As part of the GRADS project, you will have to read in student records, courses, and users from a set of databases (functionally, these are actually flat files containing student records in the JSON format). We have included samples in this ZIP, and these tables will listing the fields that may appear, and examples of the type of data that might appear in the those fields.

You will need to implement Java classes to contain and represent this data.

Student Record Mapping:

Key	Value Type
student	Student class, contains fields: <table><tr><td>**Key**</td><td>**Value Type**</td></tr><tr><td>id</td><td>string (ex: ggay)</td></tr><tr><td>firstName</td><td>string (ex: Greg)</td></tr><tr><td>lastName</td><td>string (ex: Gay)</td></tr></table>
department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE)
termBegan	Term class, contains fields: <table><tr><td>**Key**</td><td>**Value Type**</td></tr><tr><td>semester</td><td>One of: {FALL, SPRING, SUMMER}</td></tr><tr><td>year</td><td>four digit number (ex: 2010)</td></tr></table>
degreeSought	Degree class, containing fields:

	Key	Value Type
	name	One of: {BS,MS,MENG,MSE,PHD}
	graduation	Expected graduation date. Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

certificateSought	Optional, special instance of degree class, containing fields:

Key	Value Type
name	INFORMATION_ASSURANCE
graduation	Expected graduation date. Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}

		year	four digit number (ex: 2010)

previousDegrees	List, containing 0 or more Degree instances, each with the following fields:

Key	Value Type
name	One of: {BS,MS,MENG,MSE,PHD}
graduation	Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

advisors	List, containing 0 or more professors, each with the following fields:

Key	Value Type
department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department

	firstName	string (ex: Greg)
	lastName	string (ex: Gay)

committee	List, containing 0 or more professors, each with the following fields:	
	Key	**Value Type**
	department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department
	firstName	string (ex: Greg)
	lastName	string (ex: Gay)

coursesTaken	List, containing 0 or more "taken courses", each with the following fields:	
	Field	**Value Type**
	course	Instance of a course, containing the following fields:

	Key	**Value Type**
	name	string (ex: Machine Learning)
	id	four letters, followed by three numbers (ex: csce740)
	numCredits	String, represents a number

	term	Instance of Term class, containing:	
		Key	**Value Type**
		semester	One of: {FALL, SPRING, SUMMER}
		year	four digit number (ex: 2010)
	grade	One of: {A, B, C, D, F, P, _ (no grade - for in progress courses)}	

milestonesSet	List containing 0 or more completed milestones. Each completed milestone has the following fields:	
	Key	**Value Type**
	milestone	string, all uppercase with no spaces (ex: DEFENSE_PASSED)
	term	Instance of Term class, containing:

Within the "term" cell:

Key	**Value Type**
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

notes	List of 0 or more strings. Notes left by a GPC on a student's record.

You will also need to produce progress summaries in a similar format. An example has been included, but here is a guide to the fields that should appear.

Progress Summary Mapping:

Key	Value Type
student	Student class, contains fields: <table><tr><td>**Key**</td><td>**Value Type**</td></tr><tr><td>id</td><td>string of letters and numbers (ex: gayxx067)</td></tr><tr><td>firstName</td><td>string (ex: Greg)</td></tr><tr><td>lastName</td><td>string (ex: Gay)</td></tr></table>
department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE)
termBegan	Term class, contains fields: <table><tr><td>**Key**</td><td>**Value Type**</td></tr><tr><td>semester</td><td>One of: {FALL, SPRING, SUMMER}</td></tr><tr><td>year</td><td>four digit number (ex: 2010)</td></tr></table>
degreeSought	Degree class, containing fields: <table><tr><td>**Key**</td><td>**Value Type**</td></tr></table>

	name	One of: {BS,MS,MENG,MSE,PHD}
	expectedGraduation	Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

certificateSought	Optional, special instance of degree class, containing fields:

Key	Value Type
name	INFORMATION_ASSURANCE
expectedGraduation	Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

advisors	List, containing 0 or more professors, each with the following fields:	
	Key	**Value Type**
	department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department
	firstName	string (ex: Greg)
	lastName	string (ex: Gay)

committee	List, containing 0 or more professors, each with the following fields:	
	Key	**Value Type**
	department	formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department
	firstName	string (ex: Greg)
	lastName	string (ex: Gay)

requirementCheckResults	List of graduation requirements checks, one per degree requirement, with the following fields:	
	Key	**Value Type**
	name	String, all uppercase, no spaces (ex: BREADTH_REQUIREMENT)

passed	Boolean
details	Details on the requirement, contains the following fields:

Key	**Value Type**
gpa	Optional, used for GPA-based requirements, number from 0.0-4.0
courses	Optional, used for course-based requirements, list containing 0 or more taken courses instances. Each taken course has the following fields:

Field	**Value Type**
course	Instance of Course, containing:

Key	**Value Type**
name	string (ex: Machine Learning)
id	four letters, followed by three numbers (ex: csce740)
numCredits	String, represents a number

Field	**Value Type**
term	Instance of Term class, containing:

Key	Value Type
semester	One of: {FALL, SPRING, SUMMER}
year	four digit number (ex: 2010)

grade	One of: {A, B, C, D, F, P _ (course in progress)}

milestones	Optional, list containing 0 or more completed milestones. Each completed milestone has the following fields:

Key	Value Type
milestone	string, all uppercase with no spaces (ex: DEFENSE_PASSED)
term	Instance of Term class, containing:

Key	Value Type
semester	One of:

				{FALL, SPRING, SUMMER}
			year	four digit number (ex: 2010)
		notes	Optional, list containing 0 or more strings. Use this for describing why a graduation rule was not passed.	

Two additional data sets also must be loaded into your system. One contains a full data set of the courses that the department offers currently, along with the number of credit hours:

Courses Mappings:

Key	Value Type
name	String, (example: Operating Systems)
id	String, four lowercase letters followed by three numbers (example: csce740)
numCredits	String, either a single number or a range (ex: 3, ex: 1-3)

Note - this is only a list of CSCE courses. Courses from other departments do not need to be validated (you can assume out-of-department courses are accurate).

The other is a simple permissions data set:

Permissions Mappings:

Key	Value Type
id	string of letters and numbers (ex: gayxx067)
firstName	string (ex: Greg)
lastName	string (ex: Gay)
role	String, one of {STUDENT, GRADUATE_PROGRAM_COORDINATOR}
department	String, all uppercase, with underscores instead of spaces (example: COMPUTER_SCIENCE)