Person

Room

Professor

Staff

Student

Section

Course

Exam

composition: tail is apart of head,
where head owns its own tail
in memory

Department

Controller

Window

Window

Window

Window

Window

Professor — Section: 0..1, *, 1
Student — Section: *, *, 1
Room — Section: 1, 1
Room — Exam: 1, 1
Section — Course: *, 1
Course — Exam: 1, 1
Staff — Department: *, 1
Staff — Course: *

Person

Person

name: String

email: String

id: String

+ CMP: Comparator<Person>

+ Person(List<String>)

+ getName(): String

+ getEmail(): String

+ getID(): String

+ compareTo(Person): int

Person is an
abstract class

Student

Student

- courses: List<String>

+ Student(List<String>)

+ getCourses(): List<String>

courses: List<String> should
have the CAT#s of courses
the student is taking

Staff

Staff

- role: String

+ Staff(List<String>)

+ getRole(): String

Exam

Exam

- date: Date

- room: Room

+ Exam(Date, Room)

+ getDate(): Date

+ getRoom(): Room

Room

Room

- roomNum: int

+ Room(int)

+ getRoomNum(): int

Section

Section

- letter: String

- prof: Professor

- catNum: String

+ Section(String, Professor, String)

+ getLetter(): String

+ getProfessor(): Professor

+ getCatNum(): String

+ na

+ de

+ se

+ ex

+ Co

+ ge

+ ge

+ ge

+ ge

Department
+ field: type
+ method(type): type

Professor
- courses: List<String>
+ Professor(List<String>) + getCourses(): List<String>

courses: List<String> should
have the CAT#s of courses
the professor teaches

Course
me: String sc: String ction: Section am: Exam
ourse(String, Section, String, Exam) tName(): String tDesc(): String tSection(): Section tExam(): Exam