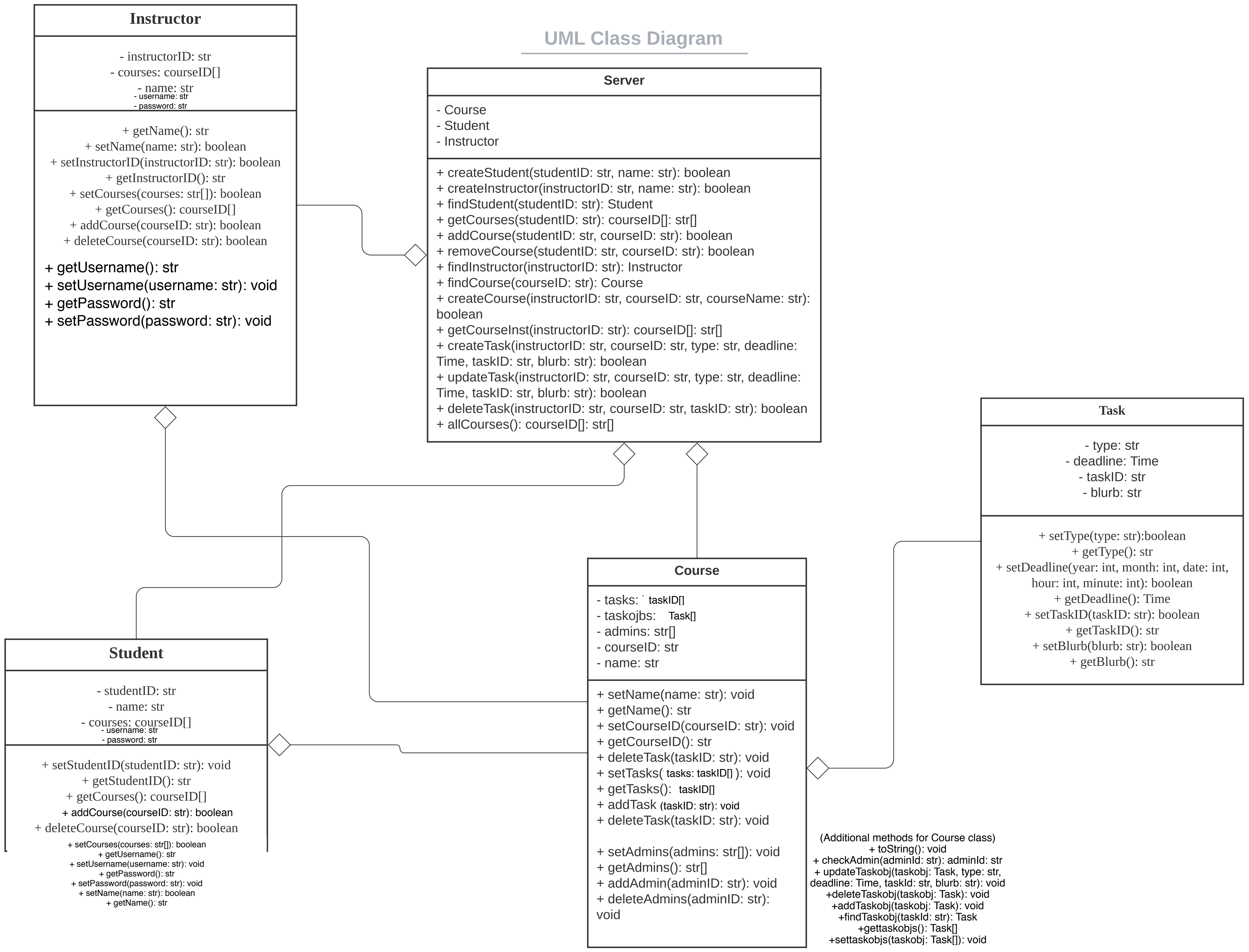
UML Class Diagram


Instructor
- instructorID: str
- courses: courseID[]
- name: str
- username: str
- password: str
+ getName(): str
+ setName(name: str): boolean
+ setInstructorID(instructorID: str): boolean
+ getInstructorID(): str
+ setCourses(courses: str[]): boolean
+ getCourses(): courseID[]
+ addCourse(courseID: str): boolean
+ deleteCourse(courseID: str): boolean
+ getUsername(): str
+ setUsername(username: str): void
+ getPassword(): str
+ setPassword(password: str): void
Server
- Course
- Student
- Instructor
+ createStudent(studentID: str, name: str): boolean
+ createInstructor(instructorID: str, name: str): boolean
+ findStudent(studentID: str): Student
+ getCourses(studentID: str): courseID[]: str[]
+ addCourse(studentID: str, courseID: str): boolean
+ removeCourse(studentID: str, courseID: str): boolean
+ findInstructor(instructorID: str): Instructor
+ findCourse(courseID: str): Course
+ createCourse(instructorID: str, courseID: str, courseName: str): boolean
+ getCourseInst(instructorID: str): courseID[]: str[]
+ createTask(instructorID: str, courseID: str, type: str, deadline: Time, taskID: str, blurb: str): boolean
+ updateTask(instructorID: str, courseID: str, type: str, deadline: Time, taskID: str, blurb: str): boolean
+ deleteTask(instructorID: str, courseID: str, taskID: str): boolean
+ allCourses(): courseID[]: str[]
Task
- type: str
- deadline: Time
- taskID: str
- blurb: str
+ setType(type: str):boolean
+ getType(): str
+ setDeadline(year: int, month: int, date: int, hour: int, minute: int): boolean
+ getDeadline(): Time
+ setTaskID(taskID: str): boolean
+ getTaskID(): str
+ setBlurb(blurb: str): boolean
+ getBlurb(): str
Course
- tasks: taskID[]
- taskojbs: Task[]
- admins: str[]
- courseID: str
- name: str
+ setName(name: str): void
+ getName(): str
+ setCourseID(courseID: str): void
+ getCourseID(): str
+ deleteTask(taskID: str): void
+ setTasks(tasks: taskID[]): void
+ getTasks(): taskID[]
+ addTask (taskID: str): void
+ deleteTask(taskID: str): void
+ setAdmins(admins: str[]): void
+ getAdmins(): str[]
+ addAdmin(adminID: str): void
+ deleteAdmins(adminID: str): void
(Additional methods for Course class)
+ toString(): void
+ checkAdmin(adminId: str): adminId: str
+ updateTaskobj(taskobj: Task, type: str, deadline: Time, taskId: str, blurb: str): void
+deleteTaskobj(taskobj: Task): void
+addTaskobj(taskobj: Task): void
+findTaskobj(taskId: str): Task
+gettaskobjs(): Task[]
+settaskobjs(taskobj: Task[]): void
Student
- studentID: str
- name: str
- courses: courseID[]
- username: str
- password: str
+ setStudentID(studentID: str): void
+ getStudentID(): str
+ getCourses(): courseID[]
+ addCourse(courseID: str): boolean
+ deleteCourse(courseID: str): boolean
+ setCourses(courses: str[]): boolean
+ getUsername(): str
+ setUsername(username: str): void
+ getPassword(): str
+ setPassword(password: str): void
+ setName(name: str): boolean
+ getName(): str