

Section

- sectionId: int

- course: Course

- instructor: Instructor

- scheduleList: List<Schedule>

+ getCourse(): Course

+ getInstructor(): Instructor

+ getScheduleList(): List<Schedule>

+ getSectionId(): int

+ setCourse(Course): void

+ setInstructor(Instructor): void

+ setScheduleList(List<Schedule>): Section

+ setSectionId(int): void

- DDSTable: Map<Integer,String>
- course: Course

- addFailPassed(Student, String): void
- assignLetterGrade(int,Grade): void
- assignRandomGrade(): void
- generateRandomGrade(): int
+ getCourse(): Course
- initializeGrade(Student, int, int): void
- round(double): int
+ setCourse(Course):
+ startGrading(): void

- transcriptMap: Map<String, Object>
- student: Student

+ addCourse(Course): void
+ addError(Error): void
+ addSemester(Semester): void
+ createSemester(Semester): Map<String, Object>
+ getStudent(): Student
+ getTranscriptMap(): Map<String, Object>
+ setStudent(Student): void
+ setTranscriptMap(Map<String, Object>): void

Approver

- student: Student

- course: Course

+ approveCourse(Course): boolean

+ capacityChecker(): boolean

+ getCourse(): Course

+ getStudent(): Student

+ isElectiveTaken(): boolean

+ prerequisiteChecker(): boolean

+ setCourse(Course): void

+ setStudent(Student): void

- studentMap: Map<Integer,Student>
- instructors: List<Instructors>

+ createStudent(int, String,String, List<String>, Semester): void
+ getInstructors(): List<Instructor>
- getStudent(int, String, String, List<String>, Semester): Student
+ getStudentMap(): Map<Integer,Student>
+ setInstructors(List<Instructor>): void
+ setStudentMap(Map<Integer, Student>): void

Schedule
- day: String
- startTime: int
- endTime: int
+ compareTo(Schedule): int
+ getDay(): String
+ setDay(String): Schedule

InputReader

- parser: JSONParser

+ getParser(): JSONParser

+ readCourseJson(CourseExpert, InstructorExpert): void

+ readInstructorJson(InstructorExpert): void

+ readPrerequisiteJson(CourseExpert): void

+ readStudentJson(int, StudentExpert, Semester): void

InstructorExpert

- insructorMap: Map<String, Instructor>

+ createInstructor(int, String,String): void

+ findInstructor(String): Instructor

+ getInstructorMao(): Map<String,Instructor>

+ setInstructorMap(Map<String,Instructor>): void

- studentExpert: StudentExpert

+ startWriter(): void

+ writeTranscript(Map<String, Object, int): void

TranscriptReader

- parser: JSONParser

+ startWriter(): void

+ readTranscriptJson(StudentExpert, CourseExpert, InstructorExpert): void