {

                "id": 1,

                "name": "chandu"

            }

        ]

    },

    {

        "id": 111,

        "title": "JAVA",

        "students": [

            {

                "id": 1,

                "name": "chandu"

            }

        ]

    }

]

**Bidirectional\_Course\_Controller**

**@GetMapping("/studentsByCourse/{courseId}")  
public List<Bidirectional\_Student> getStudentsByCourseId(@PathVariable Long courseId) {  
 // Fetch the course by ID  
 Bidirectional\_Course course = bidirectionalCourseRepo.findById(courseId)  
 .orElseThrow(() -> new RuntimeException("Course not found"));  
  
 // Return the list of students associated with the course  
 return course.getStudents();  
}**

**}**

**Out Put :** http://localhost:8080/BICourse/studentsByCourse/111

[

    {

        "id": 1,

        "name": "chandu"

    }

]

===================== \*\*\*\* \*\* \*\*\*\*\* ===============================