As part of the GRADS project, you will have to read in student records from a “records database” (actually a flat file containing student records in the JSON format). We have included a sample file in this ZIP, but here is a table listing the fields that may appear, and examples of the type of data that might appear in the that field.

Student Record Mapping:

<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>student</td><td>Student class, contains fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>id</td><td>string of letters and numbers (ex: gayxx067)</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>degreeSought</td><td>One of: {MS-A,MS-B,MS-C,PHD}, must be selection admitted by Enum class Degree</td></tr>
<tr><td>termBegan</td><td>Term class, contains fields:
<table>
<tr><th>Key</th><th>Value Type</th></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>
</td></tr>
<tr><td>advisors</td><td>List, containing 0 or more instances of Professor, each with the following fields:</td></tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td>
</tr>
<tr>
<td>committee</td>
<td>List, containing 0 or more Professor instances, each with the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td>
</tr>
<tr>
<td>coursesTaken</td>
<td>List, containing 0 or more instances of CourseTaken with the following fields:
<table>
<tr><th>Field</th><th>Value Type</th></tr>
<tr><td>course</td><td>Instance of class Course, containing the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
</table>
</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr>
<td></td>
<td>
<table>
<tr><td>name</td><td>string (ex: Machine Learning)</td></tr>
<tr><td>id</td><td>four letters, followed by four numbers (ex: csci5801)</td></tr>
<tr><td>numCredits</td><td>String, represents a number</td></tr>
</table>
</td>
</tr>
<tr>
<td>term</td>
<td>Instance of Term class, containing:
<table>
<tr><th>Key</th><th>Value Type</th></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>
</td>
</tr>
<tr>
<td>grade</td>
<td>One of: {A, B, C, D, F, S, N, _ (no grade - for in progress courses)}, allowed grades defined i Grade enum class.</td>
</tr>
</table>
</td>
</tr>
<tr>
<td>milestonesSet</td>
<td>List containing 0 or more MilestoneSet instances. Each MilestoneSet instance has the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>milestone</td><td>string, all uppercase with no spaces (ex: DEFENSE_PASSED), must be admitted by Milestone enum</td></tr>
<tr><td>term</td><td>Instance of Term class, containing:</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr>
<td></td>
<td>
<table>
<tr><th>Key</th><th>Value Type</th></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>
</td>
</tr>
</table>
</td>
</tr>
<tr>
<td>notes</td>
<td>List of 0 or more strings</td>
</tr>
</table>

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:

<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>student</td><td>Student class, contains fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>id</td><td>string of letters and numbers (ex: gayxx067)</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>degreeSought</td><td>One of: {MS-A,MS-B,MS-C,PHD}, must be selection admitted by Enum class Degree</td></tr>
<tr><td>termBegan</td><td>Term class, contains fields:
<table>
<tr><th>Key</th><th>Value Type</th></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>
</td></tr>
<tr><td>advisors</td><td>List, containing 0 or more instances of Professor, each with the following fields:</td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td></tr>
<tr><td>committee</td><td>
List, containing 0 or more Professor instances, each with the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>department</td><td>formatted text, all uppercase, no spaces (ex: COMPUTER_SCIENCE), must be selection admitted by Enum class Department</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td></tr>
<tr><td>requirementCheckResults</td><td>
List of RequirementCheckResult instances, one per degree requirement, with the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>name</td><td>String, all uppercase, no spaces (ex: BREADTH_REQUIREMENT)</td></tr>
<tr><td>passed</td><td>Boolean</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr>
<td>details</td>
<td>Optional, used for pass verdicts, instance of CheckResultDetails class contains the following fields:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>gpa</td><td>Optional, used for GPA-based requirements, number from 0.0-4.0</td></tr>
<tr><td>courses</td><td>Optional, used for course-based requirements, list containing 0 or more CourseTaken instances. Each CourseTaken has the following fields:
<table>
<tr><th>Field</th><th>Value Type</th></tr>
<tr><td>course</td><td>Instance of Course, containing:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>name</td><td>string (ex: Machine Learning)</td></tr>
<tr><td>id</td><td>four letters, followed by four numbers (ex: csci5801)</td></tr>
<tr><td>numCredits</td><td>String, represents a number</td></tr>
<tr><td>courseArea</td><td>String, all uppercase, no spaces (one of: THEORY_ALGORITHMS,</td></tr>
</table>
</td></tr>
</table>
</td></tr>
</table>
</td>
</tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td></td><td>
<table>
<tr><td></td><td>
<table>
<tr><td></td><td>
<table>
<tr><td></td><td>ARCHITECTURE_SYSTEMS_SOFTWARE, APPLICATIONS), must match item in CourseTaken enum</td></tr>
</table>
</td></tr>
<tr><td>term</td><td>Instance of Term class, containing:
<table>
<tr><th>Key</th><th>Value Type</th></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>
</td></tr>
<tr><td>grade</td><td>One of: {A, B, C, D, F, S, N, _ (course in progress)}, member of Grade enum</td></tr>
</table>
</td></tr>
<tr><td>other</td><td>Optional, list containing 0 or more strings</td></tr>
</table>
</td></tr>
<tr><td>errorMsgs</td><td>Optional, used for fail verdicts, ;ist containing 0 or more strings</td></tr>
</table>
</td></tr>
<tr><td>notes</td><td>0 or more strings</td></tr>
</table>

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 and the course area (for breadth requirements):

Courses Mappings:

Key	Value Type
name	String, (example: Operating Systems)
id	String, four lowercase letters followed by four numbers (example: csci5103)
numCredits	String, either a single number or a range (ex: 3, ex: 1-3)
area	formatted string, all uppercase with underscores instead of spaces, one of: {ARCHITECTURE_SYSTEMS_SOFTWARE, APPLICATIONS, THEORY_ALGORITHMS}

The other is a simple permissions data set:

Permissions Mappings:

<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>user</td><td>Struct, containing:
<table>
<tr><th>Key</th><th>Value Type</th></tr>
<tr><td>id</td><td>string of letters and numbers (ex: gayxx067)</td></tr>
<tr><td>firstName</td><td>string (ex: Luan)</td></tr>
<tr><td>lastName</td><td>string (ex: Nguyen)</td></tr>
</table>
</td></tr>
</table>

role	String, one of {STUDENT, GRADUATE_PROGRAM_COORDINATOR}
department	String, all uppercase, with underscores instead of spaces (example: COMPUTER_SCIENCE)