## main ~ E: EnrollmentManager ~ users: List<User> ~ s: Scanner + logIn(): User $\Diamond$ ~ createUser(c: int): User + selectedCourse(p: Professor, course: Course) ~ menu(p: Professor) ~ menu(a: Admin) ~ menu(stu: Student) ~ listProfessors() ~ createPreexistingData() User(Abstract) ~ enrollment: EnrollmentManager ~ username: String ~ password: String ~ userType: String ~ name: String + User(name: String, username: String, password: String) ~ loginVerification(password: String): boolean ~ viewCourses() + logInVerification(username: String, password: String): boolean + viewCourses() + getUsername(): String + setUsername(username: String) + getPassword(): String + setPassword(password: String) + setUserType(userType: String) + getUserType(): String + setName(name: String) + getName(): String

Student	Professor	Admin	
+ coursesEnrolled: List <course></course>	+ coursesTaught: List <course></course>		Pair
+ Student(name: String, username: String, password: String) ~ viewCourses() + unEnroll(c: Course): int ~ enroll(c: Course)	~ viewCourses() + viewStudents(C: Course) + changeGrade(C: Course, name: String, grade: int) + getTaughtByName(name: String) + getTaughtByCode(code: String) + Professor(name: String, username: String, password: String) + leaveCourse(c: Course): int	+ Admin(name: String, username: String, password: String) + addCourse(courseName: String, description: String, code: String, professor: Professor) + deleteCourseByCode(code: String) + deleteCourseByName(name: String) + viewCourses()	+ student: Student + grade: int  + Pair(s: Student, g: int) + equals(o: Object): boolean + getStudent(): Student  Course  - courseName: String - courseDescription: String - courseCode: String
	EnrollmentManager  + eM: EnrollmentManager  - CourseOffering: List <course>  + addCourse(c: Course) + removeCourse(c: Course) + getManager(): EnrollmentManager + printCourseOptions() + getCourseByName(name: String) + getCourseByCode(code: String)</course>		- studentList: List <pair> - Course(cN:String, cD: String, cC: String, p: Professor) + clearClass() + printCourse() + addStudent(s: Student): int + getGrade(s: Student): int + setGrade(s: Student, g: int) + getCourseCode(): String + getCourseName(): String + setCourseName(courseName: String) + setCourseDescription(): String + setCourseDescription(): String + setCourseDescription(courseDescription: String) + getProfessor(): User + setProfessor(p: Professor) + getStudents(): List<pair> + getStudent(name: String): Student</pair></pair>