

Student
+ Student(int, String, String, String, String, byte):
- semester: byte - logger: Logger - advisor: Advisor - availableCoursesToDrop: List<Course> - labSections: List<LaboratorySection> - transcript: Transcript - availableCoursesToAdd: List<Course> - notification: Notification - schedule: Course[][] - draft: List<Course> - hasRequest: boolean
+ setTranscript(Transcript): void - checkThePrerequisiteAndCourseThatWasTaken(Course): boolean + getTranscript(): Transcript + getAdvisor(): Advisor + hasCourseOverlap(Course, boolean): boolean + isHasRequest(): boolean - computeAvailableCoursesToDrop(): void + getSemester(): byte - addFacultyTechnicalElective(): void + getSchedule(): Course[][] - maxCoursesReached(): boolean - addTechnicalElective(): void - chooseLabSection(Course): void + setDraft(List<Course>): void + getDraft(): List<Course> - computeAvailableMandatoryCourses(): void - addCourseToDrop(): void + setHasRequest(boolean): void - computeAvailableTEAndFTECourses(String): void - addNonTechnicalElective(): void - computeAvailableNTECourses(): void + setNotification(Notification): void + setPassword(String): void + login(String, String): boolean + setAdvisor(Advisor): void - addMandatoryCourse(): void + removeCourseFromDraft(): void + calculateNumberOfCourses(): int + addCourseToDraft(): void + printMenu(String): void + showRequestStatus(): void + getDepartmentName(): String + sendRequest(): void + setUsername(String): void + getLabSections(): List<LaboratorySection>