

CourseOfStudy
+ name: string + studentNumber: int + semester: int + cosUsed: bool + visitedCourses: List<Course>

Block
+ timespan: TimeSpan + blockNumber: int + blockCourses: List<Course> + day: DayEnum
+ Block(TimeSpan zeit, int blockNumber, DayEnum day)

Helper
+ <u>ContainsAllItems<T>(this IEnumerable<T> a, this IEnumerable<T> b) : bool</u> + <u>CountStudents() : int</u> + <u>MapCourseOfStudyNameToCourseOfStudy(string courseOfStudyName)</u> + <u>TimeCheck(DateTime a, DayEnum d, Prof p) : bool</u> + <u>SetBlock(OptionalCourse o) : void</u> + <u>PrintCourse(string courseofStudyName) : void</u> + <u>PrintPossibleOptionalCourses(CourseOfStudy courseOfStudy) : void</u> + <u>PrintOptionalCourseData(Block b, Day day, OptionalCourse o) : void</u>

Prof
+ name: string + profUsed: bool + courses: List<string> + busy: List<TimeSpanDay>

Room
+ name: string + seats: int + roomUsed: bool + roomEquipment: List<string>

TimeSpanDay
+ start: DateTime + end: DateTime + day: DayEnum
+ TimeSpanDay(DateTime start, DateTime end)

TimeSpan
+ start: DateTime + end: DateTime
+ TimeSpan(DateTime start, DateTime end)

OptionalCourse
+ name: string + description: string + room: Room + studentNumber: int + prof: Prof + block: Block + timeSpanDay: TimeSpanDay + optionalCourseUsed: bool + equipment: List<string> + shouldPrint: bool

Course
+ name: string + description: string + studentNumber: int + room: Room + prof: Prof + block: Block + courseUsed: bool + equipment: List<string> + coursesOfStudy: List<CourseOfStudy>

<<enumeration>> DayEnum
Montag Dienstag Mittwoch Donnerstag Freitag

Day
+ day: DayEnum + blocks: List<Block>
+ Day(DayEnum day)

CoursesOfStudy Json
"name" "studentNumber" "semester" "visitedCourses" "visitedCourses:name"

Module Json
"name" "description" "equipment" "coursesOfStudy" "coursesOfStudy:name"

Prof Json
"name" "courses" "busy:start" "busy:end" "busy:day"

Room Json
"name" "seats" "roomEquipment"

OptionalCourse Json
"name" "description" "equipment" "timeSpanDay" "timeSpanDay:start" "timeSpanDay:end" "timeSpanDay:day"

TimeTableGenerator
+ <u>week: List<Day></u> + <u>times: TimeSpan[]</u> + <u>courses: List<Course></u> + <u>optionalCourses: List<OptionalCourse></u> + <u>rooms: List<Room></u> + <u>profs: List<Prof></u> + <u>coursesOfStudy: List<CourseOfStudy></u>
+ <u>LoadJsonRooms () : void</u> + <u>LoadJsonProf () : void</u> + <u>LoadJsonCoursesOfStudy () : void</u> + <u>LoadJsonCourses () : void</u> + <u>LoadvisitedCourses () : void</u> + <u>LoadProfsToCourses () : void</u> + <u>LoadOptionalCourses () : void</u> + <u>Initializer () : void</u> + <u>GenerateOptionalTimetable () : void</u> + <u>GenerateTimetable () : void</u> + <u>SortBy () : void</u>