

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

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

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

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

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

ListHelper
<u>+ ContainsAllItems <T></u>
<u>+ ContainsAllItems <T> (IEnumerable <T> a)</u> <u>+ ContainsAllItems <T> (IEnumerable <T> b)</u>

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

OptionalCourse
+ name: string + description: string + block: Block + prof: Prof + room: Room + equipment List<string>

Main TimetableGenerator
+ times: TimeSpan[] + week List<Day> + courses List<Course> + rooms List<Room> + profs List<Prof> + courseOfStudy List<CourseOfStudy>
<u>+ LoadJsonRoom()</u> <u>+ LoadJsonProf()</u> <u>+ LoadJsonCourseOfStudy</u> <u>+ LoadJsonCourse</u> <u>+ LoadJsonvisitedCourses()</u> <u>+ LoadJsonProfsToCourses()</u> <u>+ Initializer()</u> <u>+ GenerateTimetable()</u>

TimeHelper
<u>+ TimeCheck (DateTime a)</u> <u>+ TimeCheck (Days d)</u> <u>+ TimeCheck (Prof p)</u>

TimeSpanDay
+ start: DateTime + end: DateTime + day: Days
+ TimeSpanDay (DateTime start) + TimeSpanDay (DateTim end) + TimeSpanDay (Days day)

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

<<enumeration>> Days
Montag Dienstag Mittwoch Donnerstag Freitag

<<enumeration>> Equipment
Computer Kreidetafel Whiteboard Ansteigend Laborausstattung

<<enumeration>> BlockNumber
one = 1 two = 2 three = 3 four = 4 five = 5 six = 6

OptionalCourseSetter
<u>+ SetBlock(List<Day> week</u> <u>+ TimeSpan[] times</u> <u>+ OptionalCourse o</u>