

Genius Enterprise

Data Migrations

Contents

Summary	2
File Imports	2
Import Templates	3
Import File Rules.....	3
Upload an Import File.....	4
General Import Rules	4
User Import Rules.....	5
Learner Import Rules.....	5
Course Import Rules	5
Section Import Rules	6
Enrollment Import Rules	6
Application Programming Interface (API).....	7
General API Rules	7
Bulk Create	7
Types	8
Courses.....	9
Create	9
Types	9
Sections	10
Create	10
Types	10
Enrollments	11
Bulk Create	11
Types	11

Summary

The primary entities that are migrated to Genius are:

- Users
- Courses
- Sections: requires Courses
- Enrollments: requires Users, Courses, and Sections

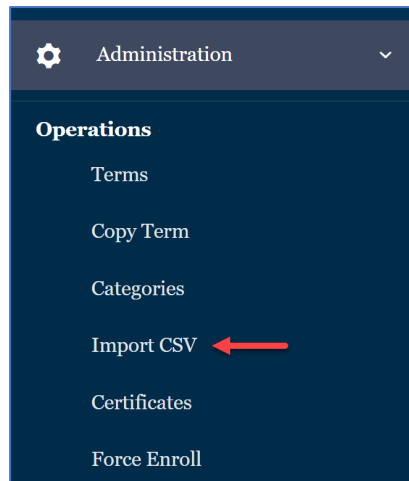
Genius has 2 standard data migration approaches:

1. [File Imports](#)
2. [Application Programming Interface \(API\)](#)

File Imports

Genius supports Comma Separated Value (CSV) file Imports.

Interface Location: Genius Application Portal -> Administration -> Operations -> Import CSV



Import Templates

Genius provides starter csv templates with headers and sample data

You can generate templates on the Import CSV page.

1. Select the Import Type you want to generate a template for
2. Click the Get Sample CSV button
3. The file will download to your browsers default download location (Most of the times it's the Downloads folder)



Import File Rules

- The Header is required and must be the first row
- Header names are not case sensitive
 - INDEX = index = InDeX
- We perform data lookups in the CSV by header name, not column position
- Below are the standard accepted bool type values (not case sensitive). Any value not in the list below is stored as a false.
 - 'y'
 - 'yes'
 - 1
 - 'true'

Upload an Import File

On the Import CSV page

1. Select the Import Type that matches the import CSV file you want to upload
2. Click the Upload button



The screenshot shows the 'Import CSV' interface. At the top, it says 'Import CSV'. Below that is a 'Type' dropdown menu. The dropdown is open, showing a list of options: 'Learners', 'Courses', 'Sections', 'Instructors', 'Enrollments', 'Meeting Times', and 'External Courses'. A red bracket on the right side of the dropdown menu groups the options from 'Courses' down to 'External Courses' and is labeled 'Supported Imports'. The 'Learners' option is currently selected. Below the dropdown menu are three buttons: 'Preview', 'Get Sample CSV', and 'Upload'. The 'Upload' button is highlighted with a red border and a red arrow points down to it from the 'Supported Imports' label.

General Import Rules

- Non-Required columns can be omitted from the file if not needed
- For Updates: no value in a cell will skip processing the cell data

User Import Rules

- See the [User Api documentation](#) for type max length information
- Required Fields
 - userName
 - firstName
 - lastName
 - email
 - Roles
 - Must have at least one
 - STUDENT
 - TEACHER
- Password: if not provided – one will be created for you
- Status (ACTIVE or ARCHIVED)
 - Uppercase the data
 - If left blank, the default is ACTIVE
- The column for affiliations can be one of 3 names depending on intent
 - AffiliationIndex: single index
 - AffiliationIndexes: multiple indexes
 - comma delimited list, enclosed in quotes, no spaces
 - ex. "77,48,23"
 - Affiliation: single affiliation name

Learner Import Rules

- See the [User Api documentation](#) for type max length information
- Required Fields
 - userName
 - firstName
 - lastName
 - email
- Password: if not provided – one will be created for you
- Status (ACTIVE or ARCHIVED)
 - Uppercase the data
 - If left blank, the default is ACTIVE

Course Import Rules

- See the [Course Api documentation](#) for type max length information
- Required Fields
 - courseCode
 - Name
 - Category
 - Status (ACTIVE or ARCHIVED)
 - Uppercase the data

Section Import Rules

- See the [Section Api documentation](#) for type max length information
- Required Fields
 - CourseCode
 - Name
 - Status
 - Lms
- LMS: the name of the LMS
- The column for instructors can be one of 2 names depending on intent
 - InstructorUsername: single name
 - InstructorUsernames: multiple indexes
 - comma delimited list, enclosed in quotes, no spaces
 - ex. "name1,name2"
- Affiliation: single affiliation name
 - If not provided, the system default is used
- Credits: system default is used if one isn't provided
- ExternalLink: if the value provided doesn't start with "http", then "http://" is prefixed to the provided value
- Competencies: comma delimited list, enclosed in quotes, no spaces
- Duration: If not provided, the system default is used

Enrollment Import Rules

- See the [Enrollment Api documentation](#) for type max length information
- Required Fields
 - UserName
 - Section (see rules below)
- SectionIndex: The column can have one of 3 value types depending on intent. SectionIndex is the first value checked; non-sectionIndex values will impact import times
 - SectionIndex: single section index
 - SectionCode: single section code
 - Name: single section name

Application Programming Interface (API)

Genius provides a REST API to our clients so they can perform many of the Genius operations programmatically, including adding and updating:

- Users (students and teachers)
- Courses
- Sections
- Enrollments

Each of the operations above have single item and bulk endpoints.
The typical workflow

General API Rules

- Non-Required fields can be omitted from the api payload
- For Updates: null value in a field will skip processing the cell data
- Bool Values
 - Yes/True values can be any of the following: “y”, “yes”, “true”, or 1
 - No/False values are any non-yes value

Users

Bulk Create

POST: `http://{hostname}:{port}/api/v1/users/bulk`

Rules:

- Required Fields
 - userName
 - firstName
 - lastName
 - email
 - Roles
 - Must have at least one
 - STUDENT
 - TEACHER
- Password: if not provided – one will be created for you
- memberOf
 - Can be AffiliationName nvarchar(256) or AffiliationIndex int

Types

- username: max 64
- firstName: max 64
- lastName: max 64
- email: max 64
- password: max 64
- roles: array of strings, each string is max 64
- phone: max 64
- isActive: Yes/Ture values can be any of the following: “y”, “yes”, “true”, or 1
- localId: max 64
- notes: max 4000
- middleName: max 64
- instructorId: int
- memberOf (Affiliations): array of AffiliationName nvarchar(256) or AffiliationIndex int

Payload Example:

```
[
  {
    "userName": " uname1",
    "firstName": "fname1",
    "lastName": "lname1",
    "email": " uname1@geniusclient.com",
    "password": null,
    "roles": [
      "role1",
      "role2"
    ],
    "phone": null,
    "isActive": false,
    "localId": null,
    "notes": null,
    "middleName": null,
    "instructorId": null,
    "memberOf": [
      {
        "affiliationId": "1",
        "affiliationName": "Affiliation1"
      },
      {
        "affiliationId": "2",
        "affiliationName": "Affiliation2"
      }
    ]
  }
]
```


Courses

Create

POST: <http://{hostname}:{port}/api/v1/courses>

Rules:

- Required Fields
 - Name
 - externalCourseCode
 - DepartmentId
 - Status
- Status:
 - Must be one of
 - ACTIVE
 - ARCHIVED

Types

- name: max 256
- externalCourseCode: max 64
- status: {ACTIVE|ARCHIVED}
- sections.sectionId: int
- sections.name: max 256
- affiliationId: int
- affiliationName: max 256
- lmsId: max 4000
- departmentId: int

Payload Example:

```
{
  "name": "courseName1", /*REQUIRED*/
  "externalCourseCode": "extCode1", /*REQUIRED*/
  "status": "ACTIVE", /*REQUIRED*/
  "sectionId": [
    {
      "sectionId": "123",
      "name": "sec1"
    }
  ],
  "affiliationId": null,
  "affiliationName": null,
  "lmsId": null,
  "departmentId": 1 /*REQUIRED*/
}
```

Sections

Create

POST: http://{hostname}:{port}/api/v1/courses/{courseId}/sections

Rules:

- Required Fields
 - Name
 - Status
 - lmsId
 - externalSectionCode
 - termId
- Status:
 - Must be one of
 - ACTIVE
 - ARCHIVED
- Instructors: If one isn't provided, the system defined default instructor/teacher is assigned

Types

- courseId: int
- name: max 256
- externalCourseCode: max 64
- status: {ACTIVE|ARCHIVED}
- affiliationId: int
- affiliationName: max 256
- lmsId: max 4000
- termId: int
- days: int
- credits: int
- instructor.instructorId: int

Payload Example:

```
{
  "name": "sectionName1", /*REQUIRED*/
  "status": "ACTIVE", /*REQUIRED*/
  "affiliationId": null,
  "affiliationName": null,
  "lmsId": "Id123", /*REQUIRED*/
  "externalSectionCode": "extSec123", /*REQUIRED*/
  "termId": 123, /*REQUIRED*/
  "days": null,
  "credits": null,
  "instructors": [
    {
      "userId": 123
    }
  ]
}
```

Enrollments

Bulk Create

PUT: <http://{hostname}:{port}/api/v1/enrollments/bulkCreateOrUpdate>

Rules:

- Required Fields
 - userId
 - sectionId
 - startDate
 - endDate
- Status:
 - If not provided, ACTIVE is assigned
 - If provided, used one of the following
 - ACTIVE
 - ARCHIVED

Types

- userId: int
- sectionId: int
- startDate: date
- endDate: date
- status: {ACTIVE|ARCHIVED}



Payload Example:

```
[  
  {  
    "userId": 153, /*REQUIRED*/  
    "sectionId": 31, /*REQUIRED*/  
    "startDate": "2022-08-16", /*REQUIRED*/  
    "endDate": "2022-08-17", /*REQUIRED*/  
    "status": "ACTIVE"  
  }  
]
```