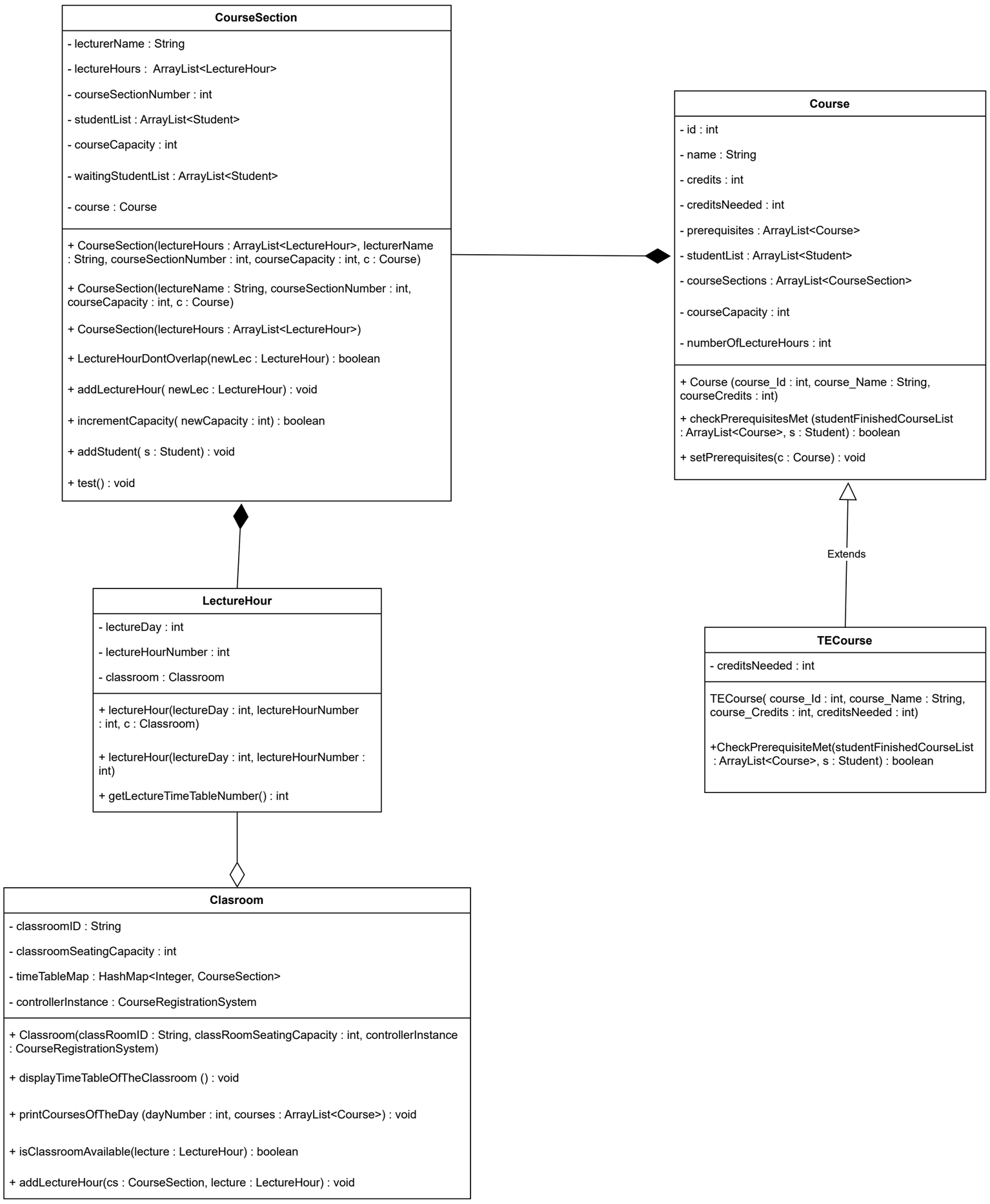
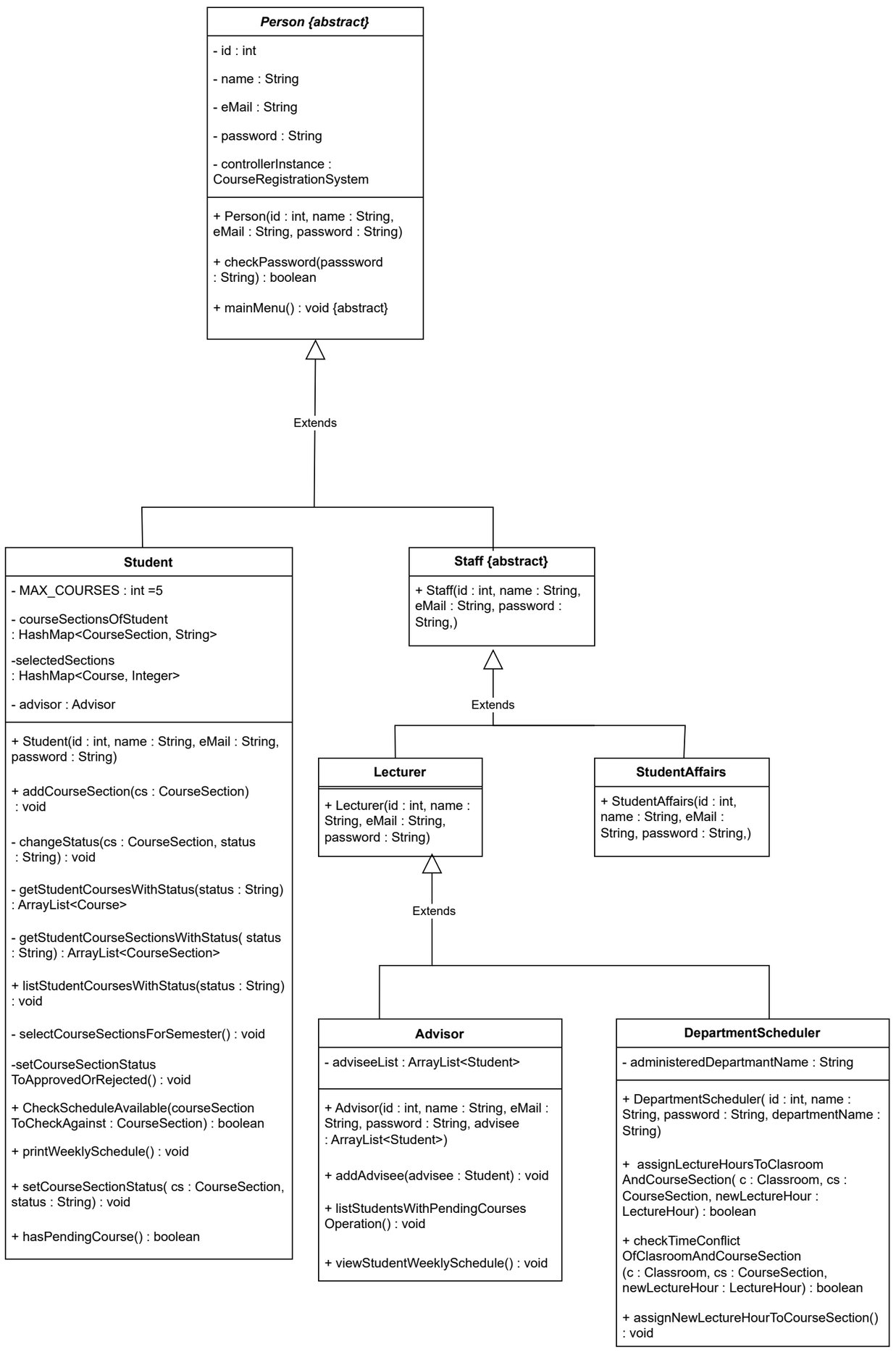
NOTE: FOR SIMPLIFICATION GETTER AND SETTER METHODS ARE OMITTED.




Person {abstract}
- id : int
- name : String
- eMail : String
- password : String
- controllerInstance : CourseRegistrationSystem
+ Person(id : int, name : String, eMail : String, password : String)
+ checkPassword(passsword : String) : boolean
+ mainMenu() : void {abstract}
Extends
Student
- MAX_COURSES : int =5
- courseSectionsOfStudent : HashMap<CourseSection, String>
-selectedSections : HashMap<Course, Integer>
- advisor : Advisor
+ Student(id : int, name : String, eMail : String, password : String)
+ addCourseSection(cs : CourseSection) : void
- changeStatus(cs : CourseSection, status : String) : void
- getStudentCoursesWithStatus(status : String) : ArrayList<Course>
- getStudentCourseSectionsWithStatus(status : String) : ArrayList<CourseSection>
+ listStudentCoursesWithStatus(status : String) : void
- selectCourseSectionsForSemester() : void
-setCourseSectionStatus ToApprovedOrRejected() : void
+ CheckScheduleAvailable(courseSection ToCheckAgainst : CourseSection) : boolean
+ printWeeklySchedule() : void
+ setCourseSectionStatus(cs : CourseSection, status : String) : void
+ hasPendingCourse() : boolean
Staff {abstract}
+ Staff(id : int, name : String, eMail : String, password : String,)
Extends
Lecturer
+ Lecturer(id : int, name : String, eMail : String, password : String)
StudentAffairs
+ StudentAffairs(id : int, name : String, eMail : String, password : String,)
Extends
Advisor
- adviseeList : ArrayList<Student>
+ Advisor(id : int, name : String, eMail : String, password : String, advisee : ArrayList<Student>)
+ addAdvisee(advisee : Student) : void
+ listStudentsWithPendingCourses Operation() : void
+ viewStudentWeeklySchedule() : void
DepartmentScheduler
- administeredDepartmantName : String
+ DepartmentScheduler(id : int, name : String, password : String, departmentName : String)
+ assignLectureHoursToClasroom AndCourseSection(c : Classroom, cs : CourseSection, newLectureHour : LectureHour) : boolean
+ checkTimeConflict OfClasroomAndCourseSection (c : Classroom, cs : CourseSection, newLectureHour : LectureHour) : boolean
+ assignNewLectureHourToCourseSection() : void

CourseRegistrationSystem

- inputScanner : Scanner

- instance : CourseRegistrationSystem=null

- allDepartmentCoursesList : ArrayList<Course>

- allClassrooms : ArrayList<Classroom>

- CourseRegistrationSystem()

+ main(args : String[]) : void

+ printAllDepartmentCourses (studentCoursesMap : HashMap<CourseSection,String>) : void

+ printAllDepartmentCourses() : void

+ printAllClassroomsWithCapacities() : void

+ reviewClassroomSchedule() : void

+ reviewAllClassroomSchedules() : void

+ selectCoursesFromAllDepartmentCourses (studentName : String, allStudentCourses : HashMap<Course,String>) : Course

+ updatethismethodnametosomething sensibleselectCoursesFromAllDepartmentCourses() : Course

+ selectCoursesSectionFromCourse (c : Course) : CourseSection

+ selectClassroomFromAllClassrooms() : Classroom

+ menuOperation(prefixText : ArrayList<String> ,numberedOptionsText : ArrayList<String>, suffixText : ArrayList<String>) : int

+ menuOperation(numberedOptionsText : ArrayList<String>,suffixText : ArrayList<String>) : void

+ menuOperation(prefixText : ArrayList<String>) : void

+ menuOperation(prefixText : ArrayList<String>) : void

+ getNextLine() : String

+ getNextInt() : int

+ printAllGivenCoursesWithTheirCapacity() : void

FileOperation

- courseMap : HashMap<Integer,Course>

-advisorMap : HashMap<Integer,Advisor>

-studentMap : HashMap<Integer,Student>
-people : ArrayList<person>

- people : ArrayList<Person>

- classrooms : ArrayList<Classroom>

-controllerInstance :CourseRegistrationSystem

+FileOperationClass(c : CourseRegistrationSystem)

+readJson(dirpath : String): void

- mapCoursesandAdvisorToStudent (dirpath:String, student : MapHashMap<Integer,Student>, courseMap: HasMap<Integer,Course> , advisorMap :HasMap<Integer,Advisor>) : void

- readDepartmentSchucler(dirpath : String) : void

- mapPrerequisiteToCourse(courseDirpath : String, courseMap : HashMap<Integer,Course) : void

- readClassroomJSON(dirPath : String) : void

- readDepartmentScheduler(dirPath : String) : void

- readCourseJSON(dirpath : String) void

- readAdvisorJSON(dirpath : String) void

- readStudentJSON(dirpath : String) void

- takeJsonFiles(dirpath : String) : File[]

+ updateJSON() : void

- updateStudentJSON() : void

- updateAdvisorJSON() : void