

Canvas LMS - REST API and Extensions Documentation

- ☐ Expand all
- ☐ Basics
- [GraphQL API Change Log](#)[SIS IDs](#)[Pagination](#)[Throttling](#)[Compound Documents](#)[File Uploads](#)[API Endpoint Attributes](#)[Masquerading](#)
- ☐ OAuth2
- [OAuth2 Overview](#)[OAuth2 Endpoints](#)[Developer Keys](#)
- ☒ Resources
- [All Resources](#)[Reference](#)[Access Tokens](#)[Account Calendars](#)[Account Domain Lookups](#)[Account Notifications](#)[Account Reports](#)[Accounts](#)[Accounts \(LTI\)](#)[Admins](#)[Analytics](#)[Announcement](#)[External Feeds](#)[Announcements](#)[API Token Scopes](#)[Appointment Groups](#)[Assignment Extensions](#)[Assignment Groups](#)[Assignments](#)[Authentication Providers](#)[Authentications](#)[Log Blackout Dates](#)[Block Editor](#)[Template](#)[Blueprint](#)[Courses](#)[Bookmarks](#)[Brand Configs](#)[Calendar](#)[Events](#)[Collaborations](#)[CommMessages](#)[Communication Channels](#)[Conferences](#)[Content Exports](#)[Content Migrations](#)[Content Security Policy](#)[Settings](#)[Content Shares](#)[Conversations](#)[Course Audit](#)[log](#)[Course Pace](#)[Course Quiz Extensions](#)[Courses](#)[Custom Gradebook](#)[Columns](#)[Discussion Topics](#)[Enrollment Terms](#)[Enrollments](#)[ePortfolios](#)[ePub Exports](#)[Error Reports](#)[History](#)[InstAccess](#)[tokens](#)[JWTs](#)[Late Policy](#)[Learning Object](#)[Dates](#)[Line Items](#)[Live Assessments](#)[Logins](#)[LTI Resource](#)[Links](#)[Media Objects](#)[Moderated](#)[Grading Modules](#)[Names and Role](#)[New Quiz Items](#)[New Quizzes](#)[New Quizzes](#)[Reports](#)[Notification Preferences](#)[Originality Reports](#)[Outcome Groups](#)[Outcome Imports](#)[Outcome Results](#)[Outcomes](#)[Pages](#)[Peer Reviews](#)[Planner](#)[Poll Sessions](#)[Poll Choices](#)[Polls](#)[Poll Submissions](#)[Proficiency Ratings](#)[Progress](#)[Public](#)[JWK Quiz Assignment Overrides](#)[Quiz Extensions](#)[Quiz IP Filters](#)[Quiz Question Groups](#)[Quiz Questions](#)[Quiz Reports](#)[Quiz Statistics](#)[Quiz Submission Events](#)[Quiz Submission Files](#)[Quiz Submission Questions](#)[Quiz Submission User List](#)[Quiz Submissions](#)[Quizzes](#)[Result Roles](#)[Rubrics Sandboxes](#)[Score Search](#)[Sections](#)[Services](#)[Shared Brand Configs](#)[SIS Import Errors](#)[SIS Imports](#)[SIS Integration](#)[Smart Search](#)[Submission Comments](#)[Submissions](#)[Tabs](#)[Temporary Enrollment](#)[Pairings](#)[User](#)[Observees](#)[Users](#)[What If](#)[Grades](#)
- ☐ Outcomes
- [Outcomes CSV Format](#)
- ☐ Group Categories
- [Group Categories CSV Format](#)
- ☐ SIS
- [SIS CSV Format](#)
- ☐ External Tools
- ☐ LTI
- [Introduction](#)[Registration](#)[Launch Overview](#)[Configuring Variable Substitutions](#)[Deep Linking](#)[Grading](#)[Provisioning](#)[Post Message](#)
- ☐ Placements
- [Placements Overview](#)[Navigation](#)[Homework Submission](#)[Editor Button](#)[Migration](#)[Selection](#)[Link Selection \(Modules\)](#)[Assignment Selection](#)[Collaborations](#)
- [xAPI Canvas Roles](#)
- ☐ Plagiarism Detection Platform
- [Overview](#)[Plagiarism Detection Platform](#)[Assignments](#)[Plagiarism Detection Platform Users](#)[Plagiarism Detection Submissions](#)[Webhooks](#)[Subscriptions for Plagiarism Platform](#)[JWT Access Tokens](#)
- ☐ Data Services
- ☐ Live Events
- ☐ Overview
- [Introduction](#)[Setup](#)[Caliper](#)[Metadata](#)
- [Event Format](#)
- ☐ Canvas
- [Account](#)[Asset](#)[Assignment](#)[Attachment](#)[Content](#)[Conversation](#)[Course](#)[Discussion](#)[Enrollment](#)[Grade](#)[Group](#)[Learning](#)[Logged](#)[Module](#)[Outcome](#)[Outcomes](#)[Plagiarism](#)[Quiz](#)[Rubric](#)[Sis](#)[Submission](#)[Syllabus](#)[User](#)[Wiki](#)
- ☐ Caliper IMS 1.1
- [Assessment](#)[Basic](#)[Forum](#)[Grading](#)[Navigation](#)[Events](#)[Session](#)

All API Resources

Access Tokens	<a href="#">Show an access token</a> <a href="#">Create an access token</a> <a href="#">Update an access token</a> <a href="#">Delete an access token</a>
Account Calendars	<a href="#">List available account calendars</a> <a href="#">Get a single account calendar</a> <a href="#">Update a calendar</a> <a href="#">Update several calendars</a> <a href="#">List all account calendars</a> <a href="#">Count of all visible account calendars</a>
Account Domain Lookups	<a href="#">Search account domains</a>
Account Notifications	<a href="#">Index of active global notification for the user</a> <a href="#">Show a global notification</a> <a href="#">Close notification for user</a> <a href="#">Create a global notification</a> <a href="#">Update a global notification</a>
Account Reports	<a href="#">List Available Reports</a> <a href="#">Start a Report</a> <a href="#">Index of Reports</a> <a href="#">Status of a Report</a> <a href="#">Delete a Report</a> <a href="#">Abort a Report</a>
Accounts	<a href="#">List accounts</a> <a href="#">Get accounts that admins can manage</a> <a href="#">Get accounts that users can create</a> <a href="#">courses in</a> <a href="#">List accounts for course admins</a> <a href="#">Get a single account</a> <a href="#">Settings</a> <a href="#">List environment settings</a> <a href="#">Permissions</a> <a href="#">Get the sub-accounts of an account</a> <a href="#">Get the Terms of Service</a> <a href="#">Get help links</a> <a href="#">Get the manually-created courses sub-account for the domain root account</a> <a href="#">List active courses in an account</a> <a href="#">Update an account</a> <a href="#">Delete a user from the root account</a> <a href="#">Restore a deleted user from a root account</a>
Subaccounts	<a href="#">Create a new sub-account</a> <a href="#">Delete a sub-account</a>
Accounts (LTI)	<a href="#">Get account</a>
Admins	<a href="#">Make an account admin</a> <a href="#">Remove account admin</a> <a href="#">List account admins</a> <a href="#">List my admin roles</a>
Analytics	<a href="#">Get department-level participation data</a> <a href="#">Get department-level grade data</a> <a href="#">Get department-level statistics</a> <a href="#">Get department-level statistics, broken down by subaccount</a> <a href="#">Get course-level participation data</a> <a href="#">Get course-level assignment data</a> <a href="#">Get course-level student summary data</a> <a href="#">Get user-in-a-course-level participation data</a> <a href="#">Get user-in-a-course-level assignment data</a> <a href="#">Get user-in-a-course-level messaging data</a>
Announcement External	
Feeds	<a href="#">List external feeds</a> <a href="#">Create an external feed</a> <a href="#">Delete an external feed</a>
Announcements	<a href="#">List announcements</a>
API Token Scopes	<a href="#">List scopes</a>
Appointment Groups	<a href="#">List appointment groups</a> <a href="#">Create an appointment group</a> <a href="#">Get a single appointment group</a> <a href="#">Update an appointment group</a> <a href="#">Delete an appointment group</a> <a href="#">List user participants</a> <a href="#">List student group participants</a> <a href="#">Get next appointment</a>
Assignment Extensions	<a href="#">Set extensions for student assignment submissions</a>
Assignment Groups	<a href="#">List assignment groups</a> <a href="#">Get an Assignment Group</a> <a href="#">Create an Assignment Group</a> <a href="#">Edit an Assignment Group</a> <a href="#">Destroy an Assignment Group</a>
Assignments	<a href="#">Delete an assignment</a> <a href="#">List assignments</a> <a href="#">List assignments for user</a> <a href="#">Duplicate assignment</a> <a href="#">List group members for a student on an assignment</a> <a href="#">Get a single assignment</a> <a href="#">Create an assignment</a> <a href="#">Edit an assignment</a> <a href="#">Bulk update assignment dates</a>
Assignment Overrides	<a href="#">List assignment overrides</a> <a href="#">Get a single assignment override</a> <a href="#">Redirect to the assignment override for a group</a> <a href="#">Redirect to the assignment override for a section</a> <a href="#">Create an assignment override</a> <a href="#">Update an assignment override</a> <a href="#">Delete an assignment override</a> <a href="#">Batch retrieve overrides in a course</a> <a href="#">Batch create overrides in a course</a> <a href="#">Batch update overrides in a course</a>
Authentication Providers	<a href="#">List authentication providers</a> <a href="#">Get authentication provider</a> <a href="#">Update authentication provider</a> <a href="#">Delete authentication provider</a> <a href="#">Restore a deleted authentication providers</a> <a href="#">show account auth settings</a> <a href="#">update account auth settings</a>
Authentications Log	<a href="#">Query by login</a> <a href="#">Query by account</a> <a href="#">Query by user</a>
Blackout Dates	<a href="#">List blackout dates</a> <a href="#">Get a single blackout date</a> <a href="#">New Blackout Date</a> <a href="#">Create Blackout Date</a> <a href="#">Update Blackout Date</a> <a href="#">Delete Blackout Date</a> <a href="#">Update a list of Blackout Dates</a>
Block Editor/Template	<a href="#">List block templates</a>
Blueprint Management	<a href="#">Get blueprint information</a> <a href="#">Get associated course information</a> <a href="#">Update associated courses</a> <a href="#">Begin a migration to push to associated courses</a> <a href="#">Set or remove restrictions on a blueprint course object</a> <a href="#">Get unsynched changes</a>
Blueprint Course History	<a href="#">List blueprint migrations</a> <a href="#">Show a blueprint migration</a> <a href="#">Get migration details</a>
Associated Course History	<a href="#">List blueprint subscriptions</a> <a href="#">List blueprint imports</a> <a href="#">Show a blueprint import</a> <a href="#">Get import details</a>
Bookmarks	<a href="#">List bookmarks</a> <a href="#">Create bookmark</a> <a href="#">Get bookmark</a> <a href="#">Update bookmark</a> <a href="#">Delete bookmark</a>
Brand Configs	<a href="#">Get the brand config variables that should be used for this domain</a>
Calendar Events	<a href="#">List calendar events</a> <a href="#">List calendar events for a user</a> <a href="#">Create a calendar event</a> <a href="#">Get a single calendar event or assignment</a> <a href="#">Reserve a time slot</a> <a href="#">Update a calendar event</a> <a href="#">Delete a calendar event</a> <a href="#">Save enabled account calendars</a> <a href="#">Set a course timetable</a> <a href="#">Get course timetable</a> <a href="#">Create or update events directly for a course timetable</a>
Collaborations	<a href="#">List collaborations</a> <a href="#">List members of a collaboration</a> <a href="#">List potential members</a>
CommMessages	<a href="#">List of CommMessages for a user</a>
Communication Channels	<a href="#">List user communication channels</a> <a href="#">Create a communication channel</a> <a href="#">Delete a communication channel</a> <a href="#">Delete a push notification endpoint</a>
Conferences	<a href="#">List conferences</a> <a href="#">List conferences for the current user</a>
Content Exports	<a href="#">List content exports</a> <a href="#">Show content export</a> <a href="#">Export content</a>
Migration Issues	<a href="#">List migration issues</a> <a href="#">Get a migration issue</a> <a href="#">Update a migration issue</a>
Content Migrations	<a href="#">List content migrations</a> <a href="#">Get a content migration</a> <a href="#">Create a content migration</a> <a href="#">Update a content migration</a> <a href="#">List Migration Systems</a> <a href="#">List items for selective import</a> <a href="#">Get asset id mapping</a>
Content Security Policy	<a href="#">Get current settings for account or course</a> <a href="#">Enable, disable, or clear explicit CSP setting</a> <a href="#">Lock or unlock current CSP settings for sub-accounts and courses</a> <a href="#">Add an allowed domain to account</a> <a href="#">Add multiple allowed domains to an account</a> <a href="#">Remove a domain from account</a>
Settings	<a href="#">Create a content share</a> <a href="#">List content shares</a> <a href="#">Get unread shares count</a> <a href="#">Get content share</a> <a href="#">Remove content share</a> <a href="#">Add users to content share</a> <a href="#">Update a content share</a>
Content Shares	
Conversations	<a href="#">List conversations</a> <a href="#">Create a conversation</a> <a href="#">Get running batches</a> <a href="#">Get a single conversation</a> <a href="#">Edit a conversation</a> <a href="#">Mark all as read</a> <a href="#">Delete a conversation</a> <a href="#">Add recipients</a> <a href="#">Add a message</a> <a href="#">Delete a message</a> <a href="#">Batch update conversations</a> <a href="#">Find recipients</a> <a href="#">Unread count</a>
Course Audit log	<a href="#">Query by course</a> <a href="#">Query by account</a>
Course Pace	<a href="#">Show a Course pace</a> <a href="#">Create a Course pace</a> <a href="#">Update a Course pace</a> <a href="#">Delete a Course pace</a>
Course Quiz Extensions	<a href="#">Set extensions for student quiz submissions</a>
Courses	<a href="#">List your courses</a> <a href="#">List courses for a user</a> <a href="#">Get user progress</a> <a href="#">Create a new course</a> <a href="#">Upload a file</a> <a href="#">List students</a> <a href="#">List users in course</a> <a href="#">List recently logged in students</a> <a href="#">Get single user</a> <a href="#">Search for content share users</a> <a href="#">Preview processed html</a> <a href="#">Course activity stream</a> <a href="#">Course activity stream summary</a> <a href="#">Course TODO items</a> <a href="#">Delete</a> <a href="#">Conclude a course</a> <a href="#">Get course settings</a> <a href="#">Update course settings</a> <a href="#">Return test student for course</a> <a href="#">Get a single course</a> <a href="#">Update a course</a> <a href="#">Update courses</a> <a href="#">Reset a course</a> <a href="#">Get effective due dates</a> <a href="#">Permissions</a> <a href="#">Get bulk user progress</a> <a href="#">Remove quiz migration alert</a> <a href="#">Get course copy status</a> <a href="#">Copy course content</a>
Custom Gradebook Columns	<a href="#">List custom gradebook columns</a> <a href="#">Create a custom gradebook column</a> <a href="#">Update a custom gradebook column</a> <a href="#">Delete a custom gradebook column</a> <a href="#">Reorder custom columns</a>
Custom Gradebook Column Data	<a href="#">List entries for a column</a> <a href="#">Update column data</a> <a href="#">Bulk update column data</a>
Discussion Topics	<a href="#">List discussion topics</a> <a href="#">Create a new discussion topic</a> <a href="#">Update a topic</a> <a href="#">Delete a topic</a> <a href="#">Reorder pinned topics</a> <a href="#">Update an entry</a> <a href="#">Delete an entry</a> <a href="#">Get a single topic</a> <a href="#">Summary</a> <a href="#">Disable summary</a> <a href="#">Summary</a> <a href="#">Feedback</a> <a href="#">Get the full topic</a> <a href="#">Post an entry</a> <a href="#">Duplicate discussion topic</a> <a href="#">List topic entries</a> <a href="#">Post a reply</a> <a href="#">List entry replies</a> <a href="#">List entries</a> <a href="#">Mark topic as read</a> <a href="#">Mark all topic as read</a> <a href="#">Mark topic as unread</a> <a href="#">Mark all entries as unread</a> <a href="#">Mark entry as read</a> <a href="#">Mark entry as unread</a> <a href="#">Rate entry</a> <a href="#">Subscribe to a topic</a> <a href="#">Unsubscribe from a topic</a>
Enrollment Terms	<a href="#">Create enrollment term</a> <a href="#">Update enrollment term</a> <a href="#">Delete enrollment term</a> <a href="#">List enrollment terms</a> <a href="#">Retrieve enrollment term</a>
Enrollments	<a href="#">List enrollments</a> <a href="#">Enrollment by ID</a> <a href="#">Enroll a user</a> <a href="#">Conclude, deactivate, or delete an enrollment</a> <a href="#">Accept Course Invitation</a> <a href="#">Reject Course Invitation</a> <a href="#">Re-activate an enrollment</a> <a href="#">Add last attended date</a> <a href="#">Show Temporary Enrollment recipient and provider status</a>
ePortfolios	<a href="#">Get all ePortfolios for a User</a> <a href="#">Get an ePortfolio</a> <a href="#">Delete an ePortfolio</a> <a href="#">Get ePortfolio</a> <a href="#">Pages</a> <a href="#">Moderate an ePortfolio</a> <a href="#">Moderate all ePortfolios for a User</a> <a href="#">Restore a deleted ePortfolio</a>
ePub Exports	<a href="#">List courses with their latest ePub export</a> <a href="#">Create ePub Export</a> <a href="#">Show ePub export</a>
Error Reports	<a href="#">Create Error Report</a>

External Tools	<a href="#">List external tools</a> <a href="#">Get a sessionless launch url for an external tool</a> <a href="#">Get a single external tool</a> <a href="#">Create an external tool</a> <a href="#">Edit an external tool</a> <a href="#">Delete an external tool</a> <a href="#">Add tool to RCF Favorites</a> <a href="#">Remove tool from RCF Favorites</a> <a href="#">Add tool to Top Navigation Favorites</a> <a href="#">Remove tool from Top Navigation Favorites</a> <a href="#">Get visible course navigation tools</a> <a href="#">Get visible course navigation tools for a single course</a>
Favorites	<a href="#">List favorite courses</a> <a href="#">List favorite groups</a> <a href="#">Add course to favorites</a> <a href="#">Add group to favorites</a> <a href="#">Remove course from favorites</a> <a href="#">Remove group from favorites</a> <a href="#">Reset course favorites</a> <a href="#">Reset group favorites</a>
Feature Flags	<a href="#">List features</a> <a href="#">List enabled features</a> <a href="#">List environment features</a> <a href="#">Get feature flag</a> <a href="#">Set feature flag</a> <a href="#">Remove feature flag</a>
Files	<a href="#">Get quota information</a> <a href="#">List files</a> <a href="#">Get public inline preview url</a> <a href="#">Get file</a> <a href="#">Translate file reference</a> <a href="#">Update file</a> <a href="#">Delete file</a> <a href="#">Get icon metadata</a> <a href="#">Reset link verifier</a>
Folders	<a href="#">List folders</a> <a href="#">List folders and files</a> <a href="#">List all folders</a> <a href="#">Resolve path</a> <a href="#">Get folder</a> <a href="#">Update folder</a> <a href="#">Create folder</a> <a href="#">Delete folder</a> <a href="#">Upload a file</a> <a href="#">Copy a file</a> <a href="#">Copy a folder</a> <a href="#">Get uploaded media folder for user</a>
Usage Rights	<a href="#">Set usage rights</a> <a href="#">Remove usage rights</a> <a href="#">List licenses</a>
Grade Change Log	<a href="#">Query by assignment</a> <a href="#">Query by course</a> <a href="#">Query by student</a> <a href="#">Query by grader</a> <a href="#">Advanced query</a>
Gradebook History	<a href="#">Days in gradebook history for this course</a> <a href="#">Details for a given date in gradebook history for this course</a> <a href="#">Lists submissions</a> <a href="#">List uncollated submission versions</a>
Grading Period Sets	<a href="#">List grading period sets</a> <a href="#">Create a grading period set</a> <a href="#">Update a grading period set</a> <a href="#">Delete a grading period set</a>
Grading Periods	<a href="#">List grading periods</a> <a href="#">Get a single grading period</a> <a href="#">Update a single grading period</a> <a href="#">Delete a grading period</a> <a href="#">Batch update grading periods</a>
Grading Standards	<a href="#">Create a new grading standard</a> <a href="#">List the grading standards available in a context</a> <a href="#">Get a single grading standard in a context</a>
Group Categories	<a href="#">List group categories for a context</a> <a href="#">Get a single group category</a> <a href="#">Create a Group Category</a> <a href="#">Import category groups</a> <a href="#">Update a Group Category</a> <a href="#">Delete a Group Category</a> <a href="#">List groups in group category</a> <a href="#">export groups in and users in category</a> <a href="#">List users in group category</a> <a href="#">Assign unassigned members</a>
Groups	<a href="#">List your groups</a> <a href="#">List the groups available in a context</a> <a href="#">Get a single group</a> <a href="#">Create a group</a> <a href="#">Edit a group</a> <a href="#">Delete a group</a> <a href="#">List group's users</a> <a href="#">Upload a file</a> <a href="#">Preview processed html</a> <a href="#">Group activity stream</a> <a href="#">Group activity stream summary</a> <a href="#">Permissions</a>
Group Memberships	<a href="#">Invite others to a group</a> <a href="#">List group memberships</a> <a href="#">Get a single group membership</a> <a href="#">Create a membership</a> <a href="#">Update a membership</a> <a href="#">Leave a group</a>
History	<a href="#">List recent history for a user</a>
InstAccess tokens	<a href="#">Create InstAccess token</a>
JWTs	<a href="#">Create JWT</a> <a href="#">Refresh JWT</a>
Late Policy	<a href="#">Get a late policy</a> <a href="#">Create a late policy</a> <a href="#">Patch a late policy</a>
Learning Object Dates	<a href="#">Get a learning object's date information</a> <a href="#">Update a learning object's date information</a>
Line Items	<a href="#">Create a Line Item</a> <a href="#">Update a Line Item</a> <a href="#">Show a Line Item</a> <a href="#">List line items</a> <a href="#">Delete a Line Item</a>
LiveAssessments	<a href="#">Create live assessment results</a> <a href="#">List live assessment results</a> <a href="#">Create or find a live assessment</a> <a href="#">List live assessments</a>
Logins	<a href="#">List user logins</a> <a href="#">Kickoff password recovery flow</a> <a href="#">Create a user login</a> <a href="#">Edit a user login</a> <a href="#">Delete a user login</a>
LTI Resource Links	<a href="#">List LTI Resource Links</a> <a href="#">Show an LTI Resource Link</a> <a href="#">Create an LTI Resource Link</a> <a href="#">Bulk Create LTI Resource Links</a> <a href="#">Update an LTI Resource Link</a> <a href="#">Delete an LTI Resource Link</a>
Media Tracks	<a href="#">List media tracks for a Media Object or Attachment</a> <a href="#">Update Media Tracks</a>
Media Objects	<a href="#">List Media Objects</a> <a href="#">Update Media Object</a>
Moderation Set	<a href="#">List students selected for moderation</a> <a href="#">Select students for moderation</a>
Provisional Grades	<a href="#">Bulk select provisional grades</a> <a href="#">Show provisional grade status for a student</a> <a href="#">Select provisional grade</a> <a href="#">Publish provisional grades for an assignment</a>
Anonymous Provisional Grades	<a href="#">Show provisional grade status for a student</a>
Modules	<a href="#">List modules</a> <a href="#">Show module</a> <a href="#">Create a module</a> <a href="#">Update a module</a> <a href="#">Delete module</a> <a href="#">Re-lock module progressions</a>
Module Items	<a href="#">List module items</a> <a href="#">Show module item</a> <a href="#">Create a module item</a> <a href="#">Update a module item</a> <a href="#">Select a mastery path</a> <a href="#">Delete module item</a> <a href="#">Mark module item as done/not done</a> <a href="#">Get module item sequence</a> <a href="#">Mark module item read</a>
Module Assignment Overrides	<a href="#">List a module's overrides</a> <a href="#">Update a module's overrides</a>
Names and Role	<a href="#">List Course Memberships</a> <a href="#">List Group Memberships</a>
New Quiz Items	<a href="#">Get a quiz item</a> <a href="#">List quiz items</a> <a href="#">Create a quiz item</a> <a href="#">Update a quiz item</a> <a href="#">Delete a quiz item</a> <a href="#">Get items media upload url</a>
New Quizzes	<a href="#">Get a new quiz</a> <a href="#">List new quizzes</a> <a href="#">Create a new quiz</a> <a href="#">Update a single quiz</a> <a href="#">Delete a new quiz</a>
New Quizzes Reports	<a href="#">Create a quiz report</a>
Notification Preferences	<a href="#">List preferences</a> <a href="#">List of preference categories</a> <a href="#">Get a preference</a> <a href="#">Update a preference</a> <a href="#">Update preferences by category</a> <a href="#">Update multiple preferences</a>
Originality Reports	<a href="#">Create an Originality Report</a> <a href="#">Edit an Originality Report</a> <a href="#">Show an Originality Report</a>
Outcome Groups	<a href="#">Redirect to root outcome group for context</a> <a href="#">Get all outcome groups for context</a> <a href="#">Get all outcome links for context</a> <a href="#">Show an outcome group</a> <a href="#">Update an outcome group</a> <a href="#">Delete an outcome group</a> <a href="#">List linked outcomes</a> <a href="#">Create/link an outcome</a> <a href="#">Unlink an outcome</a> <a href="#">List subgroups</a> <a href="#">Create a subgroup</a> <a href="#">Import an outcome group</a>
Outcome Imports	<a href="#">Import Outcomes</a> <a href="#">Get Outcome import status</a> <a href="#">Get IDs of outcome groups created after successful import</a>
Outcome Results	<a href="#">Get outcome results</a> <a href="#">Set outcome ordering for LMGB</a> <a href="#">Get outcome result rollups</a>
Outcomes	<a href="#">Show an outcome</a> <a href="#">Update an outcome</a> <a href="#">Get aligned assignments for an outcome in a course for a particular student</a>
Pages	<a href="#">Show front page</a> <a href="#">Duplicate page</a> <a href="#">Update/create front page</a> <a href="#">List pages</a> <a href="#">Create page</a> <a href="#">Show page</a> <a href="#">Update/create page</a> <a href="#">Delete page</a> <a href="#">List revisions</a> <a href="#">Show revision</a> <a href="#">Revert to revision</a>
Peer Reviews	<a href="#">Get all Peer Reviews</a> <a href="#">Create Peer Review</a> <a href="#">Delete Peer Review</a>
Plagiarism Detection	
Platform Assignments	<a href="#">Get a single assignment (fId)</a>
Plagiarism Detection	
Platform Users	<a href="#">Get a single user (fId)</a> <a href="#">Get all users in a group (fId)</a>
Plagiarism Detection Submissions	<a href="#">Get a single submission</a> <a href="#">Get the history of a single submission</a>
Planner	<a href="#">List planner items</a>
Planner Notes	<a href="#">List planner notes</a> <a href="#">Show a planner note</a> <a href="#">Update a planner note</a> <a href="#">Create a planner note</a> <a href="#">Delete a planner note</a>
Planner Overrides	<a href="#">List planner overrides</a> <a href="#">Show a planner override</a> <a href="#">Update a planner override</a> <a href="#">Create a planner override</a> <a href="#">Delete a planner override</a>
Poll Sessions	<a href="#">List poll sessions for a poll</a> <a href="#">Get the results for a single poll session</a> <a href="#">Create a single poll session</a> <a href="#">Update a single poll session</a> <a href="#">Delete a poll session</a> <a href="#">Open a poll session</a> <a href="#">Close an opened poll session</a> <a href="#">List opened poll sessions</a> <a href="#">List closed poll sessions</a>
PollChoices	<a href="#">List poll choices in a poll</a> <a href="#">Get a single poll choice</a> <a href="#">Create a single poll choice</a> <a href="#">Update a single poll choice</a> <a href="#">Delete a poll choice</a>
Polls	<a href="#">List polls</a> <a href="#">Get a single poll</a> <a href="#">Create a single poll</a> <a href="#">Update a single poll</a> <a href="#">Delete a poll</a>
PollSubmissions	<a href="#">Get a single poll submission</a> <a href="#">Create a single poll submission</a>
Proficiency Ratings	<a href="#">Create/update proficiency ratings</a> <a href="#">Get proficiency ratings</a>
Progress	<a href="#">Query progress</a> <a href="#">Cancel progress</a> <a href="#">Query progress</a>
Public JWK	<a href="#">Update Public JWK</a>
Quiz Assignment Overrides	<a href="#">Retrieve assignment-overridden dates for Classic Quizzes</a> <a href="#">Retrieve assignment-overridden dates for New Quizzes</a>
Quiz Extensions	<a href="#">Set extensions for student quiz submissions</a>
Quiz IP Filters	<a href="#">Get available quiz IP filters</a>
Quiz Question Groups	<a href="#">Get a single quiz group</a> <a href="#">Create a question group</a> <a href="#">Update a question group</a> <a href="#">Delete a question group</a> <a href="#">Reorder question groups</a>
Quiz Questions	<a href="#">List questions in a quiz or a submission</a> <a href="#">Get a single quiz question</a> <a href="#">Create a single quiz question</a> <a href="#">Update an existing quiz question</a> <a href="#">Delete a quiz question</a>
Quiz Reports	<a href="#">Retrieve all quiz reports</a> <a href="#">Create a quiz report</a> <a href="#">Get a quiz report</a> <a href="#">Abort the generation of a report, or remove a previously generated one</a>
Quiz Statistics	<a href="#">Fetching the latest quiz statistics</a>
Quiz Submission Events	<a href="#">Submit captured events</a> <a href="#">Retrieve captured events</a>
Quiz Submission Files	<a href="#">Upload a file</a>
Quiz Submission Questions	<a href="#">Get all quiz submission questions</a> <a href="#">Answering questions</a> <a href="#">Get a formatted student numerical answer</a> <a href="#">Flagging a question</a> <a href="#">Unflagging a question</a>
Quiz Submission User List	<a href="#">Send a message to unsubmitted or submitted users for the quiz</a>
Quiz Submissions	<a href="#">Get all quiz submissions</a> <a href="#">Get the quiz submission</a> <a href="#">Get a single quiz submission</a> <a href="#">Create the quiz submission (start a quiz-taking session)</a> <a href="#">Update student question scores and comments</a> <a href="#">Complete the quiz submission (turn it in)</a> <a href="#">Get current quiz submission times</a>
Quizzes	<a href="#">List quizzes in a course</a> <a href="#">Get a single quiz</a> <a href="#">Create a quiz</a> <a href="#">Edit a quiz</a> <a href="#">Delete a quiz</a> <a href="#">Reorder quiz items</a> <a href="#">Validate quiz access code</a>
Result	<a href="#">Show a collection of Results</a> <a href="#">Show a Result</a>
Roles	<a href="#">List roles</a> <a href="#">Get a single role</a> <a href="#">Create a new role</a> <a href="#">Deactivate a role</a> <a href="#">Activate a role</a> <a href="#">Update a role</a>
Rubrics	<a href="#">Create a single rubric</a> <a href="#">Update a single rubric</a> <a href="#">Delete a single rubric</a> <a href="#">List rubrics</a> <a href="#">Get a single rubric</a> <a href="#">Get the courses and assignments for</a> <a href="#">Creates a rubric using a CSV file</a> <a href="#">Templated file for importing a rubric</a> <a href="#">Get the status of a rubric import</a>
RubricAssessments	<a href="#">Create a single rubric assessment</a> <a href="#">Update a single rubric assessment</a> <a href="#">Delete a single rubric assessment</a>
RubricAssociations	<a href="#">Create a RubricAssociation</a> <a href="#">Update a RubricAssociation</a> <a href="#">Delete a RubricAssociation</a>
Sandboxes	<a href="#">Download UUID Mapping for this Sandbox</a>
Score	<a href="#">Create a Score</a>
Search	<a href="#">Find recipients</a> <a href="#">List all courses</a>
Sections	<a href="#">List course sections</a> <a href="#">Create course section</a> <a href="#">Cross-list a Section</a> <a href="#">De-cross-list a Section</a> <a href="#">Edit a section</a> <a href="#">Get section information</a> <a href="#">Delete a section</a>
Services	<a href="#">Get Kaltura config</a> <a href="#">Start Kaltura session</a>
Shared Brand Configs	<a href="#">Share a BrandConfig (Theme)</a> <a href="#">Update a shared theme</a> <a href="#">Un-share a BrandConfig (Theme)</a>
SIS Import Errors	<a href="#">Get SIS import error list</a>
SIS Imports	<a href="#">Get SIS import list</a> <a href="#">Get the current importing SIS import</a> <a href="#">Import SIS data</a> <a href="#">Get SIS import status</a> <a href="#">Restore workflow states of SIS imported items</a> <a href="#">Abort SIS import</a> <a href="#">Abort all pending SIS imports</a>
SIS Integration	<a href="#">Retrieve assignments enabled for grade export to SIS</a> <a href="#">Disable assignments currently enabled for grade export to SIS</a>
Smart Search	<a href="#">Search course content</a>
Submission Comments	<a href="#">Edit a submission comment</a> <a href="#">Delete a submission comment</a> <a href="#">Upload a file</a>
Submissions	<a href="#">Submit an assignment</a> <a href="#">List assignment submissions</a> <a href="#">List submissions for multiple assignments</a> <a href="#">Get a single submission</a> <a href="#">Get a single submission by anonymous id</a> <a href="#">Upload a file</a> <a href="#">Grade or comment on a submission</a> <a href="#">Grade or comment on a submission by anonymous id</a> <a href="#">List gradeable students</a> <a href="#">List multiple assignments</a> <a href="#">gradeable students</a> <a href="#">Grade or comment on multiple submissions</a> <a href="#">Mark submission as read</a> <a href="#">Mark submission as unread</a> <a href="#">Mark bulk submissions as read</a> <a href="#">Mark submission id as read</a> <a href="#">Clear unread status for all submissions</a> <a href="#">Get rubric assessments</a> <a href="#">read state</a> <a href="#">Mark rubric assessments as read</a> <a href="#">Get document annotations</a> <a href="#">read state</a> <a href="#">Mark document annotations as read</a> <a href="#">Submission Summary</a>
Tags	<a href="#">List available tabs for a course or group</a> <a href="#">Update a tab for a course</a>
Temporary Enrollment Pairings	<a href="#">List temporary enrollment pairings</a> <a href="#">Get a single temporary enrollment pairing</a> <a href="#">New Temporary Enrollment Pairing</a> <a href="#">Create Temporary Enrollment Pairing</a> <a href="#">Delete Temporary Enrollment Pairing</a>
User Observees	<a href="#">List observees</a> <a href="#">List observers</a> <a href="#">Add an observee with credentials</a> <a href="#">Show an observee</a> <a href="#">Show an observer</a> <a href="#">Add an observee</a> <a href="#">Remove an observee</a> <a href="#">Create observer pairing code</a>

Users	<a href="#">List users in account</a> <a href="#">List the activity stream</a> <a href="#">Activity stream summary</a> <a href="#">List the TODO items</a> <a href="#">List counts for todo items</a> <a href="#">List upcoming assignments, calendar events</a> <a href="#">List Missing Submissions</a> <a href="#">Hide a stream item</a> <a href="#">Hide all stream items</a> <a href="#">Upload a file</a> <a href="#">Show user details</a> <a href="#">Create a user</a> <a href="#">[DEPRECATED] Self register a user</a> <a href="#">Update user settings</a> <a href="#">Get custom colors</a> <a href="#">Get custom color</a> <a href="#">Update custom color</a> <a href="#">Update text editor preference</a> <a href="#">Get dashboard positions</a> <a href="#">Update dashboard positions</a> <a href="#">Edit a user</a> <a href="#">Terminate all user sessions</a> <a href="#">Log users out of all mobile apps</a> <a href="#">Merge user into another user</a> <a href="#">Split merged users into separate users</a> <a href="#">Get a Pandata Events jwt token and its expiration date</a> <a href="#">Get a users most recently graded submissions</a> <a href="#">Get user profile</a> <a href="#">List avatar options</a> <a href="#">List user page views</a>
Custom Data	<a href="#">Store custom data</a> <a href="#">Load custom data</a> <a href="#">Delete custom data</a>
Course Nicknames	<a href="#">List course nicknames</a> <a href="#">Get course nickname</a> <a href="#">Set course nickname</a> <a href="#">Remove course nickname</a> <a href="#">Clear course nicknames</a>
Webhooks Subscriptions for Plagiarism Platform	<a href="#">Create a Webhook Subscription</a> <a href="#">Delete a Webhook Subscription</a> <a href="#">Show a single Webhook Subscription</a> <a href="#">Update a Webhook Subscription</a> <a href="#">List all Webhook Subscription for a tool proxy</a>
What If Grades	<a href="#">Update a submission's what-if score and calculate grades</a> <a href="#">Reset the what-if scores for the current user for an entire course and recalculate grades</a>

Show an access token [TokensController#show](#)

GET /api/v1/users/:user\_id/tokens/:id

Scope: url:GET|/api/v1/users/:user\_id/tokens/:id

The ID can be the actual database ID of the token, or the ‘token\_hint’ value.

Create an access token [TokensController#create](#)

POST /api/v1/users/:user\_id/tokens

Scope: url:POST|/api/v1/users/:user\_id/tokens

Create a new access token for the specified user. If the user is not the current user, the token will be created as “pending”, and must be activated by the user before it can be used.

Request Parameters:

Parameter	Type	Description
token[purpose]	Required string	The purpose of the token.
token[expires_at]	DateTime	The time at which the token will expire.
token[scopes][]	Array	The scopes to associate with the token.

Update an access token [TokensController#update](#)

PUT /api/v1/users/:user\_id/tokens/:id

Scope: url:PUT|/api/v1/users/:user\_id/tokens/:id

Update an existing access token.

The ID can be the actual database ID of the token, or the ‘token\_hint’ value.

Regenerating an expired token requires a new expiration date.

Request Parameters:

Parameter	Type	Description
token[purpose]	string	The purpose of the token.
token[expires_at]	DateTime	The time at which the token will expire.
token[scopes][]	Array	The scopes to associate with the token.
token[regenerate]	boolean	Regenerate the actual token.

Delete an access token [TokensController#destroy](#)

DELETE /api/v1/users/:user\_id/tokens/:id

Scope: url:DELETE|/api/v1/users/:user\_id/tokens/:id

The ID can be the actual database ID of the token, or the ‘token\_hint’ value.

List available account calendars [AccountCalendarsApiController#index](#)

GET /api/v1/account\_calendars

Scope: url:GET|/api/v1/account\_calendars

Returns a paginated list of account calendars available to the current user. Includes visible account calendars where the user has an account association.

Request Parameters:

Parameter	Type	Description
search_term	string	When included, searches available account calendars for the term. Returns matching results. Term must be at least 2 characters.

Example Request:

```
curl https://canvas>/api/v1/account_calendars \
-H 'Authorization: Bearer <token>'
```

Returns a list of [AccountCalendar](#) objects

Get a single account calendar [AccountCalendarsApiController#show](#)

GET /api/v1/account\_calendars/:account\_id

Scope: url:GET|/api/v1/account\_calendars/:account\_id

Get details about a specific account calendar.

Example Request:

```
curl https://canvas>/api/v1/account_calendars/204 \
-H 'Authorization: Bearer <token>'
```

Returns an [AccountCalendar](#) object

Update a calendar [AccountCalendarsApiController#update](#)

PUT /api/v1/account\_calendars/:account\_id

Scope: url:PUT|/api/v1/account\_calendars/:account\_id

Set an account calendar’s visibility and auto\_subscribe values. Requires the ‘manage\_account\_calendar\_visibility’ permission on the account.

Request Parameters:

Parameter	Type	Description
visible	boolean	Allow administrators with ‘manage_account_calendar_events’ permission to create events on this calendar, and allow users to view this calendar and its events.
auto_subscribe	boolean	When true, users will automatically see events from this account in their calendar, even if they haven’t manually added that calendar.

Example Request:

```
curl https://<canvas>/api/v1/account_calendars/204 \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'visible=false' \
-d 'auto_subscribe=false'
```

Returns an [AccountCalendar](#) object

[Update several calendars AccountCalendarsApiController#bulk\\_update](#)

PUT /api/v1/accounts/:account\_id/account\_calendars

Scope: url:PUT|/api/v1/accounts/:account\_id/account\_calendars

Set visibility and/or auto\_subscribe on many calendars simultaneously. Requires the 'manage\_account\_calendar\_visibility' permission on the account.

Accepts a JSON array of objects containing 2-3 keys each: 'id' (the account's id, required), 'visible' (a boolean indicating whether the account calendar is visible), and 'auto\_subscribe' (a boolean indicating whether users should see these events in their calendar without manually subscribing).

Returns the count of updated accounts.

Example Request:

```
curl https://<canvas>/api/v1/accounts/1/account_calendars \
-X PUT \
-H 'Authorization: Bearer <token>' \
--data '{"id": 1, "visible": true, "auto_subscribe": false}, {"id": 13, "visible": false, "auto_subscribe": true}]'
```

[List all account calendars AccountCalendarsApiController#all\\_calendars](#)

GET /api/v1/accounts/:account\_id/account\_calendars

Scope: url:GET|/api/v1/accounts/:account\_id/account\_calendars

Returns a paginated list of account calendars for the provided account and its first level of sub-accounts. Includes hidden calendars in the response. Requires the 'manage\_account\_calendar\_visibility' permission.

Request Parameters:

Parameter	Type	Description
search_term	string	When included, searches all descendent accounts of provided account for the term. Returns matching results. Term must be at least 2 characters. Can be combined with a filter value.
filter	string	When included, only returns calendars that are either visible or hidden. Can be combined with a search term. Allowed values: visible, hidden

Example Request:

```
curl https://<canvas>/api/v1/accounts/1/account_calendars \
-H 'Authorization: Bearer <token>'
```

Returns a list of [AccountCalendar](#) objects

[Count of all visible account calendars AccountCalendarsApiController#visible\\_calendars\\_count](#)

GET /api/v1/accounts/:account\_id/visible\_calendars\_count

Scope: url:GET|/api/v1/accounts/:account\_id/visible\_calendars\_count

Returns the number of visible account calendars.

Example Request:

```
curl https://<canvas>/api/v1/accounts/1/visible_calendars_count \
-H 'Authorization: Bearer <token>'
```

[Search account domains](#)

GET /api/v1/accounts/search

Scope: url:GET|/api/v1/accounts/search

Returns a list of up to 5 matching account domains

Partial match on name / domain are supported

Request Parameters:

Parameter	Type	Description
name	string	campus name
domain	string	no description
latitude	number	no description
longitude	number	no description

Example Request:

```
curl https://<canvas>/api/v1/accounts/search \
-G -H 'Authorization: Bearer <ACCESS_TOKEN>' \
-d 'name=utah'
```

Example Response:

```
[
  {
    "name": "University of Utah",
    "domain": "utah.edu",
    "distance": null, // distance is always nil, but preserved in the api response for backwards compatibility
    "authentication_provider": "canvas", // which authentication_provider param to pass to the outh flow; may be NULL
  },
  ...
]
```

[Index of active global notification for the user AccountNotificationsController#user\\_index](#)

GET /api/v1/accounts/:account\_id/account\_notifications

Scope: url:GET|/api/v1/accounts/:account\_id/account\_notifications

Returns a list of all global notifications in the account for the current user Any notifications that have been closed by the user will not be returned, unless a include\_past parameter is passed in as true. Admins can request all global notifications for the account by passing in an include\_all parameter.

Request Parameters:

Parameter	Type	Description
include_past	boolean	Include past and dismissed global announcements.
include_all	boolean	Include all global announcements, regardless of user's role. Only available to account admins.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/2/users/self/account_notifications
```

Returns a list of [AccountNotification](#) objects

[Show a global notification AccountNotificationsController#show](#)

**GET** /api/v1/accounts/:account\_id/account\_notifications/:id

**Scope:** url:GET/api/v1/accounts/:account\_id/account\_notifications/:id

Returns a global notification for the current user A notification that has been closed by the user will not be returned

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/2/users/self/account_notifications/4
```

Returns an [AccountNotification](#) object

[Close notification for user AccountNotificationsController#user\\_close\\_notification](#)

**DELETE** /api/v1/accounts/:account\_id/account\_notifications/:id

**Scope:** url:DELETE/api/v1/accounts/:account\_id/account\_notifications/:id

If the current user no long wants to see this notification it can be excused with this call

**Example Request:**

```
curl -X DELETE -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/2/users/self/account_notifications/4
```

Returns an [AccountNotification](#) object

[Create a global notification AccountNotificationsController#create](#)

**POST** /api/v1/accounts/:account\_id/account\_notifications

**Scope:** url:POST/api/v1/accounts/:account\_id/account\_notifications

Create and return a new global notification for an account.

**Request Parameters:**

Parameter	Type	Description
account_notification[subject]	Required string	The subject of the notification.
account_notification[message]	Required string	The message body of the notification.
account_notification[start_at]	Required DateTime	The start date and time of the notification in ISO8601 format. e.g. 2014-01-01T01:00Z
account_notification[end_at]	Required DateTime	The end date and time of the notification in ISO8601 format. e.g. 2014-01-01T01:00Z
account_notification[icon]	string	The icon to display with the notification. Note: Defaults to warning. Allowed values: warning, information, question, error, calendar
account_notification_roles[]	string	The role(s) to send global notification to. Note: ommitting this field will send to everyone Example: account_notification_roles: ["StudentEnrollment", "TeacherEnrollment"]

**Example Request:**

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/2/account_notifications \
-d 'account_notification[subject]=New notification' \
-d 'account_notification[start_at]=2014-01-01T00:00:00Z' \
-d 'account_notification[end_at]=2014-02-01T00:00:00Z' \
-d 'account_notification[message]=This is a global notification'
```

**Example Response:**

```
{
  "subject": "New notification",
  "start_at": "2014-01-01T00:00:00Z",
  "end_at": "2014-02-01T00:00:00Z",
  "message": "This is a global notification"
}
```

[Update a global notification AccountNotificationsController#update](#)

**PUT** /api/v1/accounts/:account\_id/account\_notifications/:id

**Scope:** url:PUT/api/v1/accounts/:account\_id/account\_notifications/:id

Update global notification for an account.

**Request Parameters:**

Parameter	Type	Description
account_notification[subject]	string	The subject of the notification.
account_notification[message]	string	The message body of the notification.
account_notification[start_at]	DateTime	The start date and time of the notification in ISO8601 format. e.g. 2014-01-01T01:00Z
account_notification[end_at]	DateTime	The end date and time of the notification in ISO8601 format. e.g. 2014-01-01T01:00Z
account_notification[icon]	string	The icon to display with the notification. Allowed values: warning, information, question, error, calendar
account_notification_roles[]	string	The role(s) to send global notification to. Note: ommitting this field will send to everyone Example: account_notification_roles: ["StudentEnrollment", "TeacherEnrollment"]

**Example Request:**

```
curl -X PUT -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/2/account_notifications/1 \
-d 'account_notification[subject]=New notification' \
-d 'account_notification[start_at]=2014-01-01T00:00:00Z' \
-d 'account_notification[end_at]=2014-02-01T00:00:00Z' \
-d 'account_notification[message]=This is a global notification'
```

**Example Response:**

```
{
  "subject": "New notification",
  "start_at": "2014-01-01T00:00:00Z",
  "end_at": "2014-02-01T00:00:00Z",
  "message": "This is a global notification"
}
```

[List Available Reports AccountReportsController#available\\_reports](#)

**GET** /api/v1/accounts/:account\_id/reports

**Scope:** url:GET/api/v1/accounts/:account\_id/reports

Returns a paginated list of reports for the current context.

API response field:

- name  
The name of the report.
- parameters  
The parameters will vary for each report

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/<account_id>/reports/
```

Example Response:

```
{
  {
    "report": "student_assignment_outcome_map_csv",
    "title": "Student Competency",
    "parameters": null
  },
  {
    "report": "grade_export_csv",
    "title": "Grade Export",
    "parameters": {
      "term": {
        "description": "The canvas id of the term to get grades from",
        "required": true
      }
    }
  }
}
```

Start a Report AccountReportsController#create

POST /api/v1/accounts/:account\_id/reports/:report

Scope: url: POST /api/v1/accounts/:account\_id/reports/:report

Generates a report instance for the account. Note that “report” in the request must match one of the available report names. To fetch a list of available report names and parameters for each report (including whether or not those parameters are required), see [List Available Reports](#).

Request Parameters:

Parameter	Type	Description
parameters	string	The parameters will vary for each report. To fetch a list of available parameters for each report, see <a href="#">List Available Reports</a> . A few example parameters have been provided below. Note that the example parameters provided below may not be valid for every report.
parameters[skip_message]	boolean	If true, no message will be sent to the user upon completion of the report.
parameters[course_id]	integer	The id of the course to report on. Note: this parameter has been listed to serve as an example and may not be valid for every report.
parameters[users]	boolean	If true, user data will be included. If false, user data will be omitted. Note: this parameter has been listed to serve as an example and may not be valid for every report.

Example Request:

```
curl -X POST \
https://<canvas>/api/v1/accounts/1/reports/provisioning_csv \
-H 'Authorization: Bearer <token>' \
-H 'Content-Type: multipart/form-data' \
-F 'parameters[users]=true' \
-F 'parameters[courses]=true' \
-F 'parameters[enrollments]=true'
```

Returns a [Report](#) object

Index of Reports AccountReportsController#index

GET /api/v1/accounts/:account\_id/reports/:report

Scope: url: GET /api/v1/accounts/:account\_id/reports/:report

Shows all reports that have been run for the account of a specific type.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/<account_id>/reports/<report_type>
```

Returns a list of [Report](#) objects

Status of a Report AccountReportsController#show

GET /api/v1/accounts/:account\_id/reports/:report/:id

Scope: url: GET /api/v1/accounts/:account\_id/reports/:report/:id

Returns the status of a report.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/<account_id>/reports/<report_type>/<report_id>
```

Returns a [Report](#) object

Delete a Report AccountReportsController#destroy

DELETE /api/v1/accounts/:account\_id/reports/:report/:id

Scope: url: DELETE /api/v1/accounts/:account\_id/reports/:report/:id

Deletes a generated report instance.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
-X DELETE \
https://<canvas>/api/v1/accounts/<account_id>/reports/<report_type>/<id>
```

Returns a [Report](#) object

Abort a Report AccountReportsController#abort

PUT /api/v1/accounts/:account\_id/reports/:report/:id/abort

Scope: url: PUT /api/v1/accounts/:account\_id/reports/:report/:id/abort

Abort a report in progress

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
-X PUT \
https://<canvas>/api/v1/accounts/<account_id>/reports/<report_type>/<id>/abort
```

Returns a [Report](#) object

List accounts AccountsController#index

GET /api/v1/accounts

**Scope:** url:GET /api/v1/accounts

A paginated list of accounts that the current user can view or manage. Typically, students and even teachers will get an empty list in response, only account admins can view the accounts that they are in.

**Request Parameters:**

Parameter	Type	Description
		Array of additional information to include.
		"lti_guid"
		the 'tool_consumer_instance_guid' that will be sent for this account on LTI launches
		"registration_settings"
		returns info about the privacy policy and terms of use
		"services"
include[]	string	returns services and whether they are enabled (requires account management permissions)
		"course_count"
		returns the number of courses directly under each account
		"sub_account_count"
		returns the number of sub-accounts directly under each account
		Allowed values: lti_guid, registration_settings, services, course_count, sub_account_count

Returns a list of [Account](#) objects

[Get accounts that admins can manage AccountsController#manageable\\_accounts](#)

**GET /api/v1/manageable\_accounts**

**Scope:** url:GET /api/v1/manageable\_accounts

A paginated list of accounts where the current user has permission to create or manage courses. List will be empty for students and teachers as only admins can view which accounts they are in.

Returns a list of [Account](#) objects

[Get accounts that users can create courses in AccountsController#course\\_creation\\_accounts](#)

**GET /api/v1/course\_creation\_accounts**

**Scope:** url:GET /api/v1/course\_creation\_accounts

A paginated list of accounts where the current user has permission to create courses.

Returns a list of [Account](#) objects

[List accounts for course admins AccountsController#course\\_accounts](#)

**GET /api/v1/course\_accounts**

**Scope:** url:GET /api/v1/course\_accounts

A paginated list of accounts that the current user can view through their admin course enrollments. (Teacher, TA, or designer enrollments). Only returns "id", "name", "workflow\_state", "root\_account\_id" and "parent\_account\_id"

Returns a list of [Account](#) objects

[Get a single account AccountsController#show](#)

**GET /api/v1/accounts/:id**

**Scope:** url:GET /api/v1/accounts/:id

Retrieve information on an individual account, given by id or sis\_account\_id.

Returns an [Account](#) object

[Settings AccountsController#show\\_settings](#)

**GET /api/v1/accounts/:account\_id/settings**

**Scope:** url:GET /api/v1/accounts/:account\_id/settings

Returns a JSON object containing a subset of settings for the specified account. It's possible an empty set will be returned if no settings are applicable. The caller must be an Account admin with the manage\_account\_settings permission.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/settings \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{ "microsoft_sync_enabled": true, "microsoft_sync_login_attribute_suffix": false}
```

[List environment settings AccountsController#environment](#)

**GET /api/v1/settings/environment**

**Scope:** url:GET /api/v1/settings/environment

Return a hash of global settings for the root account This is the same information supplied to the web interface as ENV.SETTINGS.

**Example Request:**

```
curl 'http://<canvas>/api/v1/settings/environment' \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{ "calendar_contexts_limit": 10, "open_registration": false, ...}
```

[Permissions AccountsController#permissions](#)

**GET /api/v1/accounts/:account\_id/permissions**

**Scope:** url:GET /api/v1/accounts/:account\_id/permissions

Returns permission information for the calling user and the given account. You may use 'self' as the account id to check permissions against the domain root account. The caller must have an account role or admin (teacher/TA/designer) enrollment in a course in the account.

See also the [Course](#) and [Group](#) counterparts.

**Request Parameters:**

Parameter	Type	Description
permissions[]	string	List of permissions to check against the authenticated user. Permission names are documented in the <a href="#">Create a role</a> endpoint.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/self/permissions \
-H 'Authorization: Bearer <token>' \
-d 'permissions[]=manage_account_memberships' \
-d 'permissions[]=become_user'
```

Example Response:

```
{'manage_account_memberships': 'false', 'become_user': 'true'}
```

Get the sub-accounts of an account AccountsController#sub\_accounts

GET /api/v1/accounts/:account\_id/sub\_accounts

Scope: url:GET|/api/v1/accounts/:account\_id/sub\_accounts

List accounts that are sub-accounts of the given account.

Request Parameters:

Parameter	Type	Description
recursive	boolean	If true, the entire account tree underneath this account will be returned (though still paginated). If false, only direct sub-accounts of this account will be returned. Defaults to false.
include[]	string	Array of additional information to include. "course_count" returns the number of courses directly under each account "sub_account_count" returns the number of sub-accounts directly under each account Allowed values: course_count, sub_account_count

Example Request:

```
curl https://<canvas>/api/v1/accounts/<account_id>/sub_accounts \
-H 'Authorization: Bearer <token>'
```

Returns a list of Account objects

Get the Terms of Service AccountsController#terms\_of\_service

GET /api/v1/accounts/:account\_id/terms\_of\_service

Scope: url:GET|/api/v1/accounts/:account\_id/terms\_of\_service

Returns the terms of service for that account

Returns a TermsOfService object

Get help links AccountsController#help\_links

GET /api/v1/accounts/:account\_id/help\_links

Scope: url:GET|/api/v1/accounts/:account\_id/help\_links

Returns the help links for that account

Returns a HelpLinks object

Get the manually-created courses sub-account for the domain root account AccountsController#manually\_created\_courses\_account

GET /api/v1/manually\_created\_courses\_account

Scope: url:GET|/api/v1/manually\_created\_courses\_account

List active courses in an account AccountsController#courses\_api

GET /api/v1/accounts/:account\_id/courses

Scope: url:GET|/api/v1/accounts/:account\_id/courses

Retrieve a paginated list of courses in this account.

Request Parameters:

Parameter	Type	Description
with_enrollments	boolean	If true, include only courses with at least one enrollment. If false, include only courses with no enrollments. If not present, do not filter on course enrollment status.
enrollment_type[]	string	If set, only return courses that have at least one user enrolled in in the course with one of the specified enrollment types. Allowed values: teacher, student, ta, observer, designer
published	boolean	If true, include only published courses. If false, exclude published courses. If not present, do not filter on published status.
completed	boolean	If true, include only completed courses (these may be in state ‘completed’, or their enrollment term may have ended). If false, exclude completed courses. If not present, do not filter on completed status.
blueprint	boolean	If true, include only blueprint courses. If false, exclude them. If not present, do not filter on this basis.
blueprint_associated	boolean	If true, include only courses that inherit content from a blueprint course. If false, exclude them. If not present, do not filter on this basis.
public	boolean	If true, include only public courses. If false, exclude them. If not present, do not filter on this basis.
by_teachers[]	integer	List of User IDs of teachers; if supplied, include only courses taught by one of the referenced users.
by_subaccounts[]	integer	List of Account IDs; if supplied, include only courses associated with one of the referenced subaccounts.
hide_enrollmentless_courses	boolean	If present, only return courses that have at least one enrollment. Equivalent to ‘with_enrollments=true’; retained for compatibility.
state[]	string	If set, only return courses that are in the given state(s). By default, all states but “deleted” are returned. Allowed values: created, claimed, available, completed, deleted, all
enrollment_term_id	integer	If set, only includes courses from the specified term.
search_term	string	The partial course name, code, or full ID to match and return in the results list. Must be at least 3 characters.
include[]	string	• All explanations can be seen in the Course API index documentation • “sections”, “needs_grading_count” and “total_scores” are not valid options at the account level Allowed values: syllabus_body, term, course_progress, storage_quota_used_mb, total_students, teachers, account_name, concluded, post_manually
sort	string	The column to sort results by. Allowed values: course_status, course_name, sis_course_id, teacher, account_name



Parameter	Type	Description
order	string	The order to sort the given column by.
		Allowed values: asc, desc
search_by	string	The filter to search by. “course” searches for course names, course codes, and SIS IDs. “teacher” searches for teacher names
		Allowed values: course, teacher
starts_before	Date	If set, only return courses that start before the value (inclusive) or their enrollment term starts before the value (inclusive) or both the course’s start_at and the enrollment term’s start_at are set to null. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
ends_after	Date	If set, only return courses that end after the value (inclusive) or their enrollment term ends after the value (inclusive) or both the course’s end_at and the enrollment term’s end_at are set to null. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
homeroom	boolean	If set, only return homeroom courses.

Returns a list of [Course](#) objects

Update an account [AccountsController#update](#)

PUT /api/v1/accounts/:id

Scope: url:PUT /api/v1/accounts/:id

Update an existing account.

Request Parameters:

Parameter	Type	Description
account[name]	string	Updates the account name
account[sis_account_id]	string	Updates the account sis_account_id Must have manage_sis permission and must not be a root_account.
account[default_time_zone]	string	The default time zone of the account. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
account[default_storage_quota_mb]	integer	The default course storage quota to be used, if not otherwise specified.
account[default_user_storage_quota_mb]	integer	The default user storage quota to be used, if not otherwise specified.
account[default_group_storage_quota_mb]	integer	The default group storage quota to be used, if not otherwise specified.
account[course_template_id]	integer	The ID of a course to be used as a template for all newly created courses. Empty means to inherit the setting from parent account, 0 means to not use a template even if a parent account has one set. The course must be marked as a template.
account[parent_account_id]	integer	The ID of a parent account to move the account to. The new parent account must be in the same root account as the original. The hierarchy of sub-accounts will be preserved in the new parent account. The caller must be an administrator in both the original parent account and the new parent account.
account[settings][restrict_student_past_view][value]	boolean	Restrict students from viewing courses after end date
account[settings][restrict_student_past_view][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][restrict_student_future_view][value]	boolean	Restrict students from viewing courses before start date
account[settings][microsoft_sync_enabled]	boolean	Determines whether this account has Microsoft Teams Sync enabled or not.
		Note that if you are altering Microsoft Teams sync settings you must enable the Microsoft Group enrollment syncing feature flag. In addition, if you are enabling Microsoft Teams sync, you must also specify a tenant, login attribute, and a remote attribute. Specifying a suffix to use is optional.
account[settings][microsoft_sync_tenant]	string	The tenant this account should use when using Microsoft Teams Sync. This should be an Azure Active Directory domain name.
account[settings][microsoft_sync_login_attribute]	string	The attribute this account should use to lookup users when using Microsoft Teams Sync. Must be one of “sub”, “email”, “oid”, “preferred_username”, or “integration_id”.
account[settings][microsoft_sync_login_attribute_suffix]	string	A suffix that will be appended to the result of the login attribute when associating Canvas users with Microsoft users. Must be under 255 characters and contain no whitespace. This field is optional.
account[settings][microsoft_sync_remote_attribute]	string	The Active Directory attribute to use when associating Canvas users with Microsoft users. Must be one of “mail”, “mailNickname”, or “userPrincipalName”.
account[settings][restrict_student_future_view][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][lock_all_announcements][value]	boolean	Disable comments on announcements
account[settings][lock_all_announcements][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][usage_rights_required][value]	boolean	Copyright and license information must be provided for files before they are published.
account[settings][usage_rights_required][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][restrict_student_future_listing][value]	boolean	Restrict students from viewing future enrollments in course list
account[settings][restrict_student_future_listing][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][conditional_release][value]	boolean	Enable or disable individual learning paths for students based on assessment
account[settings][conditional_release][locked]	boolean	Lock this setting for sub-accounts and courses
account[settings][password_policy]	Hash	Hash of optional password policy configuration parameters for a root account
		allow_login_suspension boolean
		Allow suspension of user logins upon reaching maximum_login_attempts
		require_number_characters boolean
		Require the use of number characters when setting up a new password
		require_symbol_characters boolean
		Require the use of symbol characters when setting up a new password
		minimum_character_length integer
		Minimum number of characters required for a new password
		maximum_login_attempts integer
		Maximum number of login attempts before a user is locked out

Parameter	Type	Description
	<i>Required</i> feature option: Enhance password options	
account[settings][enable_as_k5_account][value]	boolean	Enable or disable Canvas for Elementary for this account
account[settings][use_classic_font_in_k5][value]	boolean	Whether or not the classic font is used on the dashboard. Only applies if enable_as_k5_account is true.
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness
account[settings][lock_outcome_proficiency][value]	DEPRECATED boolean	Restrict instructors from changing mastery scale
account[lock_outcome_proficiency][locked]	DEPRECATED boolean	Lock this setting for sub-accounts and courses
account[settings][lock_proficiency_calculation][value]	DEPRECATED boolean	Restrict instructors from changing proficiency calculation method
account[lock_proficiency_calculation][locked]	DEPRECATED boolean	Lock this setting for sub-accounts and courses
account[services]	Hash	Give this a set of keys and boolean values to enable or disable services matching the keys

Example Request:

```
curl https://<canvas>/api/v1/accounts/<account_id> \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'account[name]=New account name' \
-d 'account[default_time_zone]=Mountain Time (US & Canada)' \
-d 'account[default_storage_quota_mb]=450'
```

Returns an [Account](#) object

Delete a user from the root account [AccountsController#remove\\_user](#)

DELETE /api/v1/accounts/:account\_id/users/:user\_id

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/users/:user\_id

Delete a user record from a Canvas root account. If a user is associated with multiple root accounts (in a multi-tenant instance of Canvas), this action will NOT remove them from the other accounts.

WARNING: This API will allow a user to remove themselves from the account. If they do this, they won't be able to make API calls or log into Canvas at that account.

Example Request:

```
curl https://<canvas>/api/v1/accounts/3/users/5 \
-H 'Authorization: Bearer <ACCESS_TOKEN>' \
-X DELETE
```

Returns an [User](#) object

Restore a deleted user from a root account [AccountsController#restore\\_user](#)

PUT /api/v1/accounts/:account\_id/users/:user\_id/restore

**Scope:** url:PUT|/api/v1/accounts/:account\_id/users/:user\_id/restore

Restore a user record along with the most recently deleted pseudonym from a Canvas root account.

Example Request:

```
curl https://<canvas>/api/v1/accounts/3/users/5/restore \
-H 'Authorization: Bearer <ACCESS_TOKEN>' \
-X PUT
```

Returns an [User](#) object

Create a new sub-account [SubAccountsController#create](#)

POST /api/v1/accounts/:account\_id/sub\_accounts

**Scope:** url:POST|/api/v1/accounts/:account\_id/sub\_accounts

Add a new sub-account to a given account.

Request Parameters:

Parameter	Type	Description
account[name]	Required string	The name of the new sub-account.
account[sis_account_id]	string	The account's identifier in the Student Information System.
account[default_storage_quota_mb]	integer	The default course storage quota to be used, if not otherwise specified.
account[default_user_storage_quota_mb]	integer	The default user storage quota to be used, if not otherwise specified.
account[default_group_storage_quota_mb]	integer	The default group storage quota to be used, if not otherwise specified.

Returns an [Account](#) object

Delete a sub-account [SubAccountsController#destroy](#)

DELETE /api/v1/accounts/:account\_id/sub\_accounts/:id

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/sub\_accounts/:id

Cannot delete an account with active courses or active sub\_accounts. Cannot delete a root\_account

Returns an [Account](#) object

Get account [Lti::AccountLookupController#show](#)

GET /api/lti/accounts/:account\_id

**Scope:** url:GET|/api/lti/accounts/:account\_id

Retrieve information on an individual account, given by local or global ID.

Returns an [Account](#) object

Make an account admin [AdminsController#create](#)

POST /api/v1/accounts/:account\_id/admins

**Scope:** url:POST|/api/v1/accounts/:account\_id/admins

Flag an existing user as an admin within the account.

Request Parameters:

Parameter	Type	Description
user_id	Required integer	The id of the user to promote.
role	string	DEPRECATED The user's admin relationship with the account will be created with the given role. Defaults to 'AccountAdmin'.
role_id	integer	The user's admin relationship with the account will be created with the given role. Defaults to the built-in role for 'AccountAdmin'.
send_confirmation	boolean	Send a notification email to the new admin if true. Default is true.

Returns an [Admin](#) object

[Remove account admin AdminsController#destroy](#)

**DELETE** /api/v1/accounts/:account\_id/admins/:user\_id

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/admins/:user\_id

Remove the rights associated with an account admin role from a user.

Request Parameters:

Parameter	Type	Description
role	string	DEPRECATED Account role to remove from the user.
role_id	Required integer	The id of the role representing the user's admin relationship with the account.

Returns an [Admin](#) object

[List account admins AdminsController#index](#)

**GET** /api/v1/accounts/:account\_id/admins

**Scope:** url:GET|/api/v1/accounts/:account\_id/admins

A paginated list of the admins in the account

Request Parameters:

Parameter	Type	Description
user_id[]	[Integer]	Scope the results to those with user IDs equal to any of the IDs specified here.

Returns a list of [Admin](#) objects

[List my admin roles AdminsController#self\\_roles](#)

**GET** /api/v1/accounts/:account\_id/admins/self

**Scope:** url:GET|/api/v1/accounts/:account\_id/admins/self

A paginated list of the current user's roles in the account. The results are the same as those returned by the [List account admins](#) endpoint with `user_id` set to `self`, except the "Admins - Add / Remove" permission is not required.

Returns a list of [Admin](#) objects

[Get department-level participation data](#)

**GET** /api/v1/accounts/:account\_id/analytics/terms/:term\_id/activity

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/terms/:term\_id/activity

**GET** /api/v1/accounts/:account\_id/analytics/current/activity

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/current/activity

**GET** /api/v1/accounts/:account\_id/analytics/completed/activity

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/completed/activity

Returns page view hits summed across all courses in the department. Two groupings of these counts are returned; one by day (`by_date`), the other by category (`by_category`). The possible categories are announcements, assignments, collaborations, conferences, discussions, files, general, grades, groups, modules, other, pages, and quizzes.

This and the other department-level endpoints have three variations which all return the same style of data but for different subsets of courses. All share the prefix `/api/v1/accounts/<account_id>/analytics`. The possible suffixes are:

\* `/current`: includes all available courses in the default term  
\* `/completed`: includes all concluded courses in the default term  
\* `/terms/<term_id>`: includes all available or concluded courses in the given term.

Courses not yet offered or which have been deleted are never included.

`/current` and `/completed` are intended for use when the account has only one term. `/terms/<term_id>` is intended for use when the account has multiple terms.

The action follows the suffix.

Example Request:

```
curl https://<canvas>/api/v1/accounts/<account_id>/analytics/current/activity \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/completed/activity \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/terms/<term_id>/activity \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "by_date": {
    "2012-01-24": 1240,
    "2012-01-27": 912,
  },
  "by_category": {
    "announcements": 54,
    "assignments": 256,
    "collaborations": 18,
    "conferences": 26,
    "discussions": 354,
    "files": 132,
    "general": 59,
    "grades": 177,
    "groups": 132,
    "modules": 71,
    "other": 412,
    "pages": 105,
    "quizzes": 356
  }
}
```

[Get department-level grade data](#)

**GET** /api/v1/accounts/:account\_id/analytics/terms/:term\_id/grades

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/terms/:term\_id/grades

**GET /api/v1/accounts/:account\_id/analytics/current/grades**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/current/grades

**GET /api/v1/accounts/:account\_id/analytics/completed/grades**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/completed/grades

Returns the distribution of grades for students in courses in the department. Each data point is one student's current grade in one course; if a student is in multiple courses, he contributes one value per course, but if he's enrolled multiple times in the same course (e.g. a lecture section and a lab section), he only contributes on value for that course.

Grades are binned to the nearest integer score; anomalous grades outside the 0 to 100 range are ignored. The raw counts are returned, not yet normalized by the total count.

Shares the same variations on endpoint as the participation data.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/analytics/current/grades \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/completed/grades \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/terms/<term_id>/grades \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "0": 95,
  "1": 1,
  "2": 0,
  "3": 0,
  ...
  "93": 125,
  "94": 110,
  "95": 142,
  "96": 157,
  "97": 116,
  "98": 85,
  "99": 63,
  "100": 190
}
```

[Get department-level statistics](#)

**GET /api/v1/accounts/:account\_id/analytics/terms/:term\_id/statistics**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/terms/:term\_id/statistics

**GET /api/v1/accounts/:account\_id/analytics/current/statistics**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/current/statistics

**GET /api/v1/accounts/:account\_id/analytics/completed/statistics**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/completed/statistics

Returns numeric statistics about the department and term (or filter).

Shares the same variations on endpoint as the participation data.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/analytics/current/statistics \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/completed/statistics \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/terms/<term_id>/statistics \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "courses": 27,
  "subaccounts": 3,
  "teachers": 36,
  "students": 418,
  "discussion_topics": 77,
  "media_objects": 219,
  "attachments": 1268,
  "assignments": 290,
}
```

[Get department-level statistics, broken down by subaccount](#)

**GET /api/v1/accounts/:account\_id/analytics/terms/:term\_id/statistics\_by\_subaccount**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/terms/:term\_id/statistics\_by\_subaccount

**GET /api/v1/accounts/:account\_id/analytics/current/statistics\_by\_subaccount**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/current/statistics\_by\_subaccount

**GET /api/v1/accounts/:account\_id/analytics/completed/statistics\_by\_subaccount**

**Scope:** url:GET|/api/v1/accounts/:account\_id/analytics/completed/statistics\_by\_subaccount

Returns numeric statistics about the department subaccounts and term (or filter).

Shares the same variations on endpoint as the participation data.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/analytics/current/statistics_by_subaccount \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/completed/statistics_by_subaccount \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/accounts/<account_id>/analytics/terms/<term_id>/statistics_by_subaccount \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "accounts": [
    {
      "name": "some string",
      "id": 180,
      "courses": 27,
      "teachers": 36,
      "students": 419,
      "discussion_topics": 77,
      "media_objects": 219,
      "attachments": 1268,
      "assignments": 290,
    }
  ]
}
```

[Get course-level participation data](#)

**GET /api/v1/courses/:course\_id/analytics/activity**

**Scope:** url:GET|/api/v1/courses/:course\_id/analytics/activity

Returns page view hits and participation numbers grouped by day through the entire history of the course. Page views is returned as a hash, where the hash keys are dates in the format "YYYY-MM-DD". The page\_views result set includes page views broken out by access category. Participations is returned as an array of dates in the format "YYYY-MM-DD".

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/activity \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
[
{
```

```
    "date": "2012-01-24",
    "participations": 3,
    "views": 10
  }
}
```

Get course-level assignment data

GET /api/v1/courses/:course\_id/analytics/assignments

Scope: url:GET/ /api/v1/courses/:course\_id/analytics/assignments

Returns a list of assignments for the course sorted by due date. For each assignment returns basic assignment information, the grade breakdown, and a breakdown of on-time/late status of homework submissions.

Request Parameters:

Parameter	Type	Description
async	boolean	If async is true, then the course_assignments call can happen asynch-ronously and MAY return a response containing a progress_url key instead of an assignments array. If it does, then it is the caller's responsibility to poll the API again to see if the progress is complete. If the data is ready (possibly even on the first async call) then it will be passed back normally, as documented in the example response.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/assignments \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  {
    "assignment_id": 1234,
    "title": "Assignment 1",
    "points_possible": 10,
    "due_at": "2012-01-20T22:00:00-07:00",
    "unlock_at": "2012-01-20T22:00:00-07:00",
    "muted": false,
    "min_score": 2,
    "max_score": 10,
    "median": 7,
    "first_quartile": 4,
    "third_quartile": 8,
    "tardiness_breakdown": {
      "on_time": 0.75,
      "missing": 0.1,
      "late": 0.15
    }
  },
  {
    "assignment_id": 1235,
    "title": "Assignment 2",
    "points_possible": 15,
    "due_at": "2012-01-26T22:00:00-07:00",
    "unlock_at": null,
    "muted": true,
    "min_score": 8,
    "max_score": 8,
    "median": 8,
    "first_quartile": 8,
    "third_quartile": 8,
    "tardiness_breakdown": {
      "on_time": 0.65,
      "missing": 0.12,
      "late": 0.23
    }
  }
}
```

Get course-level student summary data

GET /api/v1/courses/:course\_id/analytics/student\_summaries

Scope: url:GET/ /api/v1/courses/:course\_id/analytics/student\_summaries

Returns a summary of per-user access information for all students in a course. This includes total page views, total participations, and a breakdown of on-time/late status for all homework submissions in the course.

Each student's summary also includes the maximum number of page views and participations by any student in the course, which may be useful for some visualizations (since determining maximums client side can be tricky with pagination).

Request Parameters:

Parameter	Type	Description
sort_column	string	The order results in which results are returned. Defaults to "name". Allowed values: name, name_descending, score, score_descending, participations, participations_descending, page_views, page_views_descending

student\_id string If set, returns only the specified student.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/student_summaries \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  {
    "id": 2346,
    "page_views": 351,
    "page_views_level": "1",
    "max_page_view": 415,
    "participations": 1,
    "participations_level": "3",
    "max_participations": 10,
    "tardiness_breakdown": {
      "total": 5,
      "on_time": 3,
      "late": 0,
      "missing": 2,
      "floating": 0
    }
  },
  {
    "id": 2345,
    "page_views": 124,
    "participations": 15,
    "tardiness_breakdown": {
      "total": 5,
      "on_time": 1,
      "late": 2,
      "missing": 3,
      "floating": 0
    }
  }
}
```

Get user-in-a-course-level participation data

GET /api/v1/courses/:course\_id/analytics/users/:student\_id/activity

Scope: url:GET/ /api/v1/courses/:course\_id/analytics/users/:student\_id/activity

Returns page view hits grouped by hour, and participation details through the entire history of the course.

'page\_views' are returned as a hash, where the keys are iso8601 dates, bucketed by the hour. 'participations' are returned as an array of hashes, sorted oldest to newest.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/users/<user_id>/activity \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "page_views": {
    "2012-01-24T13:00:00-00:00": 19,
    "2012-01-24T14:00:00-00:00": 13,
    "2012-01-27T09:00:00-00:00": 23
  },
  "participations": [
    {
      "created_at": "2012-01-21T22:00:00-06:00",
      "url": "https://canvas.example.com/path/to/canvas",
    },
  ],
}
```

```
{
  "created_at": "2012-01-27T22:00:00-06:00",
  "url": "https://canvas.example.com/path/to/canvas",
}
```

Get user-in-a-course-level assignment data

GET /api/v1/courses/:course\_id/analytics/users/:student\_id/assignments

Scope: url:GET//api/v1/courses/:course\_id/analytics/users/:student\_id/assignments

Returns a list of assignments for the course sorted by due date. For each assignment returns basic assignment information, the grade breakdown (including the student's actual grade), and the basic submission information for the student's submission if it exists.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/users/<user_id>/assignments \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  {
    "assignment_id": 1234,
    "title": "Assignment 1",
    "points_possible": 10,
    "due_at": "2012-01-25T22:00:00-07:00",
    "unlock_at": "2012-01-20T22:00:00-07:00",
    "muted": false,
    "min_score": 2,
    "max_score": 10,
    "median": 7,
    "first_quartile": 4,
    "third_quartile": 8,
    "module_ids": [
      1,
      2
    ],
    "submission": {
      "posted_at": "2012-01-23T20:00:00-07:00",
      "submitted_at": "2012-01-22T22:00:00-07:00",
      "score": 10
    }
  },
  {
    "assignment_id": 1235,
    "title": "Assignment 2",
    "points_possible": 15,
    "due_at": "2012-01-26T22:00:00-07:00",
    "unlock_at": null,
    "muted": true,
    "min_score": 8,
    "max_score": 8,
    "median": 8,
    "first_quartile": 8,
    "third_quartile": 8,
    "module_ids": [
      1
    ],
    "submission": {
      "posted_at": null,
      "submitted_at": "2012-01-22T22:00:00-07:00"
    }
  }
}
```

Get user-in-a-course-level messaging data

GET /api/v1/courses/:course\_id/analytics/users/:student\_id/communication

Scope: url:GET//api/v1/courses/:course\_id/analytics/users/:student\_id/communication

Returns messaging “hits” grouped by day through the entire history of the course. Returns a hash containing the number of instructor-to-student messages, and student-to-instructor messages, where the hash keys are dates in the format “YYYY-MM-DD”. Message hits include Conversation messages and comments on homework submissions.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/analytics/users/<user_id>/communication \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "2012-01-24":{
    "instructorMessages":1,
    "studentMessages":2
  },
  "2012-01-27":{
    "studentMessages":1
  }
}
```

List external feeds ExternalFeedsController#index

GET /api/v1/courses/:course\_id/external\_feeds

Scope: url:GET//api/v1/courses/:course\_id/external\_feeds

GET /api/v1/groups/:group\_id/external\_feeds

Scope: url:GET//api/v1/groups/:group\_id/external\_feeds

Returns the paginated list of External Feeds this course or group.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/external_feeds \
-H 'Authorization: Bearer <token>'
```

Returns a list of [ExternalFeed](#) objects

Create an external feed ExternalFeedsController#create

POST /api/v1/courses/:course\_id/external\_feeds

Scope: url:POST//api/v1/courses/:course\_id/external\_feeds

POST /api/v1/groups/:group\_id/external\_feeds

Scope: url:POST//api/v1/groups/:group\_id/external\_feeds

Create a new external feed for the course or group.

Request Parameters:

Parameter	Type	Description
url	Required string	The url to the external rss or atom feed
header_match	boolean	If given, only feed entries that contain this string in their title will be imported
verbosity	string	Defaults to “full” Allowed values: full, truncate, link_only

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/external_feeds \
-F url=http://example.com/rss.xml \
-F header_match='news flash!' \
-F verbosity='full' \
-H 'Authorization: Bearer <token>'
```

Returns an [ExternalFeed](#) object

Delete an external feed ExternalFeedsController#destroy

DELETE /api/v1/courses/:course\_id/external\_feeds/:external\_feed\_id

Scope: url:DELETE /api/v1/courses/:course\_id/external\_feeds/:external\_feed\_id

DELETE /api/v1/groups/:group\_id/external\_feeds/:external\_feed\_id

Scope: url:DELETE /api/v1/groups/:group\_id/external\_feeds/:external\_feed\_id

Deletes the external feed.

Example Request:

curl -X DELETE https://<canvas>/api/v1/courses/<course\_id>/external\_feeds/<feed\_id> \
-H 'Authorization: Bearer <token>'

Returns an ExternalFeed object

List announcements AnnouncementsApiController#index

GET /api/v1/announcements

Scope: url:GET /api/v1/announcements

Returns the paginated list of announcements for the given courses and date range. Note that a context\_code field is added to the responses so you can tell which course each announcement belongs to.

Request Parameters:

Parameter	Type	Description
context_codes[]	Required string	List of context_codes to retrieve announcements for (for example, course_123). Only courses are presently supported. The call will fail unless the caller has View Announcements permission in all listed courses.
start_date	Date	Only return announcements posted since the start_date (inclusive). Defaults to 14 days ago. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
end_date	Date	Only return announcements posted before the end_date (inclusive). Defaults to 28 days from start_date. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ. Announcements scheduled for future posting will only be returned to course administrators.
available_after	Date	Only return announcements having locked_at nil or after available_after (exclusive). The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ. Effective only for students (who don't have moderate forum right).
active_only	boolean	Only return active announcements that have been published. Applies only to requesting users that have permission to view unpublished items. Defaults to false for users with access to view unpublished items, otherwise true and unmodifiable.
latest_only	boolean	Only return the latest announcement for each associated context. The response will include at most one announcement for each specified context in the context_codes[] parameter. Defaults to false.
include	array	Optional list of resources to include with the response. May include a string of the name of the resource. Possible values are: "sections", "sections_user_count" if "sections" is passed, includes the course sections that are associated with the topic, if the topic is specific to certain sections of the course. If "sections_user_count" is passed, then: (a) If sections were asked for "and" the topic is specific to certain course sections sections, includes the number of users in each section. (as part of the section json asked for above) (b) Else, includes at the root level the total number of users in the topic's context (group or course) that the topic applies to.

Example Request:

curl https://<canvas>/api/v1/announcements?context\_codes[]=course\_1&context\_codes[]=course\_2 \
-H 'Authorization: Bearer <token>'

Example Response:

```
{
  "id": 1,
  "title": "Hear ye",
  "message": "Henceforth, all assignments must be...",
  "posted_at": "2017-01-31T22:00:00Z",
  "delayed_post_at": null,
  "context_code": "course_2",
  ...
}
```

Returns a list of DiscussionTopic objects

List scopes ScopesApiController#index

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

GET /api/v1/accounts/:account\_id/scopes

Scope: url:GET /api/v1/accounts/:account\_id/scopes

A list of scopes that can be applied to developer keys and access tokens.

Request Parameters:

Parameter	Type	Description
group_by	string	The attribute to group the scopes by. By default no grouping is done. Allowed values: resource_name

Returns a list of Scope objects

List appointment groups AppointmentGroupsController#index

GET /api/v1/appointment\_groups

Scope: url:GET /api/v1/appointment\_groups

Retrieve the paginated list of appointment groups that can be reserved or managed by the current user.

Request Parameters:

Parameter	Type	Description
scope	string	Defaults to "reservable" Allowed values: reservable, manageable
context_codes[]	string	Array of context codes used to limit returned results.
include_past_appointments	boolean	Defaults to false. If true, includes past appointment groups
include[]	string	Array of additional information to include. "appointments" calendar event time slots for this appointment group "child_events" reservations of those time slots "participant_count" number of reservations

Parameter	Type	Description
		“reserved_times”
		the event id, start time and end time of reservations the current user has made)
		“all_context_codes”
		all context codes associated with this appointment group
		Allowed values: appointments, child_events, participant_count, reserved_times, all_context_codes

Create an appointment group AppointmentGroupsController#create

POST /api/v1/appointment\_groups

Scope: url:POST /api/v1/appointment\_groups

Create and return a new appointment group. If new\_appointments are specified, the response will return a new\_appointments array (same format as appointments array, see “List appointment groups” action)

Request Parameters:

Parameter	Type	Description
appointment_group[context_codes][]	Required string	Array of context codes (courses, e.g. course_1) this group should be linked to (1 or more). Users in the course(s) with appropriate permissions will be able to sign up for this appointment group.
appointment_group[sub_context_codes][]	string	Array of sub context codes (course sections or a single group category) this group should be linked to. Used to limit the appointment group to particular sections. If a group category is specified, students will sign up in groups and the participant_type will be “Group” instead of “User”.
appointment_group[title]	Required string	Short title for the appointment group.
appointment_group[description]	string	Longer text description of the appointment group.
appointment_group[location_name]	string	Location name of the appointment group.
appointment_group[location_address]	string	Location address.
appointment_group[publish]	boolean	Indicates whether this appointment group should be published (i.e. made available for signup). Once published, an appointment group cannot be unpublished. Defaults to false.
appointment_group[participants_per_appointment]	integer	Maximum number of participants that may register for each time slot. Defaults to null (no limit).
appointment_group[min_appointments_per_participant]	integer	Minimum number of time slots a user must register for. If not set, users do not need to sign up for any time slots.
appointment_group[max_appointments_per_participant]	integer	Maximum number of time slots a user may register for.
appointment_group[new_appointments][X][]	string	Nested array of start time/end time pairs indicating time slots for this appointment group. Refer to the example request.
		“private”
		participants cannot see who has signed up for a particular time slot
appointment_group[participant_visibility]	string	“protected”
		participants can see who has signed up. Defaults to “private”.
		Allowed values: private, protected
appointment_group[allow_observer_signup]	boolean	Whether observer users can sign-up for an appointment. Defaults to false.

Example Request:

```
curl 'https://<canvas>/api/v1/appointment_groups.json' \  
-X POST \  
-F 'appointment_group[context_codes][]=<course_123>' \  
-F 'appointment_group[sub_context_codes][]=<course_section_234>' \  
-F 'appointment_group[title]=Final Presentation' \  
-F 'appointment_group[participants_per_appointment]=1' \  
-F 'appointment_group[min_appointments_per_participant]=1' \  
-F 'appointment_group[max_appointments_per_participant]=1' \  
-F 'appointment_group[new_appointments][0][]=2012-07-19T21:00:00Z' \  
-F 'appointment_group[new_appointments][0][]=2012-07-19T22:00:00Z' \  
-F 'appointment_group[new_appointments][1][]=2012-07-19T22:00:00Z' \  
-F 'appointment_group[new_appointments][1][]=2012-07-19T23:00:00Z' \  
-H "Authorization: Bearer <token>"
```

Get a single appointment group AppointmentGroupsController#show

GET /api/v1/appointment\_groups/:id

Scope: url:GET /api/v1/appointment\_groups/:id

Returns information for a single appointment group

Request Parameters:

Parameter	Type	Description
		Array of additional information to include. See include[] argument of “List appointment groups” action.
		“child_events”
		reservations of time slots time slots
include[]	string	“appointments”
		will always be returned
		“all_context_codes”
		all context codes associated with this appointment group
		Allowed values: child_events, appointments, all_context_codes

Update an appointment group AppointmentGroupsController#update

PUT /api/v1/appointment\_groups/:id

Scope: url:PUT /api/v1/appointment\_groups/:id

Update and return an appointment group. If new\_appointments are specified, the response will return a new\_appointments array (same format as appointments array, see “List appointment groups” action).

Request Parameters:

Parameter	Type	Description
appointment_group[context_codes][]	Required string	Array of context codes (courses, e.g. course_1) this group should be linked to (1 or more). Users in the course(s) with appropriate permissions will be able to sign up for this appointment group.
appointment_group[sub_context_codes][]	string	Array of sub context codes (course sections or a single group category) this group should be linked to. Used to limit the appointment group to particular sections. If a group category is specified, students will sign up in groups and the participant_type will be “Group” instead of “User”.



Parameter	Type	Description
appointment_group[title]	string	Short title for the appointment group.
appointment_group[description]	string	Longer text description of the appointment group.
appointment_group[location_name]	string	Location name of the appointment group.
appointment_group[location_address]	string	Location address.
appointment_group[publish]	boolean	Indicates whether this appointment group should be published (i.e. made available for signup). Once published, an appointment group cannot be unpublished. Defaults to false.
appointment_group[participants_per_appointment]	integer	Maximum number of participants that may register for each time slot. Defaults to null (no limit).
appointment_group[min_appointments_per_participant]	integer	Minimum number of time slots a user must register for. If not set, users do not need to sign up for any time slots.
appointment_group[max_appointments_per_participant]	integer	Maximum number of time slots a user may register for.
appointment_group[new_appointments][X][]	string	Nested array of start time/end time pairs indicating time slots for this appointment group. Refer to the example request.
	“private”	
	participants cannot see who has signed up for a particular time slot	
appointment_group[participant_visibility]	string	“protected”
	participants can see who has signed up. Defaults to “private”.	
	Allowed values: private, protected	
appointment_group[allow_observer_signup]	boolean	Whether observer users can sign-up for an appointment.

Example Request:

```
curl 'https://<canvas>/api/v1/appointment_groups/543.json' \  
-X PUT \  
-F 'appointment_group[publish]=1' \  
-H 'Authorization: Bearer <token>'
```

Delete an appointment group AppointmentGroupsController#destroy

DELETE /api/v1/appointment\_groups/:id

Scope: url:DELETE|/api/v1/appointment\_groups/:id

Delete an appointment group (and associated time slots and reservations) and return the deleted group

Request Parameters:

Parameter	Type	Description
cancel_reason	string	Reason for deleting/canceling the appointment group.

Example Request:

```
curl 'https://<canvas>/api/v1/appointment_groups/543.json' \  
-X DELETE \  
-F 'cancel_reason=El Tigre Chino got fired' \  
-H 'Authorization: Bearer <token>'
```

List user participants AppointmentGroupsController#users

GET /api/v1/appointment\_groups/:id/users

Scope: url:GET|/api/v1/appointment\_groups/:id/users

A paginated list of users that are (or may be) participating in this appointment group. Refer to the Users API for the response fields. Returns no results for appointment groups with the “Group” participant\_type.

Request Parameters:

Parameter	Type	Description
registration_status	string	Limits results to the a given participation status, defaults to “all” Allowed values: all, registered, registered

List student group participants AppointmentGroupsController#groups

GET /api/v1/appointment\_groups/:id/groups

Scope: url:GET|/api/v1/appointment\_groups/:id/groups

A paginated list of student groups that are (or may be) participating in this appointment group. Refer to the Groups API for the response fields. Returns no results for appointment groups with the “User” participant\_type.

Request Parameters:

Parameter	Type	Description
registration_status	string	Limits results to the a given participation status, defaults to “all” Allowed values: all, registered, registered

Get next appointment AppointmentGroupsController#next\_appointment

GET /api/v1/appointment\_groups/next\_appointment

Scope: url:GET|/api/v1/appointment\_groups/next\_appointment

Return the next appointment available to sign up for. The appointment is returned in a one-element array. If no future appointments are available, an empty array is returned.

Request Parameters:

Parameter	Type	Description
appointment_group_ids[]	string	List of ids of appointment groups to search.

Returns a list of [CalendarEvent](#) objects

Set extensions for student assignment submissions AssignmentExtensionsController#create

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/extensions

Scope: url:POST|/api/v1/courses/:course\_id/assignments/:assignment\_id/extensions

Responses

- 200 OK if the request was successful

- **403 Forbidden** if you are not allowed to extend assignments for this course
- **400 Bad Request** if any of the extensions are invalid

Request Parameters:

Parameter	Type	Description
assignment_extensions[][user_id]	Required integer	The ID of the user we want to add assignment extensions for.
assignment_extensions[][extra_attempts]	Required integer	Number of times the student is allowed to re-take the assignment over the limit.

Example Request:

```
{
  "assignment_extensions": [{
    "user_id": 3,
    "extra_attempts": 2
  },{
    "user_id": 2,
    "extra_attempts": 2
  }]
}
```

Example Response:

```
{
  "assignment_extensions": [AssignmentExtension]
}
```

List assignment groups [AssignmentGroupsController#index](#)

GET /api/v1/courses/:course\_id/assignment\_groups

**Scope:** url:GET|/api/v1/courses/:course\_id/assignment\_groups

Returns the paginated list of assignment groups for the current context. The returned groups are sorted by their position field.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the group. "discussion_topic", "all_dates", "can_edit", "assignment_visibility" & "submission" are only valid if "assignments" is also included. "score_statistics" requires that the "assignments" and "submission" options are included. The "assignment_visibility" option additionally requires that the Differentiated Assignments course feature be turned on. If "observed_users" is passed along with "assignments" and "submission", submissions for observed users will also be included as an array.  Allowed values: assignments, discussion_topic, all_dates, assignment_visibility, overrides, submission, observed_users, can_edit, score_statistics
assignment_ids[]	string	If "assignments" are included, optionally return only assignments having their ID in this array. This argument may also be passed as a comma separated string.
exclude_assignment_submission_types[]	string	If "assignments" are included, those with the specified submission types will be excluded from the assignment groups.  Allowed values: online_quiz, discussion_topic, wiki_page, external_tool
override_assignment_dates	boolean	Apply assignment overrides for each assignment, defaults to true.
grading_period_id	integer	The id of the grading period in which assignment groups are being requested (Requires grading periods to exist.)
scope_assignments_to_student	boolean	If true, all assignments returned will apply to the current user in the specified grading period. If assignments apply to other students in the specified grading period, but not the current user, they will not be returned. (Requires the grading_period_id argument and grading periods to exist. In addition, the current user must be a student.)

Returns a list of [AssignmentGroup](#) objects

Get an Assignment Group [AssignmentGroupsApiController#show](#)

GET /api/v1/courses/:course\_id/assignment\_groups/:assignment\_group\_id

**Scope:** url:GET|/api/v1/courses/:course\_id/assignment\_groups/:assignment\_group\_id

Returns the assignment group with the given id.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the group. "discussion_topic" and "assignment_visibility" and "submission" are only valid if "assignments" is also included. "score_statistics" is only valid if "submission" and "assignments" are also included. The "assignment_visibility" option additionally requires that the Differentiated Assignments course feature be turned on.  Allowed values: assignments, discussion_topic, assignment_visibility, submission, score_statistics
override_assignment_dates	boolean	Apply assignment overrides for each assignment, defaults to true.
grading_period_id	integer	The id of the grading period in which assignment groups are being requested (Requires grading periods to exist on the account)

Returns an [AssignmentGroup](#) object

Create an Assignment Group [AssignmentGroupsApiController#create](#)

POST /api/v1/courses/:course\_id/assignment\_groups

**Scope:** url:POST|/api/v1/courses/:course\_id/assignment\_groups

Create a new assignment group for this course.

Request Parameters:

Parameter	Type	Description
name	string	The assignment group's name
position	integer	The position of this assignment group in relation to the other assignment groups
group_weight	number	The percent of the total grade that this assignment group represents
sis_source_id	string	The sis source id of the Assignment Group
integration_data	Object	The integration data of the Assignment Group

Returns an [AssignmentGroup](#) object

Edit an Assignment Group [AssignmentGroupsApiController#update](#)

PUT /api/v1/courses/:course\_id/assignment\_groups/:assignment\_group\_id

**Scope:** url:PUT|/api/v1/courses/:course\_id/assignment\_groups/:assignment\_group\_id

Modify an existing Assignment Group.

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
name	string	The assignment group's name
position	integer	The position of this assignment group in relation to the other assignment groups
group_weight	number	The percent of the total grade that this assignment group represents
sis_source_id	string	The sis source id of the Assignment Group
integration_data	Object	The integration data of the Assignment Group
rules	string	The grading rules that are applied within this assignment group See the Assignment Group object definition for format

Returns an [AssignmentGroup](#) object

**[Destroy an Assignment Group AssignmentGroupsApiController#destroy](#)**

**DELETE** /api/v1/courses/:course\_id/assignment\_groups:assignment\_group\_id

**Scope:** url:DELETE|/api/v1/courses/:course\_id/assignment\_groups/:assignment\_group\_id

Deletes the assignment group with the given id.

**Request Parameters:**

Parameter	Type	Description
move_assignments_to	integer	The ID of an active Assignment Group to which the assignments that are currently assigned to the destroyed Assignment Group will be assigned. NOTE: If this argument is not provided, any assignments in this Assignment Group will be deleted.

Returns an [AssignmentGroup](#) object

**[Delete an assignment AssignmentsController#destroy](#)**

**DELETE** /api/v1/courses/:course\_id/assignments/:id

**Scope:** url:DELETE|/api/v1/courses/:course\_id/assignments/:id

Delete the given assignment.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns an [Assignment](#) object

**[List assignments AssignmentsApiController#index](#)**

**GET** /api/v1/courses/:course\_id/assignments

**Scope:** url:GET|/api/v1/courses/:course\_id/assignments

**GET** /api/v1/courses/:course\_id/assignment\_groups:assignment\_group\_id/assignments

**Scope:** url:GET|/api/v1/courses/:course\_id/assignment\_groups:assignment\_group\_id/assignments

Returns the paginated list of assignments for the current course or assignment group.

**Request Parameters:**

Parameter	Type	Description
include[]	string	Optional information to include with each assignment:
		submission
		The current user's current Submission
		assignment_visibility
		An array of ids of students who can see the assignment
		all_dates
		An array of AssignmentDate structures, one for each override, and also a base if the assignment has an "Everyone" / "Everyone Else" date
		overrides
		An array of AssignmentOverride structures
		observed_users
		An array of submissions for observed users
		can_edit
		an extra Boolean value will be included with each Assignment (and AssignmentDate if all_dates is supplied) to indicate whether the caller can edit the assignment or date. Moderated grading and closed grading periods may restrict a user's ability to edit an assignment.
		score_statistics
		An object containing min, max, and mean score on this assignment. This will not be included for students if there are less than 5 graded assignments or if disabled by the instructor. Only valid if 'submission' is also included.
search_term	string	ab_guid
		An array of guid strings for academic benchmarks
		Allowed values: submission, assignment_visibility, all_dates, overrides, observed_users, can_edit, score_statistics, ab_guid
override_assignment_dates	boolean	The partial title of the assignments to match and return.
needs_grading_count_by_section	boolean	Apply assignment overrides for each assignment, defaults to true.
bucket	string	Split up "needs_grading_count" by sections into the "needs_grading_count_by_section" key, defaults to false
assignment_ids[]	string	If included, only return certain assignments depending on due date and submission status.
		Allowed values: past, overdue, undated, ungraded, unsubmitted, upcoming, future
order_by	string	if set, return only assignments specified
post_to_sis	boolean	Determines the order of the assignments. Defaults to "position".
		Allowed values: position, name, due_at
new_quizzes	boolean	Return only assignments that have post_to_sis set or not set.
	boolean	Return only New Quizzes assignments

Returns a list of [Assignment](#) objects

[List assignments for user AssignmentsApiController#user\\_index](#)

GET /api/v1/users/:user\_id/courses/:course\_id/assignments

Scope: url:GET/api/v1/users/:user\_id/courses/:course\_id/assignments

Returns the paginated list of assignments for the specified user if the current user has rights to view. See [List assignments](#) for valid arguments.

[Duplicate assignment AssignmentsApiController#duplicate](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/duplicate

Scope: url:POST/api/v1/courses/:course\_id/assignments/:assignment\_id/duplicate

Duplicate an assignment and return a json based on result\_type argument.

Request Parameters:

Parameter	Type	Description
result_type	string	Optional information: When the root account has the feature 'newquizzes_on_quiz_page' enabled and this argument is set to "Quiz" the response will be serialized into a quiz format( <a href="#">quizzes</a> ); When this argument isn't specified the response will be serialized into an assignment format;  Allowed values: Quiz

Example Request:

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/assignments/123/duplicate
```

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/assignments/123/duplicate?result_type=Quiz
```

Returns an [Assignment](#) object

[List group members for a student on an assignment AssignmentsApiController#student\\_group\\_members](#)

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/users/:user\_id/group\_members

Scope: url:GET/api/v1/courses/:course\_id/assignments/:assignment\_id/users/:user\_id/group\_members

Returns student ids and names for the group.

Example Request:

```
curl https://<canvas>/api/v1/courses/1/assignments/1/users/1/group_members
```

Returns a list of [BasicUser](#) objects

[Get a single assignment AssignmentsApiController#show](#)

GET /api/v1/courses/:course\_id/assignments/:id

Scope: url:GET/api/v1/courses/:course\_id/assignments/:id

Returns the assignment with the given id.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the assignment. The "assignment_visibility" option requires that the Differentiated Assignments course feature be turned on. If "observed_users" is passed, submissions for observed users will also be included. For "score_statistics" to be included, the "submission" option must also be set.  Allowed values: submission, assignment_visibility, overrides, observed_users, can_edit, score_statistics, ab_guid
override_assignment_dates	boolean	Apply assignment overrides to the assignment, defaults to true.
needs_grading_count_by_section	boolean	Split up "needs_grading_count" by sections into the "needs_grading_count_by_section" key, defaults to false
all_dates	boolean	All dates associated with the assignment, if applicable

Returns an [Assignment](#) object

[Create an assignment AssignmentsApiController#create](#)

POST /api/v1/courses/:course\_id/assignments

Scope: url:POST/api/v1/courses/:course\_id/assignments

Create a new assignment for this course. The assignment is created in the active state.

Request Parameters:

Parameter	Type	Description
assignment[name]	Required string	The assignment name.
assignment[position]	integer	The position of this assignment in the group when displaying assignment lists.
assignment[submission_types][]	string	List of supported submission types for the assignment. Unless the assignment is allowing online submissions, the array should only have one element.  If not allowing online submissions, your options are:  "online_quiz" "none" "on_paper" "discussion_topic" "external_tool"  If you are allowing online submissions, you can have one or many allowed submission types:  "online_upload" "online_text_entry" "online_url" "media_recording" (Only valid when the Kaltura plugin is enabled) "student_annotation"  Allowed values: online_quiz, none, on_paper, discussion_topic, external_tool, online_upload, online_text_entry, online_url, media_recording, student_annotation
assignment[allowed_extensions][]	string	Allowed extensions if submission_types includes "online_upload"  Example:  allowed_extensions: ["docx","ppt"]
assignment[turnitin_enabled]	boolean	Only applies when the Turnitin plugin is enabled for a course and the submission_types array includes "online_upload". Toggles Turnitin submissions for the assignment. Will be ignored if Turnitin is not available for the course.
assignment[vericite_enabled]	boolean	Only applies when the VeriCite plugin is enabled for a course and the submission_types array includes "online_upload". Toggles VeriCite submissions for the assignment. Will be ignored if VeriCite is not available for the course.
assignment[turnitin_settings]	string	Settings to send along to turnitin. See Assignment object definition for format.

Parameter	Type	Description
assignment[integration_data]	string	Data used for SIS integrations. Requires admin-level token with the "Manage SIS" permission. JSON string required.
assignment[integration_id]	string	Unique ID from third party integrations
assignment[peer_reviews]	boolean	If submission_types does not include external_tool,discussion_topic, online_quiz, or on_paper, determines whether or not peer reviews will be turned on for the assignment.
assignment[automatic_peer_reviews]	boolean	Whether peer reviews will be assigned automatically by Canvas or if teachers must manually assign peer reviews. Does not apply if peer reviews are not enabled.
assignment[notify_of_update]	boolean	If true, Canvas will send a notification to students in the class notifying them that the content has changed.
assignment[group_category_id]	integer	If present, the assignment will become a group assignment assigned to the group.
assignment[grade_group_students_individually]	integer	If this is a group assignment, teachers have the options to grade students individually. If false, Canvas will apply the assignment's score to each member of the group. If true, the teacher can manually assign scores to each member of the group.
assignment[external_tool_tag_attributes]	string	Hash of external tool parameters if submission_types is ["external_tool"]. See Assignment object definition for format.
assignment[points_possible]	number	The maximum points possible on the assignment.
assignment[grading_type]	string	The strategy used for grading the assignment. The assignment defaults to "points" if this field is omitted. Allowed values: pass_fail, percent, letter_grade, gpa_scale, points, not_graded
assignment[due_at]	DateTime	The day/time the assignment is due. Must be between the lock dates if there are lock dates. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[lock_at]	DateTime	The day/time the assignment is locked after. Must be after the due date if there is a due date. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[unlock_at]	DateTime	The day/time the assignment is unlocked. Must be before the due date if there is a due date. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[description]	string	The assignment's description, supports HTML.
assignment[assignment_group_id]	integer	The assignment group id to put the assignment in. Defaults to the top assignment group in the course.
assignment[assignment_overrides][]	AssignmentOverride	List of overrides for the assignment.
assignment[only_visible_to_overrides]	boolean	Whether this assignment is only visible to overrides (Only useful if 'differentiated assignments' account setting is on)
assignment[published]	boolean	Whether this assignment is published. (Only useful if 'draft state' account setting is on) Unpublished assignments are not visible to students.
assignment[grading_standard_id]	integer	The grading standard id to set for the course. If no value is provided for this argument the current grading_standard will be un-set from this course. This will update the grading_type for the course to 'letter_grade' unless it is already 'gpa_scale'.
assignment[omit_from_final_grade]	boolean	Whether this assignment is counted towards a student's final grade.
assignment[hide_in_gradebook]	boolean	Whether this assignment is shown in the gradebook.
assignment[quiz_lti]	boolean	Whether this assignment should use the Quizzes 2 LTI tool. Sets the submission type to 'external_tool' and configures the external tool attributes to use the Quizzes 2 LTI tool configured for this course. Has no effect if no Quizzes 2 LTI tool is configured.
assignment[moderated_grading]	boolean	Whether this assignment is moderated.
assignment[grader_count]	integer	The maximum number of provisional graders who may issue grades for this assignment. Only relevant for moderated assignments. Must be a positive value, and must be set to 1 if the course has fewer than two active instructors. Otherwise, the maximum value is the number of active instructors in the course minus one, or 10 if the course has more than 11 active instructors.
assignment[final_grader_id]	integer	The user ID of the grader responsible for choosing final grades for this assignment. Only relevant for moderated assignments.
assignment[grader_comments_visible_to_graders]	boolean	Boolean indicating if provisional graders' comments are visible to other provisional graders. Only relevant for moderated assignments.
assignment[graders_anonymous_to_graders]	boolean	Boolean indicating if provisional graders' identities are hidden from other provisional graders. Only relevant for moderated assignments.
assignment[graders_names_visible_to_final_grader]	boolean	Boolean indicating if provisional grader identities are visible to the the final grader. Only relevant for moderated assignments.
assignment[anonymous_grading]	boolean	Boolean indicating if the assignment is graded anonymously. If true, graders cannot see student identities.
assignment[allowed_attempts]	integer	The number of submission attempts allowed for this assignment. Set to -1 for unlimited attempts.
assignment[annotatable_attachment_id]	integer	The Attachment ID of the document being annotated. Only applies when submission_types includes "student_annotation".

Returns an [Assignment](#) object

[Edit an assignment AssignmentsApiController#update](#)

PUT /api/v1/courses/:course\_id/assignments/:id

Scope: url:PUT|/api/v1/courses/:course\_id/assignments/:id

Modify an existing assignment.

Request Parameters:

Parameter	Type	Description
assignment[name]	string	The assignment name.
assignment[position]	integer	The position of this assignment in the group when displaying assignment lists.
assignment[submission_types][]	string	Only applies if the assignment doesn't have student submissions.  List of supported submission types for the assignment. Unless the assignment is allowing online submissions, the array should only have one element.  If not allowing online submissions, your options are:  "online_quiz" "none" "on_paper" "discussion_topic" "external_tool"  If you are allowing online submissions, you can have one or many allowed submission types:  "online_upload"

Parameter	Type	Description
		<div>"online_text_entry" "online_url" "media_recording" (Only valid when the Kaltura plugin is enabled) "student_annotation"</div> <div>Allowed values: online_quiz, none, on_paper, discussion_topic, external_tool, online_upload, online_text_entry, online_url, media_recording, student_annotation</div>
assignment[allowed_extensions[]]	string	<div>Allowed extensions if submission_types includes "online_upload"</div> <div>Example:  allowed_extensions: ["docx","ppt"]</div>
assignment[turnitin_enabled]	boolean	Only applies when the Turnitin plugin is enabled for a course and the submission_types array includes "online_upload". Toggles Turnitin submissions for the assignment. Will be ignored if Turnitin is not available for the course.
assignment[vericite_enabled]	boolean	Only applies when the VeriCite plugin is enabled for a course and the submission_types array includes "online_upload". Toggles VeriCite submissions for the assignment. Will be ignored if VeriCite is not available for the course.
assignment[turnitin_settings]	string	Settings to send along to turnitin. See Assignment object definition for format.
assignment[sis_assignment_id]	string	The sis id of the Assignment
assignment[integration_data]	string	Data used for SIS integrations. Requires admin-level token with the "Manage SIS" permission. JSON string required.
assignment[integration_id]	string	Unique ID from third party integrations
assignment[peer_reviews]	boolean	If submission_types does not include external_tool,discussion_topic, online_quiz, or on_paper, determines whether or not peer reviews will be turned on for the assignment.
assignment[automatic_peer_reviews]	boolean	Whether peer reviews will be assigned automatically by Canvas or if teachers must manually assign peer reviews. Does not apply if peer reviews are not enabled.
assignment[notify_of_update]	boolean	If true, Canvas will send a notification to students in the class notifying them that the content has changed.
assignment[group_category_id]	integer	If present, the assignment will become a group assignment assigned to the group.
assignment[grade_group_students_individually]	integer	If this is a group assignment, teachers have the options to grade students individually. If false, Canvas will apply the assignment's score to each member of the group. If true, the teacher can manually assign scores to each member of the group.
assignment[external_tool_tag_attributes]	string	Hash of external tool parameters if submission_types is ["external_tool"]. See Assignment object definition for format.
assignment[points_possible]	number	The maximum points possible on the assignment.
assignment[grading_type]	string	<div>The strategy used for grading the assignment. The assignment defaults to "points" if this field is omitted.</div> <div>Allowed values: pass_fail, percent, letter_grade, gpa_scale, points, not_graded</div>
assignment[due_at]	DateTime	The day/time the assignment is due. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[lock_at]	DateTime	The day/time the assignment is locked after. Must be after the due date if there is a due date. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[unlock_at]	DateTime	The day/time the assignment is unlocked. Must be before the due date if there is a due date. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z.
assignment[description]	string	The assignment's description, supports HTML.
assignment[assignment_group_id]	integer	The assignment group id to put the assignment in. Defaults to the top assignment group in the course.
assignment[assignment_overrides[]]	AssignmentOverride	List of overrides for the assignment. If the assignment[assignment_overrides] key is absent, any existing overrides are kept as is. If the assignment[assignment_overrides] key is present, existing overrides are updated or deleted (and new ones created, as necessary) to match the provided list.
assignment[only_visible_to_overrides]	boolean	Whether this assignment is only visible to overrides (Only useful if 'differentiated assignments' account setting is on)
assignment[published]	boolean	Whether this assignment is published. (Only useful if 'draft state' account setting is on) Unpublished assignments are not visible to students.
assignment[grading_standard_id]	integer	The grading standard id to set for the course. If no value is provided for this argument the current grading_standard will be un-set from this course. This will update the grading_type for the course to 'letter_grade' unless it is already 'gpa_scale'.
assignment[omit_from_final_grade]	boolean	Whether this assignment is counted towards a student's final grade.
assignment[hide_in_gradebook]	boolean	Whether this assignment is shown in the gradebook.
assignment[moderated_grading]	boolean	Whether this assignment is moderated.
assignment[grader_count]	integer	The maximum number of provisional graders who may issue grades for this assignment. Only relevant for moderated assignments. Must be a positive value, and must be set to 1 if the course has fewer than two active instructors. Otherwise, the maximum value is the number of active instructors in the course minus one, or 10 if the course has more than 11 active instructors.
assignment[final_grader_id]	integer	The user ID of the grader responsible for choosing final grades for this assignment. Only relevant for moderated assignments.
assignment[grader_comments_visible_to_graders]	boolean	Boolean indicating if provisional graders' comments are visible to other provisional graders. Only relevant for moderated assignments.
assignment[graders_anonymous_to_graders]	boolean	Boolean indicating if provisional graders' identities are hidden from other provisional graders. Only relevant for moderated assignments.
assignment[graders_names_visible_to_final_grader]	boolean	Boolean indicating if provisional grader identities are visible to the the final grader. Only relevant for moderated assignments.
assignment[anonymous_grading]	boolean	Boolean indicating if the assignment is graded anonymously. If true, graders cannot see student identities.
assignment[allowed_attempts]	integer	The number of submission attempts allowed for this assignment. Set to -1 or null for unlimited attempts.
assignment[annotatable_attachment_id]	integer	<div>The Attachment ID of the document being annotated.</div> <div>Only applies when submission_types includes "student_annotation".</div>
assignment[force_updated_at]	boolean	If true, updated_at will be set even if no changes were made.
assignment[submission_types[]]	string	<div><b>[DEPRECATED]</b> Effective 2021-05-26 (notice given 2021-02-18)</div> <div>Only applies if the assignment doesn't have student submissions.</div>

Returns an [Assignment](#) object

[Bulk update assignment dates AssignmentsApiController#bulk\\_update](#)

**PUT /api/v1/courses/:course\_id/assignments/bulk\_update**

**Scope:** url:PUT /api/v1/courses/:course\_id/assignments/bulk\_update

Update due dates and availability dates for multiple assignments in a course.

Accepts a JSON array of objects containing two keys each: id, the assignment id, and all\_dates, an array of AssignmentDate structures containing the base and/or override dates for the assignment, as returned from the [List assignments](#) endpoint with include[]=all\_dates.

This endpoint cannot create or destroy assignment overrides; any existing assignment overrides that are not referenced in the arguments will be left alone. If an override is given, any dates that are not supplied with it will be defaulted. To clear a date, specify null explicitly.

All referenced assignments will be validated before any are saved. A list of errors will be returned if any provided dates are invalid, and no changes will be saved.

The bulk update is performed in a background job, use the [Progress API](#) to check its status.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/1/assignments/bulk_update' \
-X PUT \
--data '{
  "id": 1,
  "all_dates": [
    {
      "base": true,
      "due_at": "2020-08-29T23:59:00-06:00"
    },
    {
      "id": 2,
      "due_at": "2020-08-30T23:59:00-06:00"
    }
  ]
}' \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <token>"
```

Returns a [Progress](#) object

[List assignment overrides AssignmentOverridesController#index](#)

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides**

**Scope:** url:GET /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides

Returns the paginated list of overrides for this assignment that target sections/groups/students visible to the current user.

Returns a list of [AssignmentOverride](#) objects

[Get a single assignment override AssignmentOverridesController#show](#)

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id**

**Scope:** url:GET /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id

Returns details of the the override with the given id.

Returns an [AssignmentOverride](#) object

[Redirect to the assignment override for a group AssignmentOverridesController#group\\_alias](#)

**GET /api/v1/groups/:group\_id/assignments/:assignment\_id/override**

**Scope:** url:GET /api/v1/groups/:group\_id/assignments/:assignment\_id/override

Responds with a redirect to the override for the given group, if any (404 otherwise).

[Redirect to the assignment override for a section AssignmentOverridesController#section\\_alias](#)

**GET /api/v1/sections/:course\_section\_id/assignments/:assignment\_id/override**

**Scope:** url:GET /api/v1/sections/:course\_section\_id/assignments/:assignment\_id/override

Responds with a redirect to the override for the given section, if any (404 otherwise).

[Create an assignment override AssignmentOverridesController#create](#)

**POST /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides**

**Scope:** url:POST /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides

One of student\_ids, group\_id, or course\_section\_id must be present. At most one should be present; if multiple are present only the most specific (student\_ids first, then group\_id, then course\_section\_id) is used and any others are ignored.

**Request Parameters:**

Parameter	Type	Description
assignment_override[student_ids[]]	integer	The IDs of the override's target students. If present, the IDs must each identify a user with an active student enrollment in the course that is not already targetted by a different adhoc override.
assignment_override[title]	string	The title of the adhoc assignment override. Required if student_ids is present, ignored otherwise (the title is set to the name of the targetted group or section instead).
		The ID of the override's target group. If present, the following conditions must be met for the override to be successful:
		1. the assignment MUST be a group assignment (a group_category_id is assigned to it)
assignment_override[group_id]	integer	2. the ID must identify an active group in the group set the assignment is in
		3. the ID must not be targetted by a different override
		See <a href="#">Appendix: Group assignments</a> for more info.
assignment_override[course_section_id]	integer	The ID of the override's target section. If present, must identify an active section of the assignment's course not already targetted by a different override.
assignment_override[due_at]	DateTime	The day/time the overridden assignment is due. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect due date. May be present but null to indicate the override removes any previous due date.
assignment_override[unlock_at]	DateTime	The day/time the overridden assignment becomes unlocked. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect the unlock date. May be present but null to indicate the override removes any previous unlock date.
assignment_override[lock_at]	DateTime	The day/time the overridden assignment becomes locked. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect the lock date. May be present but null to indicate the override removes any previous lock date.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/2/overrides.json' \
-X POST \
-F 'assignment_override[student_ids][]=0' \
-F 'assignment_override[title]=Fred Flinstone' \
-F 'assignment_override[due_at]=2012-10-00T21:00:00Z' \
-H "Authorization: Bearer <token>"
```

Returns an [AssignmentOverride](#) object

[Update an assignment override AssignmentOverridesController#update](#)

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id**

**Scope:** url:PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id

All current overridden values must be supplied if they are to be retained; e.g. if due\_at was overridden, but this PUT omits a value for due\_at, due\_at will no longer be overridden. If the override is adhoc and student\_ids is not supplied, the target

override set is unchanged. Target override sets cannot be changed for group or section overrides.

Request Parameters:

Parameter	Type	Description
assignment_override[student_ids][ ]	integer	The IDs of the override's target students. If present, the IDs must each identify a user with an active student enrollment in the course that is not already targetted by a different adhoc override. Ignored unless the override being updated is adhoc.
assignment_override[title]	string	The title of an adhoc assignment override. Ignored unless the override being updated is adhoc.
assignment_override[due_at]	DateTime	The day/time the overridden assignment is due. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect due date. May be present but null to indicate the override removes any previous due date.
assignment_override[unlock_at]	DateTime	The day/time the overridden assignment becomes unlocked. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect the unlock date. May be present but null to indicate the override removes any previous unlock date.
assignment_override[lock_at]	DateTime	The day/time the overridden assignment becomes locked. Accepts times in ISO 8601 format, e.g. 2014-10-21T18:48:00Z. If absent, this override will not affect the lock date. May be present but null to indicate the override removes any previous lock date.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1/assignments/2/overrides/3.json' \  
-X PUT \  
-F 'assignment_override[title]=Fred Flintstone' \  
-F 'assignment_override[due_at]=2012-10-08T21:00:00Z' \  
-H 'Authorization: Bearer <token>'
```

Returns an [AssignmentOverride](#) object

[Delete an assignment override AssignmentOverridesController#destroy](#)

**DELETE /api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id**

**Scope:** url:DELETE//api/v1/courses/:course\_id/assignments/:assignment\_id/overrides/:id

Deletes an override and returns its former details.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1/assignments/2/overrides/3.json' \  
-X DELETE \  
-H 'Authorization: Bearer <token>'
```

Returns an [AssignmentOverride](#) object

[Batch retrieve overrides in a course AssignmentOverridesController#batch\\_retrieve](#)

**GET /api/v1/courses/:course\_id/assignments/overrides**

**Scope:** url:GET//api/v1/courses/:course\_id/assignments/overrides

Returns a list of specified overrides in this course, providing they target sections/groups/students visible to the current user. Returns null elements in the list for requests that were not found.

Request Parameters:

Parameter	Type	Description
assignment_overrides[][id]	Required string	Ids of overrides to retrieve
assignment_overrides[][assignment_id]	Required string	Ids of assignments for each override

Example Request:

```
curl 'https://<canvas>/api/v1/courses/12/assignments/overrides.json?assignment_overrides[][id]=109&assignment_overrides[][assignment_id]=122&assignment_overrides[][id]=99&assignment_overrides[][assignment_id]=111' \  
-H 'Authorization: Bearer <token>'
```

Returns a list of [AssignmentOverride](#) objects

[Batch create overrides in a course AssignmentOverridesController#batch\\_create](#)

**POST /api/v1/courses/:course\_id/assignments/overrides**

**Scope:** url:POST//api/v1/courses/:course\_id/assignments/overrides

Creates the specified overrides for each assignment. Handles creation in a transaction, so all records are created or none are.

One of student\_ids, group\_id, or course\_section\_id must be present. At most one should be present; if multiple are present only the most specific (student\_ids first, then group\_id, then course\_section\_id) is used and any others are ignored.

Errors are reported in an errors attribute, an array of errors corresponding to inputs. Global errors will be reported as a single element errors array

Request Parameters:

Parameter	Type	Description
assignment_overrides[]	Required AssignmentOverride	Attributes for the new assignment overrides. See <a href="#">Create an assignment override</a> for available attributes

Example Request:

```
curl 'https://<canvas>/api/v1/courses/12/assignments/overrides.json' \  
-X POST \  
-F 'assignment_overrides[][assignment_id]=109' \  
-F 'assignment_overrides[][student_ids][]=8' \  
-F 'assignment_overrides[][title]=foo' \  
-F 'assignment_overrides[][assignment_id]=13' \  
-F 'assignment_overrides[][course_section_id]=200' \  
-F 'assignment_overrides[][due_at]=2012-10-08T21:00:00Z' \  
-H 'Authorization: Bearer <token>'
```

Returns a list of [AssignmentOverride](#) objects

[Batch update overrides in a course AssignmentOverridesController#batch\\_update](#)

**PUT /api/v1/courses/:course\_id/assignments/overrides**

**Scope:** url:PUT//api/v1/courses/:course\_id/assignments/overrides

Updates a list of specified overrides for each assignment. Handles overrides in a transaction, so either all updates are applied or none. See [Update an assignment override](#) for available attributes.

All current overridden values must be supplied if they are to be retained; e.g. if due\_at was overridden, but this PUT omits a value for due\_at, due\_at will no longer be overridden. If the override is adhoc and student\_ids is not supplied, the target override set is unchanged. Target override sets cannot be changed for group or section overrides.

Errors are reported in an errors attribute, an array of errors corresponding to inputs. Global errors will be reported as a single element errors array

Request Parameters:

Parameter	Type	Description
assignment_overrides[]	Required AssignmentOverride	Attributes for the updated overrides.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/12/assignments/overrides.json' \  
-X PUT \  
-F 'assignment_overrides[][id]=122' \  
-F 'assignment_overrides[][assignment_id]=109' \  
-F 'assignment_overrides[][title]=foo' \  
-F 'assignment_overrides[][id]=993' \  
-F 'assignment_overrides[][assignment_id]=13' \  
-F 'assignment_overrides[][due_at]=2012-10-08T21:00:00Z' \  
-H 'Authorization: Bearer <token>'
```



```
-H "Authorization: Bearer <token>"
```

Returns a list of [AssignmentOverride](#) objects

[List authentication providers AuthenticationProvidersController#index](#)

**GET /api/v1/accounts/:account\_id/authentication\_providers**

**Scope:** url:GET|/api/v1/accounts/:account\_id/authentication\_providers

Returns a paginated list of authentication providers

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/authentication_providers' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [AuthenticationProvider](#) objects

[Get authentication provider AuthenticationProvidersController#show](#)

**GET /api/v1/accounts/:account\_id/authentication\_providers/:id**

**Scope:** url:GET|/api/v1/accounts/:account\_id/authentication\_providers/:id

Get the specified authentication provider

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/authentication_providers/<id>' \
-H 'Authorization: Bearer <token>'
```

Returns an [AuthenticationProvider](#) object

[Update authentication provider AuthenticationProvidersController#update](#)

**PUT /api/v1/accounts/:account\_id/authentication\_providers/:id**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/authentication\_providers/:id

Update an authentication provider using the same options as the create endpoint. You can not update an existing provider to a new authentication type.

**Example Request:**

```
# update SAML config
curl -X PUT 'https://<canvas>/api/v1/accounts/<account_id>/authentication_providers/<id>' \
-F 'idp_entity_id=<new_idp_entity_id>' \
-F 'log_in_url=<new_url>' \
-H 'Authorization: Bearer <token>'
```

Returns an [AuthenticationProvider](#) object

[Delete authentication provider AuthenticationProvidersController#destroy](#)

**DELETE /api/v1/accounts/:account\_id/authentication\_providers/:id**

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/authentication\_providers/:id

Delete the config

**Example Request:**

```
curl -X DELETE 'https://<canvas>/api/v1/accounts/<account_id>/authentication_providers/<id>' \
-H 'Authorization: Bearer <token>'
```

[Restore a deleted authentication provider AuthenticationProvidersController#restore](#)

**PUT /api/v1/accounts/:account\_id/authentication\_providers/:id/restore**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/authentication\_providers/:id/restore

Restore an authentication provider back to active that was previously deleted. Only available to admins who can manage\_account\_settings for given root account.

**Example Request:**

```
curl -X PUT 'https://<canvas>/api/v1/accounts/<account_id>/authentication_providers/<id>/restore' \
-H 'Authorization: Bearer <token>'
```

Returns an [AuthenticationProvider](#) object

[show account auth settings AuthenticationProvidersController#show\\_sso\\_settings](#)

**GET /api/v1/accounts/:account\_id/sso\_settings**

**Scope:** url:GET|/api/v1/accounts/:account\_id/sso\_settings

The way to get the current state of each account level setting that's relevant to Single Sign On configuration

You can list the current state of each setting with "update\_sso\_settings"

**Example Request:**

```
curl -XGET 'https://<canvas>/api/v1/accounts/<account_id>/sso_settings' \
-H 'Authorization: Bearer <token>'
```

Returns a [SSOSettings](#) object

[update account auth settings AuthenticationProvidersController#update\\_sso\\_settings](#)

**PUT /api/v1/accounts/:account\_id/sso\_settings**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/sso\_settings

For various cases of mixed SSO configurations, you may need to set some configuration at the account level to handle the particulars of your setup.

This endpoint accepts a PUT request to set several possible account settings. All setting are optional on each request, any that are not provided at all are simply retained as is. Any that provide the key but a null-ish value (blank string, null, undefined) will be UN-set.

You can list the current state of each setting with "show\_sso\_settings"

**Example Request:**

```
curl -XPUT 'https://<canvas>/api/v1/accounts/<account_id>/sso_settings' \
-F 'sso_settings[auth_discovery_url]=<new_url>' \
-F 'sso_settings[change_password_url]=<new_url>' \
-F 'sso_settings[login_handle_name]=<new_handle>' \
-H 'Authorization: Bearer <token>'
```

Returns a [SSOSettings](#) object

[Query by login. AuthenticationAuditApiController#for\\_login](#)

**GET /api/v1/audit/authentication/logins/:login\_id**

**Scope:** url:GET|/api/v1/audit/authentication/logins/:login\_id

List authentication events for a given login.

**Request Parameters:**

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events. Events are stored for one year.

Parameter	Type	Description
end_time	DateTime	The end of the time range from which you want events.

[Query by account. AuthenticationAuditApiController#for\\_account](#)

**GET /api/v1/audit/authentication/accounts/:account\_id**

**Scope:** url:GET|/api/v1/audit/authentication/accounts/:account\_id

List authentication events for a given account.

**Request Parameters:**

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events. Events are stored for one year.
end_time	DateTime	The end of the time range from which you want events.

[Query by user. AuthenticationAuditApiController#for\\_user](#)

**GET /api/v1/audit/authentication/users/:user\_id**

**Scope:** url:GET|/api/v1/audit/authentication/users/:user\_id

List authentication events for a given user.

**Request Parameters:**

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events. Events are stored for one year.
end_time	DateTime	The end of the time range from which you want events.

[List blackout dates BlackoutDatesController#index](#)

**GET /api/v1/courses/:course\_id/blackout\_dates**

**Scope:** url:GET|/api/v1/courses/:course\_id/blackout\_dates

**GET /api/v1/accounts/:account\_id/blackout\_dates**

**Scope:** url:GET|/api/v1/accounts/:account\_id/blackout\_dates

Returns the list of blackout dates for the current context.

Returns a list of [BlackoutDate](#) objects

[Get a single blackout date BlackoutDatesController#show](#)

**GET /api/v1/courses/:course\_id/blackout\_dates/:id**

**Scope:** url:GET|/api/v1/courses/:course\_id/blackout\_dates/:id

**GET /api/v1/accounts/:account\_id/blackout\_dates/:id**

**Scope:** url:GET|/api/v1/accounts/:account\_id/blackout\_dates/:id

Returns the blackout date with the given id.

Returns a [BlackoutDate](#) object

[New Blackout Date BlackoutDatesController#new](#)

**GET /api/v1/courses/:course\_id/blackout\_dates/new**

**Scope:** url:GET|/api/v1/courses/:course\_id/blackout\_dates/new

**GET /api/v1/accounts/:account\_id/blackout\_dates/new**

**Scope:** url:GET|/api/v1/accounts/:account\_id/blackout\_dates/new

Initialize an unsaved Blackout Date for the given context.

Returns a [BlackoutDate](#) object

[Create Blackout Date BlackoutDatesController#create](#)

**POST /api/v1/courses/:course\_id/blackout\_dates**

**Scope:** url:POST|/api/v1/courses/:course\_id/blackout\_dates

**POST /api/v1/accounts/:account\_id/blackout\_dates**

**Scope:** url:POST|/api/v1/accounts/:account\_id/blackout\_dates

Create a blackout date for the given context.

**Request Parameters:**

Parameter	Type	Description
start_date	Date	The start date of the blackout date.
end_date	Date	The end date of the blackout date.
event_title	string	The title of the blackout date.

Returns a [BlackoutDate](#) object

[Update Blackout Date BlackoutDatesController#update](#)

**PUT /api/v1/courses/:course\_id/blackout\_dates/:id**

**Scope:** url:PUT|/api/v1/courses/:course\_id/blackout\_dates/:id

**PUT /api/v1/accounts/:account\_id/blackout\_dates/:id**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/blackout\_dates/:id

Update a blackout date for the given context.

**Request Parameters:**

Parameter	Type	Description
start_date	Date	The start date of the blackout date.

Parameter	Type	Description
end_date	Date	The end date of the blackout date.
event_title	string	The title of the blackout date.

Returns a [BlackoutDate](#) object

**Delete Blackout Date [BlackoutDatesController#destroy](#)**

**DELETE /api/v1/courses/:course\_id/blackout\_dates/:id**

**Scope:** url:DELETE|/api/v1/courses/:course\_id/blackout\_dates/:id

**DELETE /api/v1/accounts/:account\_id/blackout\_dates/:id**

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/blackout\_dates/:id

Delete a blackout date for the given context.

Returns a [BlackoutDate](#) object

**Update a list of Blackout Dates [BlackoutDatesController#bulk\\_update](#)**

**PUT /api/v1/courses/:course\_id/blackout\_dates**

**Scope:** url:PUT|/api/v1/courses/:course\_id/blackout\_dates

Create, update, and delete blackout dates to sync the db with the incoming data.

**Request Parameters:**

Parameter	Type	Description
		blackout_date, ...
blackout_dates	string	An object containing the array of BlackoutDates we want to exist after this operation. For array entries, if it has an id it will be updated, if not created, and if an existing BlackoutDate id is missing from the array, it will be deleted.

**List block templates [BlockEditorTemplatesApiController#index](#)**

**GET /api/v1/courses/:course\_id/block\_editor\_templates**

**Scope:** url:GET|/api/v1/courses/:course\_id/block\_editor\_templates

A list of the block templates available to the current user.

**Request Parameters:**

Parameter	Type	Description
		Sort results by this field.
sort	string	Allowed values: name, created_at, updated_at
		The sorting order. Defaults to 'asc'.
order	string	Allowed values: asc, desc
		If true, include draft templates. If false or omitted only published templates will be returned.
drafts	boolean	
		What type of templates should be returned.
type[]	string	Allowed values: page, section, block
		no description
include[]	string	Allowed values: node_tree, thumbnail

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/block_editor_templates?sort=name&order=asc&drafts=true
```

Returns a list of [BlockEditorTemplate](#) objects

**Get blueprint information [MasterCourses::MasterTemplatesController#show](#)**

**GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id**

**Scope:** url:GET|/api/v1/courses/:course\_id/blueprint\_templates/:template\_id

Using 'default' as the template\_id should suffice for the current implementation (as there should be only one template per course). However, using specific template ids may become necessary in the future

**Example Request:**

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a [BlueprintTemplate](#) object

**Get associated course information [MasterCourses::MasterTemplatesController#associated\\_courses](#)**

**GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/associated\_courses**

**Scope:** url:GET|/api/v1/courses/:course\_id/blueprint\_templates/:template\_id/associated\_courses

Returns a list of courses that are configured to receive updates from this blueprint

**Example Request:**

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/associated_courses \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a list of [Course](#) objects

**Update associated courses [MasterCourses::MasterTemplatesController#update\\_associations](#)**

**PUT /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/update\_associations**

**Scope:** url:PUT|/api/v1/courses/:course\_id/blueprint\_templates/:template\_id/update\_associations

Send a list of course ids to add or remove new associations for the template. Cannot add courses that do not belong to the blueprint course's account. Also cannot add other blueprint courses or courses that already have an association with another blueprint course.

After associating new courses, [start a sync](#) to populate their contents from the blueprint.

**Request Parameters:**

Parameter	Type	Description
course_ids_to_add	Array	Courses to add as associated courses

Parameter	Type	Description
course_ids_to_remove	Array	Courses to remove as associated courses

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/update_associations \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'course_ids_to_add[]=1' \
-d 'course_ids_to_remove[]=2' \
```

Begin a migration to push to associated courses MasterCourses::MasterTemplatesController#queue\_migration

POST /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations

Scope: url: POST /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations

Begins a migration to push recently updated content to all associated courses. Only one migration can be running at a time.

Request Parameters:

Parameter	Type	Description
comment	string	An optional comment to be included in the sync history.
send_notification	boolean	Send a notification to the calling user when the sync completes.
copy_settings	boolean	Whether course settings should be copied over to associated courses. Defaults to true for newly associated courses.
send_item_notifications	boolean	By default, new-item notifications are suppressed in blueprint syncs. If this option is set, teachers and students may receive notifications for items such as announcements and assignments that are created in associated courses (subject to the usual notification settings). This option requires the Blueprint Item Notifications feature to be enabled.
publish_after_initial_sync	boolean	If set, newly associated courses will be automatically published after the sync completes

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/migrations \
-X POST \
-F 'comment=Fixed spelling in question 3 of midterm exam' \
-F 'send_notification=true' \
-H 'Authorization: Bearer <token>'
```

Returns a [BlueprintMigration](#) object

Set or remove restrictions on a blueprint course object MasterCourses::MasterTemplatesController#restrict\_item

PUT /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/restrict\_item

Scope: url: PUT /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/restrict\_item

If a blueprint course object is restricted, editing will be limited for copies in associated courses.

Request Parameters:

Parameter	Type	Description
content_type	string	String, “assignment”   “attachment”   “discussion_topic”   “external_tool”   “lti-quiz”   “quiz”   “wiki_page” The type of the object.
content_id	integer	The ID of the object.
restricted	boolean	Whether to apply restrictions.
restrictions	BlueprintRestriction (Optional)	If the object is restricted, this specifies a set of restrictions. If not specified, the course-level restrictions will be used. See <a href="#">Course API update documentation</a>

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/restrict_item \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'content_type=assignment' \
-d 'content_id=2' \
-d 'restricted=true'
```

Get unsynced changes MasterCourses::MasterTemplatesController#unsynced\_changes

GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/unsynced\_changes

Scope: url: GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/unsynced\_changes

Retrieve a list of learning objects that have changed since the last blueprint sync operation. If no syncs have been completed, a ChangeRecord with a change\_type of initial\_sync is returned.

Returns a list of [ChangeRecord](#) objects

List blueprint migrations MasterCourses::MasterTemplatesController#migrations\_index

GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations

Scope: url: GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations

Shows a paginated list of migrations for the template, starting with the most recent. This endpoint can be called on a blueprint course. See also [the associated course side](#).

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/migrations \
-H 'Authorization: Bearer <token>'
```

Returns a list of [BlueprintMigration](#) objects

Show a blueprint migration MasterCourses::MasterTemplatesController#migrations\_show

GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations/:id

Scope: url: GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations/:id

Shows the status of a migration. This endpoint can be called on a blueprint course. See also [the associated course side](#).

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/migrations/:id \
-H 'Authorization: Bearer <token>'
```

Returns a [BlueprintMigration](#) object

Get migration details MasterCourses::MasterTemplatesController#migration\_details

GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations/:id/details

Scope: url: GET /api/v1/courses/:course\_id/blueprint\_templates/:template\_id/migrations/:id/details

Show the changes that were propagated in a blueprint migration. This endpoint can be called on a blueprint course. See also [the associated course side](#).

Example Request:

```
curl https://<canvas>/api/v1/courses/1/blueprint_templates/default/migrations/2/details \
-H 'Authorization: Bearer <token>'
```

Returns a list of [ChangeRecord](#) objects

[List blueprint subscriptions MasterCourses::MasterTemplatesController#subscriptions\\_index](#)

**GET /api/v1/courses/:course\_id/blueprint\_subscriptions**

**Scope:** url:GET[/api/v1/courses/:course\_id/blueprint\_subscriptions

Returns a list of blueprint subscriptions for the given course. (Currently a course may have no more than one.)

**Example Request:**

```
curl https://<canvas>/api/v1/courses/2/blueprint_subscriptions \
-H 'Authorization: Bearer <token>'
```

Returns a list of [BlueprintSubscription](#) objects

[List blueprint imports MasterCourses::MasterTemplatesController#imports\\_index](#)

**GET /api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations**

**Scope:** url:GET[/api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations

Shows a paginated list of migrations imported into a course associated with a blueprint, starting with the most recent. See also [the blueprint course side](#).

Use 'default' as the subscription\_id to use the currently active blueprint subscription.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/2/blueprint_subscriptions/default/migrations \
-H 'Authorization: Bearer <token>'
```

Returns a list of [BlueprintMigration](#) objects

[Show a blueprint import MasterCourses::MasterTemplatesController#imports\\_show](#)

**GET /api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations/:id**

**Scope:** url:GET[/api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations/:id

Shows the status of an import into a course associated with a blueprint. See also [the blueprint course side](#).

**Example Request:**

```
curl https://<canvas>/api/v1/courses/2/blueprint_subscriptions/default/migrations/:id \
-H 'Authorization: Bearer <token>'
```

Returns a [BlueprintMigration](#) object

[Get import details MasterCourses::MasterTemplatesController#import\\_details](#)

**GET /api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations/:id/details**

**Scope:** url:GET[/api/v1/courses/:course\_id/blueprint\_subscriptions/subscription\_id/migrations/:id/details

Show the changes that were propagated to a course associated with a blueprint. See also [the blueprint course side](#).

**Example Request:**

```
curl https://<canvas>/api/v1/courses/2/blueprint_subscriptions/default/7/details \
-H 'Authorization: Bearer <token>'
```

Returns a list of [ChangeRecord](#) objects

[List bookmarks Bookmarks::BookmarksController#index](#)

**GET /api/v1/users/self/bookmarks**

**Scope:** url:GET[/api/v1/users/self/bookmarks

Returns the paginated list of bookmarks.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/bookmarks' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Bookmark](#) objects

[Create bookmark Bookmarks::BookmarksController#create](#)

**POST /api/v1/users/self/bookmarks**

**Scope:** url:POST[/api/v1/users/self/bookmarks

Creates a bookmark.

**Request Parameters:**

Parameter	Type	Description
name	string	The name of the bookmark
url	string	The url of the bookmark
position	integer	The position of the bookmark. Defaults to the bottom.
data	string	The data associated with the bookmark

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/bookmarks' \
-F 'name=Biology 101' \
-F 'url=/courses/1' \
-H 'Authorization: Bearer <token>'
```

Returns a [Bookmark](#) object

[Get bookmark Bookmarks::BookmarksController#show](#)

**GET /api/v1/users/self/bookmarks/:id**

**Scope:** url:GET[/api/v1/users/self/bookmarks/:id

Returns the details for a bookmark.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/bookmarks/1' \
-H 'Authorization: Bearer <token>'
```

Returns a [Bookmark](#) object

[Update bookmark Bookmarks::BookmarksController#update](#)

**PUT /api/v1/users/self/bookmarks/:id**

**Scope:** url:PUT[/api/v1/users/self/bookmarks/:id

Updates a bookmark

Request Parameters:

Parameter	Type	Description
name	string	The name of the bookmark
url	string	The url of the bookmark
position	integer	The position of the bookmark. Defaults to the bottom.
data	string	The data associated with the bookmark

Example Request:

```
curl -X PUT 'https://<canvas>/api/v1/users/self/bookmarks/1' \  
-F 'name=Biology 101' \  
-F 'url=/courses/1' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Folder](#) object

Delete bookmark [Bookmarks::BookmarksController#destroy](#)

DELETE /api/v1/users/self/bookmarks/:id

Scope: url:DELETE|/api/v1/users/self/bookmarks/:id

Deletes a bookmark

Example Request:

```
curl -X DELETE 'https://<canvas>/api/v1/users/self/bookmarks/1' \  
-H 'Authorization: Bearer <token>'
```

Get the brand config variables that should be used for this domain [BrandConfigsApiController#show](#)

GET /api/v1/brand\_variables

Scope: url:GET|/api/v1/brand\_variables

Will redirect to a static json file that has all of the brand variables used by this account. Even though this is a redirect, do not store the redirected url since if the account makes any changes it will redirect to a new url. Needs no authentication.

Example Request:

```
curl 'https://<canvas>/api/v1/brand_variables'
```

List calendar events [CalendarEventsApiController#index](#)

GET /api/v1/calendar\_events

Scope: url:GET|/api/v1/calendar\_events

Retrieve the paginated list of calendar events or assignments for the current user

Request Parameters:

Parameter	Type	Description
type	string	Defaults to “event” Allowed values: event, assignment, sub_assignment
start_date	Date	Only return events since the start_date (inclusive). Defaults to today. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
end_date	Date	Only return events before the end_date (inclusive). Defaults to start_date. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ. If end_date is the same as start_date, then only events on that day are returned.
undated	boolean	Defaults to false (dated events only). If true, only return undated events and ignore start_date and end_date.
all_events	boolean	Defaults to false (uses start_date, end_date, and undated criteria). If true, all events are returned, ignoring start_date, end_date, and undated criteria.
context_codes[]	string	List of context codes of courses, groups, users, or accounts whose events you want to see. If not specified, defaults to the current user (i.e personal calendar, no course/group events). Limited to 10 context codes, additional ones are ignored. The format of this field is the context type, followed by an underscore, followed by the context id. For example: course_42
excludes[]	Array	Array of attributes to exclude. Possible values are “description”, “child_events” and “assignment”
includes[]	Array	Array of optional attributes to include. Possible values are “web_conference” and “series_natural_language”
important_dates	boolean	Defaults to false. If true, only events with important dates set to true will be returned.
blackout_date	boolean	Defaults to false. If true, only events with blackout date set to true will be returned.

Returns a list of [CalendarEvent](#) objects

List calendar events for a user [CalendarEventsApiController#user\\_index](#)

GET /api/v1/users/:user\_id/calendar\_events

Scope: url:GET|/api/v1/users/:user\_id/calendar\_events

Retrieve the paginated list of calendar events or assignments for the specified user. To view calendar events for a user other than yourself, you must either be an observer of that user or an administrator.

Request Parameters:

Parameter	Type	Description
type	string	Defaults to “event” Allowed values: event, assignment
start_date	Date	Only return events since the start_date (inclusive). Defaults to today. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
end_date	Date	Only return events before the end_date (inclusive). Defaults to start_date. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ. If end_date is the same as start_date, then only events on that day are returned.
undated	boolean	Defaults to false (dated events only). If true, only return undated events and ignore start_date and end_date.
all_events	boolean	Defaults to false (uses start_date, end_date, and undated criteria). If true, all events are returned, ignoring start_date, end_date, and undated criteria.
context_codes[]	string	List of context codes of courses, groups, users, or accounts whose events you want to see. If not specified, defaults to the current user (i.e personal calendar, no course/group events). Limited to 10 context codes, additional ones are ignored. The format of this field is the context type, followed by an underscore, followed by the context id. For example: course_42
excludes[]	Array	Array of attributes to exclude. Possible values are “description”, “child_events” and “assignment”

Parameter	Type	Description
submission_types[]	Array	When type is “assignment”, specifies the allowable submission types for returned assignments. Ignored if type is not “assignment” or if exclude_submission_types is provided.
exclude_submission_types[]	Array	When type is “assignment”, specifies the submission types to be excluded from the returned assignments. Ignored if type is not “assignment”.
includes[]	Array	Array of optional attributes to include. Possible values are “web_conference” and “series_natural_language”
important_dates	boolean	Defaults to false If true, only events with important dates set to true will be returned.
blackout_date	boolean	Defaults to false If true, only events with blackout date set to true will be returned.
Returns a list of <a href="#">CalendarEvent</a> objects		

Create a calendar event [CalendarEventsApiController#create](#)

POST /api/v1/calendar\_events

**Scope:** url:POST[/api/v1/calendar\_events]  
Create and return a new calendar event

Request Parameters:

Parameter	Type	Description
calendar_event[context_code]	Required string	Context code of the course, group, user, or account whose calendar this event should be added to.
calendar_event[title]	string	Short title for the calendar event.
calendar_event[description]	string	Longer HTML description of the event.
calendar_event[start_at]	DateTime	Start date/time of the event.
calendar_event[end_at]	DateTime	End date/time of the event.
calendar_event[location_name]	string	Location name of the event.
calendar_event[location_address]	string	Location address
calendar_event[time_zone_edited]	string	Time zone of the user editing the event. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
calendar_event[all_day]	boolean	When true event is considered to span the whole day and times are ignored.
calendar_event[child_event_data][X][start_at]	DateTime	Section-level start time(s) if this is a course event. X can be any identifier, provided that it is consistent across the start_at, end_at and context_code
calendar_event[child_event_data][X][end_at]	DateTime	Section-level end time(s) if this is a course event.
calendar_event[child_event_data][X][context_code]	string	Context code(s) corresponding to the section-level start and end time(s).
calendar_event[duplicate][count]	number	Number of times to copy/duplicate the event. Count cannot exceed 200.
calendar_event[duplicate][interval]	number	Defaults to 1 if duplicate ‘count’ is set. The interval between the duplicated events.
calendar_event[duplicate][frequency]	string	Defaults to “weekly”. The frequency at which to duplicate the event Allowed values: daily, weekly, monthly
calendar_event[duplicate][append_iterator]	boolean	Defaults to false. If set to ‘true’, an increasing counter number will be appended to the event title when the event is duplicated. (e.g. Event 1, Event 2, Event 3, etc)
calendar_event[rrule]	string	The recurrence rule to create a series of recurring events. Its value is the <a href="#">iCalendar RRULE</a> defining how the event repeats. Unending series not supported.
calendar_event[blackout_date]	boolean	If the blackout_date is true, this event represents a holiday or some other special day that does not count in course pacing.

Example Request:

```
curl 'https://<canvas>/api/v1/calendar_events.json' \  
-X POST \  
-F 'calendar_event[context_code]=course_123' \  
-F 'calendar_event[title]=Paintball Fight!' \  
-F 'calendar_event[start_at]=2012-07-19T21:00:00Z' \  
-F 'calendar_event[end_at]=2012-07-19T22:00:00Z' \  
-H "Authorization: Bearer <token>"
```

Get a single calendar event or assignment [CalendarEventsApiController#show](#)

GET /api/v1/calendar\_events/:id

**Scope:** url:GET[/api/v1/calendar\_events/:id]

Reserve a time slot [CalendarEventsApiController#reserve](#)

POST /api/v1/calendar\_events/:id/reservations

**Scope:** url:POST[/api/v1/calendar\_events/:id/reservations]

POST /api/v1/calendar\_events/:id/reservations/:participant\_id

**Scope:** url:POST[/api/v1/calendar\_events/:id/reservations/:participant\_id]  
Reserves a particular time slot and return the new reservation

Request Parameters:

Parameter	Type	Description
participant_id	string	User or group id for whom you are making the reservation (depends on the participant type). Defaults to the current user (or user’s candidate group).
comments	string	Comments to associate with this reservation
cancel_existing	boolean	Defaults to false. If true, cancel any previous reservation(s) for this participant and appointment group.

Example Request:

```
curl 'https://<canvas>/api/v1/calendar_events/345/reservations.json' \  
-X POST \  
-F 'cancel_existing=true' \  
-H "Authorization: Bearer <token>"
```

Update a calendar event [CalendarEventsApiController#update](#)

PUT /api/v1/calendar\_events/:id

**Scope:** url:PUT /api/v1/calendar\_events/:id

Update and return a calendar event

**Request Parameters:**

Parameter	Type	Description
calendar_event[context_code]	string	Context code of the course, group, user, or account to move this event to. Scheduler appointments and events with section-specific times cannot be moved between calendars.
calendar_event[title]	string	Short title for the calendar event.
calendar_event[description]	string	Longer HTML description of the event.
calendar_event[start_at]	DateTime	Start date/time of the event.
calendar_event[end_at]	DateTime	End date/time of the event.
calendar_event[location_name]	string	Location name of the event.
calendar_event[location_address]	string	Location address
calendar_event[time_zone_edited]	string	Time zone of the user editing the event. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
calendar_event[all_day]	boolean	When true event is considered to span the whole day and times are ignored.
calendar_event[child_event_data][X][start_at]	DateTime	Section-level start time(s) if this is a course event. X can be any identifier, provided that it is consistent across the start_at, end_at and context_code
calendar_event[child_event_data][X][end_at]	DateTime	Section-level end time(s) if this is a course event.
calendar_event[child_event_data][X][context_code]	string	Context code(s) corresponding to the section-level start and end time(s).
calendar_event[rrule]	string	Valid if the event whose ID is in the URL is part of a series. This defines the shape of the recurring event series after it's updated. Its value is the iCalendar RRULE. Unending series are not supported.
which	string	Valid if the event whose ID is in the URL is part of a series. Update just the event whose ID is in in the URL, all events in the series, or the given event and all those following. Some updates may create a new series. For example, changing the start time of this and all following events from the middle of a series.  Allowed values: one, all, following
calendar_event[blackout_date]	boolean	If the blackout_date is true, this event represents a holiday or some other special day that does not count in course pacing.

**Example Request:**

```
curl 'https://<canvas>/api/v1/calendar_events/234' \  
-X PUT \  
-F 'calendar_event[title]=Epic Paintball Fight!' \  
-H "Authorization: Bearer <token>"
```

[Delete a calendar event CalendarEventsApiController#destroy](#)

**DELETE /api/v1/calendar\_events/:id**

**Scope:** url:DELETE /api/v1/calendar\_events/:id

Delete an event from the calendar and return the deleted event

**Request Parameters:**

Parameter	Type	Description
cancel_reason	string	Reason for deleting/canceling the event.
which	string	Valid if the event whose ID is in the URL is part of a series. Delete just the event whose ID is in in the URL, all events in the series, or the given event and all those following.  Allowed values: one, all, following

**Example Request:**

```
curl 'https://<canvas>/api/v1/calendar_events/234' \  
-X DELETE \  
-F 'cancel_reason=Greendale layed off the janitorial staff :(' \  
-F 'which=following' \  
-H "Authorization: Bearer <token>"
```

[Save enabled account calendars CalendarEventsApiController#save\\_enabled\\_account\\_calendars](#)

**POST /api/v1/calendar\_events/save\_enabled\_account\_calendars**

**Scope:** url:POST /api/v1/calendar\_events/save\_enabled\_account\_calendars

Creates and updates the enabled\_account\_calendars and mark\_feature\_as\_seen user preferences

**Request Parameters:**

Parameter	Type	Description
mark_feature_as_seen	boolean	Flag to mark account calendars feature as seen
enabled_account_calendars[]	Array	An array of account Ids to remember in the calendars list of the user

**Example Request:**

```
curl 'https://<canvas>/api/v1/calendar_events/save_enabled_account_calendars' \  
-X POST \  
-F 'mark_feature_as_seen=true' \  
-F 'enabled_account_calendars[]=1' \  
-F 'enabled_account_calendars[]=2' \  
-H "Authorization: Bearer <token>"
```

[Set a course timetable CalendarEventsApiController#set\\_course\\_timetable](#)

**POST /api/v1/courses/:course\_id/calendar\_events/timetable**

**Scope:** url:POST /api/v1/courses/:course\_id/calendar\_events/timetable

Creates and updates "timetable" events for a course. Can automatically generate a series of calendar events based on simple schedules (e.g. "Monday and Wednesday at 2:00pm")

Existing timetable events for the course and course sections will be updated if they still are part of the timetable. Otherwise, they will be deleted.

**Request Parameters:**

Parameter	Type	Description
timetables[course_section_id][]	Array	An array of timetable objects for the course section specified by course_section_id. If course_section_id is set to "all", events will be created for the entire course.



Parameter	Type	Description
timetables[course_section_id][weekdays]	string	A comma-separated list of abbreviated weekdays (Mon-Monday, Tue-Tuesday, Wed-Wednesday, Thu-Thursday, Fri-Friday, Sat-Saturday, Sun-Sunday)
timetables[course_section_id][start_time]	string	Time to start each event at (e.g. "9:00 am")
timetables[course_section_id][end_time]	string	Time to end each event at (e.g. "9:00 am")
timetables[course_section_id][location_name]	string	A location name to set for each event

Example Request:

```
curl 'https://<canvas>/api/v1/calendar_events/timetable' \
-X POST \
-F 'timetables[all][weekdays]=Mon,Wed,Fri' \
-F 'timetables[all][start_time]=11:00 am' \
-F 'timetables[all][end_time]=11:50 am' \
-F 'timetables[all][location_name]=Room 237' \
-H 'Authorization: Bearer <token>'
```

Get course timetable [CalendarEventsApiController#get\\_course\\_timetable](#)

GET /api/v1/courses/:course\_id/calendar\_events/timetable

Scope: url:GET /api/v1/courses/:course\_id/calendar\_events/timetable

Returns the last timetable set by the [Set a course timetable](#) endpoint

Create or update events directly for a course timetable [CalendarEventsApiController#set\\_course\\_timetable\\_events](#)

POST /api/v1/courses/:course\_id/calendar\_events/timetable\_events

Scope: url:POST /api/v1/courses/:course\_id/calendar\_events/timetable\_events

Creates and updates "timetable" events for a course or course section. Similar to [setting a course timetable](#), but instead of generating a list of events based on a timetable schedule, this endpoint expects a complete list of events.

Request Parameters:

Parameter	Type	Description
course_section_id	string	Events will be created for the course section specified by course_section_id. If not present, events will be created for the entire course.
events[]	Array	An array of event objects to use.
events[][start_at]	DateTime	Start time for the event
events[][end_at]	DateTime	End time for the event
events[][location_name]	string	Location name for the event
events[][code]	string	A unique identifier that can be used to update the event at a later time If one is not specified, an identifier will be generated based on the start and end times
events[][title]	string	Title for the meeting. If not present, will default to the associated course's name

List collaborations [CollaborationsController#api\\_index](#)

GET /api/v1/courses/:course\_id/collaborations

Scope: url:GET /api/v1/courses/:course\_id/collaborations

GET /api/v1/groups/:group\_id/collaborations

Scope: url:GET /api/v1/groups/:group\_id/collaborations

A paginated list of collaborations the current user has access to in the context of the course provided in the url. NOTE: this only returns ExternalToolCollaboration type collaborations.

```
curl https://<canvas>/api/v1/courses/1/collaborations/
```

Returns a list of [Collaboration](#) objects

List members of a collaboration. [CollaborationsController#members](#)

GET /api/v1/collaborations/:id/members

Scope: url:GET /api/v1/collaborations/:id/members

A paginated list of the collaborators of a given collaboration

Request Parameters:

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"collaborator_lti_id": Optional information to include with each member. Represents an identifier to be used for the member in an LTI context.</li><li>"avatar_image_url": Optional information to include with each member. The url for the avatar of a collaborator with type 'user'.</li></ul> Allowed values: collaborator_lti_id, avatar_image_url

Example Request:

```
curl https://<canvas>/api/v1/courses/1/collaborations/1/members
```

Returns a list of [Collaborator](#) objects

List potential members [CollaborationsController#potential\\_collaborators](#)

GET /api/v1/courses/:course\_id/potential\_collaborators

Scope: url:GET /api/v1/courses/:course\_id/potential\_collaborators

GET /api/v1/groups/:group\_id/potential\_collaborators

Scope: url:GET /api/v1/groups/:group\_id/potential\_collaborators

A paginated list of the users who can potentially be added to a collaboration in the given context.

For courses, this consists of all enrolled users. For groups, it is comprised of the group members plus the admins of the course containing the group.

Returns a list of [User](#) objects

List of CommMessages for a user [CommMessagesApiController#index](#)

GET /api/v1/comm\_messages

Scope: url:GET /api/v1/comm\_messages

Retrieve a paginated list of messages sent to a user.

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
user_id	Required string	The user id for whom you want to retrieve CommMessages
start_time	DateTime	The beginning of the time range you want to retrieve message from. Up to a year prior to the current date is available.
end_time	DateTime	The end of the time range you want to retrieve messages for. Up to a year prior to the current date is available.

Returns a list of [CommMessage](#) objects

List user communication channels [CommunicationChannelsController#index](#)

GET /api/v1/users/:user\_id/communication\_channels

Scope: url:GET[/api/v1/users/:user\_id/communication\_channels]

Returns a paginated list of communication channels for the specified user, sorted by position.

Example Request:

```
curl https://<canvas>/api/v1/users/12345/communication_channels \
-H 'Authorization: Bearer <token>'
```

Returns a list of [CommunicationChannel](#) objects

Create a communication channel [CommunicationChannelsController#create](#)

POST /api/v1/users/:user\_id/communication\_channels

Scope: url:POST[/api/v1/users/:user\_id/communication\_channels]

Creates a new communication channel for the specified user.

Request Parameters:

Parameter	Type	Description
communication_channel[address]	Required string	An email address or SMS number. Not required for "push" type channels.
communication_channel[type]	Required string	The type of communication channel.  In order to enable push notification support, the server must be properly configured (via 'sns_creds' in Vault) to communicate with Amazon Simple Notification Services, and the developer key used to create the access token from this request must have an SNS ARN configured on it.  Allowed values: email, sms, push
communication_channel[token]	string	A registration id, device token, or equivalent token given to an app when registering with a push notification provider. Only valid for "push" type channels.
skip_confirmation	boolean	Only valid for site admins and account admins making requests; If true, the channel is automatically validated and no confirmation email or SMS is sent. Otherwise, the user must respond to a confirmation message to confirm the channel.

Example Request:

```
curl https://<canvas>/api/v1/users/1/communication_channels \
-H 'Authorization: Bearer <token>' \
-d 'communication_channel[address]=new@example.com' \
-d 'communication_channel[type]=email' \
-X DELETE
```

Returns a [CommunicationChannel](#) object

Delete a communication channel [CommunicationChannelsController#destroy](#)

DELETE /api/v1/users/:user\_id/communication\_channels/:id

Scope: url:DELETE[/api/v1/users/:user\_id/communication\_channels/:id]

DELETE /api/v1/users/:user\_id/communication\_channels/:type/:address

Scope: url:DELETE[/api/v1/users/:user\_id/communication\_channels/:type/:address]

Delete an existing communication channel.

Example Request:

```
curl https://<canvas>/api/v1/users/5/communication_channels/3 \
-H 'Authorization: Bearer <token>' \
-X DELETE
```

Returns a [CommunicationChannel](#) object

Delete a push notification endpoint [CommunicationChannelsController#delete\\_push\\_token](#)

DELETE /api/v1/users/self/communication\_channels/push

Scope: url:DELETE[/api/v1/users/self/communication\_channels/push]

Example Request:

```
curl https://<canvas>/api/v1/users/self/communication_channels/push \
-H 'Authorization: Bearer <token>' \
-X DELETE \
-d 'push_token=<push_token>'
```

List conferences [ConferencesController#index](#)

GET /api/v1/courses/:course\_id/conferences

Scope: url:GET[/api/v1/courses/:course\_id/conferences]

GET /api/v1/groups/:group\_id/conferences

Scope: url:GET[/api/v1/groups/:group\_id/conferences]

Retrieve the paginated list of conferences for this context

This API returns a JSON object containing the list of conferences, the key for the list of conferences is "conferences"

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/conferences' \
-H 'Authorization: Bearer <token>' \
curl 'https://<canvas>/api/v1/groups/<group_id>/conferences' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Conference](#) objects

List conferences for the current user [ConferencesController#for\\_user](#)

GET /api/v1/conferences

Scope: url:GET[/api/v1/conferences]

Retrieve the paginated list of conferences for all courses and groups the current user belongs to

This API returns a JSON object containing the list of conferences. The key for the list of conferences is "conferences".

Request Parameters:

Parameter	Type	Description
state	string	If set to "live", returns only conferences that are live (i.e., have started and not finished yet). If omitted, returns all conferences for this user's groups and courses.

Example Request:

```
curl 'https://<canvas>/api/v1/conferences' \  
-H "Authorization: Bearer <token>"
```

Returns a list of [Conference](#) objects

List content exports [ContentExportsApiController#index](#)

GET /api/v1/courses/:course\_id/content\_exports

Scope: url:GET//api/v1/courses/:course\_id/content\_exports

GET /api/v1/groups/:group\_id/content\_exports

Scope: url:GET//api/v1/groups/:group\_id/content\_exports

GET /api/v1/users/:user\_id/content\_exports

Scope: url:GET//api/v1/users/:user\_id/content\_exports

A paginated list of the past and pending content export jobs for a course, group, or user. Exports are returned newest first.

Returns a list of [ContentExport](#) objects

Show content export [ContentExportsApiController#show](#)

GET /api/v1/courses/:course\_id/content\_exports/:id

Scope: url:GET//api/v1/courses/:course\_id/content\_exports/:id

GET /api/v1/groups/:group\_id/content\_exports/:id

Scope: url:GET//api/v1/groups/:group\_id/content\_exports/:id

GET /api/v1/users/:user\_id/content\_exports/:id

Scope: url:GET//api/v1/users/:user\_id/content\_exports/:id

Get information about a single content export.

Returns a [ContentExport](#) object

Export content [ContentExportsApiController#create](#)

POST /api/v1/courses/:course\_id/content\_exports

Scope: url:POST//api/v1/courses/:course\_id/content\_exports

POST /api/v1/groups/:group\_id/content\_exports

Scope: url:POST//api/v1/groups/:group\_id/content\_exports

POST /api/v1/users/:user\_id/content\_exports

Scope: url:POST//api/v1/users/:user\_id/content\_exports

Begin a content export job for a course, group, or user.

You can use the [Progress API](#) to track the progress of the export. The migration's progress is linked to with the *progress\_url* value.

When the export completes, use the [Show content export](#) endpoint to retrieve a download URL for the exported content.

Request Parameters:

Parameter	Type	Description
export_type	Required string	"common_cartridge" Export the contents of the course in the Common Cartridge (.imsc) format
		"qti" Export quizzes from a course in the QTI format
		"zip" Export files from a course, group, or user in a zip file
		Allowed values: common_cartridge, qti, zip
skip_notifications	boolean	Don't send the notifications about the export to the user. Default: false
select	Hash	The select parameter allows exporting specific data. The keys are object types like 'files', 'folders', 'pages', etc. The value for each key is a list of object ids. An id can be an integer or a string.
		Multiple object types can be selected in the same call. However, not all object types are valid for every export_type. Common Cartridge supports all object types. Zip and QTI only support the object types as described below.
		"folders" Also supported for zip export_type.
		"files" Also supported for zip export_type.
		"quizzes" Also supported for qti export_type.
		Allowed values: folders, files, attachments, quizzes, assignments, announcements, calendar_events, discussion_topics, modules, module_items, pages, rubrics

Returns a [ContentExport](#) object

List migration issues [MigrationIssuesController#index](#)

GET /api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues

Scope: url:GET//api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues

GET /api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues

Scope: url:GET//api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues

GET /api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues

Scope: url:GET//api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues

GET /api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues

Scope: url:GET//api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues

Returns paginated migration issues

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/content_migrations/<content_migration_id>/migration_issues \
-H 'Authorization: Bearer <token>'
```

Returns a list of [MigrationIssue](#) objects

**[Get a migration issue MigrationIssuesController#show](#)**

**GET /api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:GET//api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**GET /api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:GET//api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**GET /api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:GET//api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**GET /api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:GET//api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

Returns data on an individual migration issue

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/content_migrations/<content_migration_id>/migration_issues/<id> \
-H 'Authorization: Bearer <token>'
```

Returns a [MigrationIssue](#) object

**[Update a migration issue MigrationIssuesController#update](#)**

**PUT /api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:PUT//api/v1/accounts/:account\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**PUT /api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:PUT//api/v1/courses/:course\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**PUT /api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:PUT//api/v1/groups/:group\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

**PUT /api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id**

**Scope:** url:PUT//api/v1/users/:user\_id/content\_migrations/:content\_migration\_id/migration\_issues/:id

Update the workflow\_state of a migration issue

**Request Parameters:**

Parameter	Type	Description
workflow_state	Required string	Set the workflow_state of the issue. Allowed values: active, resolved

**Example Request:**

```
curl -X PUT https://<canvas>/api/v1/courses/<course_id>/content_migrations/<content_migration_id>/migration_issues/<id> \
-H 'Authorization: Bearer <token>' \
-F 'workflow_state=resolved'
```

Returns a [MigrationIssue](#) object

**[List content migrations ContentMigrationsController#index](#)**

**GET /api/v1/accounts/:account\_id/content\_migrations**

**Scope:** url:GET//api/v1/accounts/:account\_id/content\_migrations

**GET /api/v1/courses/:course\_id/content\_migrations**

**Scope:** url:GET//api/v1/courses/:course\_id/content\_migrations

**GET /api/v1/groups/:group\_id/content\_migrations**

**Scope:** url:GET//api/v1/groups/:group\_id/content\_migrations

**GET /api/v1/users/:user\_id/content\_migrations**

**Scope:** url:GET//api/v1/users/:user\_id/content\_migrations

Returns paginated content migrations

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/content_migrations \
-H 'Authorization: Bearer <token>'
```

Returns a list of [ContentMigration](#) objects

**[Get a content migration ContentMigrationsController#show](#)**

**GET /api/v1/accounts/:account\_id/content\_migrations/:id**

**Scope:** url:GET//api/v1/accounts/:account\_id/content\_migrations/:id

**GET /api/v1/courses/:course\_id/content\_migrations/:id**

**Scope:** url:GET//api/v1/courses/:course\_id/content\_migrations/:id

**GET /api/v1/groups/:group\_id/content\_migrations/:id**

**Scope:** url:GET//api/v1/groups/:group\_id/content\_migrations/:id

**GET /api/v1/users/:user\_id/content\_migrations/:id**

**Scope:** url:GET//api/v1/users/:user\_id/content\_migrations/:id

Returns data on an individual content migration

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/content_migrations/<id> \
-H 'Authorization: Bearer <token>'
```

Returns a [ContentMigration](#) object

**[Create a content migration ContentMigrationsController#create](#)**

**POST /api/v1/accounts/:account\_id/content\_migrations**

**Scope:** url:POST//api/v1/accounts/:account\_id/content\_migrations

**POST /api/v1/courses/:course\_id/content\_migrations**

**Scope:** url:POST//api/v1/courses/:course\_id/content\_migrations

POST /api/v1/groups/:group\_id/content\_migrations

Scope: url:POST|/api/v1/groups/:group\_id/content\_migrations

POST /api/v1/users/:user\_id/content\_migrations

Scope: url:POST|/api/v1/users/:user\_id/content\_migrations

Create a content migration. If the migration requires a file to be uploaded the actual processing of the file will start once the file upload process is completed. File uploading works as described in the [File Upload Documentation](#) except that the values are set on a **pre\_attachment** sub-hash.

For migrations that don't require a file to be uploaded, like course copy, the processing will begin as soon as the migration is created.

You can use the [Progress API](#) to track the progress of the migration. The migration's progress is linked to with the *progress\_url* value.

The two general workflows are:

If no file upload is needed:

- 1. POST to create
- 2. Use the [Progress](#) specified in *progress\_url* to monitor progress

For file uploading:

- 1. POST to create with file info in **pre\_attachment**
- 2. Do [file upload processing](#) using the data in the **pre\_attachment** data
- 3. [GET](#) the ContentMigration
- 4. Use the [Progress](#) specified in *progress\_url* to monitor progress

(required if doing .zip file upload)

Request Parameters:

Parameter	Type	Description
migration_type	Required string	The type of the migration. Use the <a href="#">Migrator</a> endpoint to see all available migrators. Default allowed values: canvas_cartridge_importer, common_cartridge_importer, course_copy_importer, zip_file_importer, qti_converter, moodle_converter
pre_attachment[name]	string	Required if uploading a file. This is the first step in uploading a file to the content migration. See the <a href="#">File Upload Documentation</a> for details on the file upload workflow.
pre_attachment[*]	string	Other file upload properties. See <a href="#">File Upload Documentation</a>
settings[file_url]	string	A URL to download the file from. Must not require authentication.
settings[content_export_id]	string	The id of a ContentExport to import. This allows you to import content previously exported from Canvas without needing to download and re-upload it.
settings[source_course_id]	string	The course to copy from for a course copy migration. (required if doing course copy)
settings[folder_id]	string	The folder to unzip the .zip file into for a zip_file_import.
settings[overwrite_quizzes]	boolean	Whether to overwrite quizzes with the same identifiers between content packages.
settings[question_bank_id]	integer	The existing question bank ID to import questions into if not specified in the content package.
settings[question_bank_name]	string	The question bank to import questions into if not specified in the content package, if both bank id and name are set, id will take precedence.
settings[insert_into_module_id]	integer	The id of a module in the target course. This will add all imported items (that can be added to a module) to the given module.
settings[insert_into_module_type]	string	If provided (and insert_into_module_id is supplied), only add objects of the specified type to the module. Allowed values: assignment, discussion_topic, file, page, quiz
settings[insert_into_module_position]	integer	The (1-based) position to insert the imported items into the course (if insert_into_module_id is supplied). If this parameter is omitted, items will be added to the end of the module.
settings[move_to_assignment_group_id]	integer	The id of an assignment group in the target course. If provided, all imported assignments will be moved to the given assignment group.
settings[importer_skips]	Array	Set of importers to skip, even if otherwise selected by migration settings. Allowed values: all_course_settings, visibility_settings
settings[import_blueprint_settings]	boolean	Import the "use as blueprint course" setting as well as the list of locked items from the source course or package. The destination course must not be associated with an existing blueprint course and cannot have any student or observer enrollments.
date_shift_options[shift_dates]	boolean	Whether to shift dates in the copied course
date_shift_options[old_start_date]	Date	The original start date of the source content/course
date_shift_options[old_end_date]	Date	The original end date of the source content/course
date_shift_options[new_start_date]	Date	The new start date for the content/course
date_shift_options[new_end_date]	Date	The new end date for the source content/course
date_shift_options[day_substitutions][X]	integer	Move anything scheduled for day 'X' to the specified day. (0-Sunday, 1-Monday, 2-Tuesday, 3-Wednesday, 4-Thursday, 5-Friday, 6-Saturday)
date_shift_options[remove_dates]	boolean	Whether to remove dates in the copied course. Cannot be used in conjunction with <b>shift_dates</b> .
selective_import	boolean	If set, perform a selective import instead of importing all content. The migration will identify the contents of the package and then stop in the waiting_for_select workflow state. At this point, use the <a href="#">List items endpoint</a> to enumerate the contents of the package, identifying the copy parameters for the desired content. Then call the <a href="#">Update endpoint</a> and provide these copy parameters to start the import.
select	Hash	For course_copy_importer migrations, this parameter allows you to select the objects to copy without using the selective_import argument and waiting_for_select state as is required for uploaded imports (though that workflow is also supported for course copy migrations). The keys are object types like 'files', 'folders', 'pages', etc. The value for each key is a list of object ids. An id can be an integer or a string. Multiple object types can be selected in the same call.  Allowed values: folders, files, attachments, quizzes, assignments, announcements, calendar_events, discussion_topics, modules, module_items, pages, rubrics

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/content_migrations' \  
-F 'migration_type=common_cartridge_importer' \  
-F 'settings[question_bank_name]=importquestions' \  
-F 'date_shift_options[old_start_date]=1999-01-01' \  
-F 'date_shift_options[new_start_date]=2013-09-01' \  
-F 'date_shift_options[old_end_date]=1999-04-15' \  
-F 'date_shift_options[new_end_date]=2013-12-15' \  
-F 'date_shift_options[day_substitutions][1]=2' \  
-F 'date_shift_options[day_substitutions][2]=3' \  
-F 'date_shift_options[shift_dates]=true' \  
-F 'pre_attachment[name]=mycourse.imsc' \  
-F 'pre_attachment[size]=12345' \  
-H 'Authorization: Bearer <token>'
```

Returns a [ContentMigration](#) object

Update a content migration ContentMigrationsController#update

PUT /api/v1/accounts/:account\_id/content\_migrations/:id

Scope: url:PUT//api/v1/accounts/:account\_id/content\_migrations/:id

PUT /api/v1/courses/:course\_id/content\_migrations/:id

Scope: url:PUT//api/v1/courses/:course\_id/content\_migrations/:id

PUT /api/v1/groups/:group\_id/content\_migrations/:id

Scope: url:PUT//api/v1/groups/:group\_id/content\_migrations/:id

PUT /api/v1/users/:user\_id/content\_migrations/:id

Scope: url:PUT//api/v1/users/:user\_id/content\_migrations/:id

Update a content migration. Takes same arguments as [create](#) except that you can't change the migration type. However, changing most settings after the migration process has started will not do anything. Generally updating the content migration will be used when there is a file upload problem, or when importing content selectively. If the first upload has a problem you can supply new *pre\_attachment* values to start the process again.

Returns a [ContentMigration](#) object

List Migration Systems ContentMigrationsController#available\_migrators

GET /api/v1/accounts/:account\_id/content\_migrations/migrators

Scope: url:GET//api/v1/accounts/:account\_id/content\_migrations/migrators

GET /api/v1/courses/:course\_id/content\_migrations/migrators

Scope: url:GET//api/v1/courses/:course\_id/content\_migrations/migrators

GET /api/v1/groups/:group\_id/content\_migrations/migrators

Scope: url:GET//api/v1/groups/:group\_id/content\_migrations/migrators

GET /api/v1/users/:user\_id/content\_migrations/migrators

Scope: url:GET//api/v1/users/:user\_id/content\_migrations/migrators

Lists the currently available migration types. These values may change.

Returns a list of [Migrator](#) objects

List items for selective import ContentMigrationsController#content\_list

GET /api/v1/accounts/:account\_id/content\_migrations/:id/selective\_data

Scope: url:GET//api/v1/accounts/:account\_id/content\_migrations/:id/selective\_data

GET /api/v1/courses/:course\_id/content\_migrations/:id/selective\_data

Scope: url:GET//api/v1/courses/:course\_id/content\_migrations/:id/selective\_data

GET /api/v1/groups/:group\_id/content\_migrations/:id/selective\_data

Scope: url:GET//api/v1/groups/:group\_id/content\_migrations/:id/selective\_data

GET /api/v1/users/:user\_id/content\_migrations/:id/selective\_data

Scope: url:GET//api/v1/users/:user\_id/content\_migrations/:id/selective\_data

Enumerates the content available for selective import in a tree structure. Each node provides a *property* copy argument that can be supplied to the [Update endpoint](#) to selectively copy the content associated with that tree node and its children. Each node may also provide a *sub\_items\_url* or an array of *sub\_items* which you can use to obtain copy parameters for a subset of the resources in a given node.

If no type is sent you will get a list of the top-level sections in the content. It will look something like this:

```
{
  "type": "course_settings",
  "property": "copy[all_course_settings]",
  "title": "Course Settings"
},
{
  "type": "context_modules",
  "property": "copy[all_context_modules]",
  "title": "Modules",
  "count": 5,
  "sub_items_url": "http://example.com/api/v1/courses/22/content_migrations/77/selective_data?type=context_modules"
},
{
  "type": "assignments",
  "property": "copy[all_assignments]",
  "title": "Assignments",
  "count": 2,
  "sub_items_url": "http://localhost:3000/api/v1/courses/22/content_migrations/77/selective_data?type=assignments"
}
```

When a type is provided, nodes may be further divided via *sub\_items*. For example, using *type=assignments* results in a node for each assignment group and a *sub\_item* for each assignment, like this:

```
{
  "type": "assignment_groups",
  "title": "An Assignment Group",
  "property": "copy[assignment_groups][id_i855cf145eSacc7435e1bf1c6e2126e5f]",
  "sub_items": [
    {
      "type": "assignments",
      "title": "Assignment 1",
      "property": "copy[assignments][id_i2102a7fa93b29226774949298626719d]"
    },
    {
      "type": "assignments",
      "title": "Assignment 2",
      "property": "copy[assignments][id_i310cba275dc3f4aa8a3306bbe380979]"
    }
  ]
}
```

To import the items corresponding to a particular tree node, use the *property* as a parameter to the [Update endpoint](#) and assign a value of 1, for example:

copy[assignments][id\_i310cba275dc3f4aa8a3306bbe380979]=1

You can include multiple copy parameters to selectively import multiple items or groups of items.

Request Parameters:

Parameter	Type	Description
		The type of content to enumerate.
type	string	Allowed values: context_modules, assignments, quizzes, assessment_question_banks, discussion_topics, wiki_pages, context_external_tools, tool_profiles, announcements, calendar_events, rubrics, groups, learning_outcomes, attachments

Get asset id mapping ContentMigrationsController#asset\_id\_mapping

GET /api/v1/courses/:course\_id/content\_migrations/:id/asset\_id\_mapping

Scope: url:GET//api/v1/courses/:course\_id/content\_migrations/:id/asset\_id\_mapping

Given a complete course copy or blueprint import content migration, return a mapping of asset ids from the source course to the destination course that were copied in this migration or an earlier one with the same course pair and migration\_type (course copy or blueprint).

The returned object's keys are asset types as they appear in API URLs (announcements, assignments, discussion\_topics, files, module\_items, modules, pages, and quizzes). The values are a mapping from id in source course to id in destination course for objects of this type.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/content_migrations/<id>/asset_id_mapping \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "assignments": { "13": "740", "14": "741" },
```

```
}
  "discussion_topics": {"15": "743", "16": "744"}
}
```

Get current settings for account or course CspSettingsController#get\_csp\_settings

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

GET /api/v1/courses/:course\_id/csp\_settings

Scope: url:GET//api/v1/courses/:course\_id/csp\_settings

GET /api/v1/accounts/:account\_id/csp\_settings

Scope: url:GET//api/v1/accounts/:account\_id/csp\_settings

Update multiple modules in an account.

API response field:

- enabled  
Whether CSP is enabled.
- inherited  
Whether the current CSP settings are inherited from a parent account.
- settings\_locked  
Whether current CSP settings can be overridden by sub-accounts and courses.
- effective\_whitelist  
If enabled, lists the currently allowed domains (includes domains automatically allowed through external tools).
- tools\_whitelist  
(Account-only) Lists the automatically allowed domains with their respective external tools
- current\_account\_whitelist  
(Account-only) Lists the current list of domains explicitly allowed by this account. (Note: this list will not take effect unless CSP is explicitly enabled on this account)

Enable, disable, or clear explicit CSP setting CspSettingsController#set\_csp\_setting

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

PUT /api/v1/courses/:course\_id/csp\_settings

Scope: url:PUT//api/v1/courses/:course\_id/csp\_settings

PUT /api/v1/accounts/:account\_id/csp\_settings

Scope: url:PUT//api/v1/accounts/:account\_id/csp\_settings

Either explicitly sets CSP to be on or off for courses and sub-accounts, or clear the explicit settings to default to those set by a parent account

Note: If “inherited” and “settings\_locked” are both true for this account or course, then the CSP setting cannot be modified.

Request Parameters:

Parameter	Type	Description
status	Required string	If set to “enabled” for an account, CSP will be enabled for all its courses and sub-accounts (that have not explicitly enabled or disabled it), using the allowed domains set on this account. If set to “disabled”, CSP will be disabled for this account or course and for all sub-accounts that have not explicitly re-enabled it. If set to “inherited”, this account or course will reset to the default state where CSP settings are inherited from the first parent account to have them explicitly set.
		Allowed values: enabled, disabled, inherited

Lock or unlock current CSP settings for sub-accounts and courses CspSettingsController#set\_csp\_lock

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

PUT /api/v1/accounts/:account\_id/csp\_settings/lock

Scope: url:PUT//api/v1/accounts/:account\_id/csp\_settings/lock

Can only be set if CSP is explicitly enabled or disabled on this account (i.e. “inherited” is false).

Request Parameters:

Parameter	Type	Description
settings_locked	Required boolean	Whether sub-accounts and courses will be prevented from overriding settings inherited from this account.

Add an allowed domain to account CspSettingsController#add\_domain

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

POST /api/v1/accounts/:account\_id/csp\_settings/domains

Scope: url:POST//api/v1/accounts/:account\_id/csp\_settings/domains

Adds an allowed domain for the current account. Note: this will not take effect unless CSP is explicitly enabled on this account.

Request Parameters:

Parameter	Type	Description
domain	Required string	no description

Add multiple allowed domains to an account CspSettingsController#add\_multiple\_domains

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

POST /api/v1/accounts/:account\_id/csp\_settings/domains/batch\_create

Scope: url:POST//api/v1/accounts/:account\_id/csp\_settings/domains/batch\_create

Adds multiple allowed domains for the current account. Note: this will not take effect unless CSP is explicitly enabled on this account.

Request Parameters:

Parameter	Type	Description
domains	Required Array	no description

Remove a domain from account CspSettingsController#remove\_domain

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

DELETE /api/v1/accounts/:account\_id/csp\_settings/domains

Scope: url:DELETE//api/v1/accounts/:account\_id/csp\_settings/domains

Removes an allowed domain from the current account.

Request Parameters:

Parameter	Type	Description
domain	Required string	no description

Create a content share ContentSharesController#create

POST /api/v1/users/:user\_id/content\_shares

Scope: url: POST /api/v1/users/:user\_id/content\_shares

Share content directly between two or more users

Request Parameters:

Parameter	Type	Description
receiver_ids	Required Array	IDs of users to share the content with.
content_type	Required string	Type of content you are sharing. Allowed values: assignment, discussion_topic, page, quiz, module, module_item
content_id	Required integer	The id of the content that you are sharing

Example Request:

```
curl 'https://<canvas>/api/v1/users/self/content_shares \
-d 'content_type=assignment' \
-d 'content_id=1' \
-H 'Authorization: Bearer <tokens>' \
-X POST
```

Returns a ContentShare object

List content shares ContentSharesController#index

GET /api/v1/users/:user\_id/content\_shares/sent

Scope: url: GET /api/v1/users/:user\_id/content\_shares/sent

GET /api/v1/users/:user\_id/content\_shares/received

Scope: url: GET /api/v1/users/:user\_id/content\_shares/received

Return a paginated list of content shares a user has sent or received. Use self as the user\_id to retrieve your own content shares. Only linked observers and administrators may view other users' content shares.

Example Request:

```
curl 'https://<canvas>/api/v1/users/self/content_shares/received'
```

Returns a list of ContentShare objects

Get unread shares count ContentSharesController#unread\_count

GET /api/v1/users/:user\_id/content\_shares/unread\_count

Scope: url: GET /api/v1/users/:user\_id/content\_shares/unread\_count

Return the number of content shares a user has received that have not yet been read. Use self as the user\_id to retrieve your own content shares. Only linked observers and administrators may view other users' content shares.

Example Request:

```
curl 'https://<canvas>/api/v1/users/self/content_shares/unread_count'
```

Get content share ContentSharesController#show

GET /api/v1/users/:user\_id/content\_shares/:id

Scope: url: GET /api/v1/users/:user\_id/content\_shares/:id

Return information about a single content share. You may use self as the user\_id to retrieve your own content share.

Example Request:

```
curl 'https://<canvas>/api/v1/users/self/content_shares/123'
```

Returns a ContentShare object

Remove content share ContentSharesController#destroy

DELETE /api/v1/users/:user\_id/content\_shares/:id

Scope: url: DELETE /api/v1/users/:user\_id/content\_shares/:id

Remove a content share from your list. Use self as the user\_id. Note that this endpoint does not delete other users' copies of the content share.

Example Request:

```
curl -X DELETE 'https://<canvas>/api/v1/users/self/content_shares/123'
```

Add users to content share ContentSharesController#add\_users

POST /api/v1/users/:user\_id/content\_shares/:id/add\_users

Scope: url: POST /api/v1/users/:user\_id/content\_shares/:id/add\_users

Send a previously created content share to additional users

Request Parameters:

Parameter	Type	Description
receiver_ids	Array	IDs of users to share the content with.

Example Request:

```
curl -X POST 'https://<canvas>/api/v1/users/self/content_shares/123/add_users?receiver_ids[]=789'
```

Returns a ContentShare object

Update a content share ContentSharesController#update

PUT /api/v1/users/:user\_id/content\_shares/:id

Scope: url: PUT /api/v1/users/:user\_id/content\_shares/:id

Mark a content share read or unread

Request Parameters:

Parameter	Type	Description
-----------	------	-------------



Parameter	Type	Description
read_state	string	Read state for the content share
		Allowed values: read, unread

Example Request:

```
curl -X PUT 'https://<canvas>/api/v1/users/self/content_shares/123?read_state=read'
```

Returns a [ContentShare](#) object

List conversations [ConversationsController#index](#)

GET /api/v1/conversations

Scope: url:GET//api/v1/conversations

Returns the paginated list of conversations for the current user, most recent ones first.

Request Parameters:

Parameter	Type	Description
scope	string	When set, only return conversations of the specified type. For example, set to “unread” to return only conversations that haven’t been read. The default behavior is to return all non-archived conversations (i.e. read and unread).
		Allowed values: unread, starred, archived
filter[]	string	When set, only return conversations for the specified courses, groups or users. The id should be prefixed with its type, e.g. “user_123” or “course_456”. Can be an array (by setting “filter[]”) or single value (by setting “filter”)
filter_mode	string	When filter[] contains multiple filters, combine them with this mode, filtering conversations that at have at least all of the contexts (“and”) or at least one of the contexts (“or”) Allowed values: and, or, default or
interleave_submissions	boolean	(Obsolete) Submissions are no longer linked to conversations. This parameter is ignored.
include_all_conversation_ids	boolean	Default is false. If true, the top-level element of the response will be an object rather than an array, and will have the keys “conversations” which will contain the paged conversation data, and “conversation_ids” which will contain the ids of all conversations under this scope/filter in the same order.
include[]	string	“participant_avatars”
		Optionally include an “avatar_url” key for each user participating in the conversation Allowed values: participant_avatars

API response field:

- id  
The unique identifier for the conversation.
- subject  
The subject of the conversation.
- workflow\_state  
The current state of the conversation (read, unread or archived)
- last\_message  
A <=100 character preview from the most recent message
- last\_message\_at  
The timestamp of the latest message
- message\_count  
The number of messages in this conversation
- subscribed  
Indicates whether the user is actively subscribed to the conversation
- private  
Indicates whether this is a private conversation (i.e. audience of one)
- starred  
Whether the conversation is starred
- properties  
Additional conversation flags (last\_author, attachments, media\_objects). Each listed property means the flag is set to true (i.e. the current user is the most recent author, there are attachments, or there are media objects)
- audience  
Array of user ids who are involved in the conversation, ordered by participation level, then alphabetical. Excludes current user, unless this is a monologue.
- audience\_contexts  
Most relevant shared contexts (courses and groups) between current user and other participants. If there is only one participant, it will also include that user’s enrollment(s)/ membership type(s) in each course/group
- avatar\_url  
URL to appropriate icon for this conversation (custom, individual or group avatar, depending on audience)
- participants  
Array of users (id, name, full\_name) participating in the conversation. Includes current user. If ‘include[]=participant\_avatars’ was passed as an argument, each user in the array will also have an “avatar\_url” field
- visible  
Boolean, indicates whether the conversation is visible under the current scope and filter. This attribute is always true in the index API response, and is primarily useful in create/update responses so that you can know if the record should be displayed in the UI. The default scope is assumed, unless a scope or filter is passed to the create/update API call.

Example Response:

```
{
  {
    "id": 2,
    "subject": "conversations api example",
    "workflow_state": "unread",
    "last_message": "sure thing, here's the file.",
    "last_message_at": "2011-09-02T12:00:00Z",
    "message_count": 2,
    "subscribed": true,
    "private": true,
    "starred": false,
    "properties": [{"attachments"}],
    "audience": [2],
    "audience_contexts": {"courses": [{"1": ["StudentEnrollment"]}], "groups": {}},
    "avatar_url": "https://canvas.instructure.com/images/messages/avatar-group-50.png",
    "participants": [
      {"id": 1, "name": "Joe", "full_name": "Joe TA"},
      {"id": 2, "name": "Jane", "full_name": "Jane Teacher"}
    ],
    "visible": true,
    "context_name": "Canvas 101"
  }
}
```

Returns a list of [Conversation](#) objects

Create a conversation ConversationsController#create

POST /api/v1/conversations

Scope: url:POST|/api/v1/conversations

Create a new conversation with one or more recipients. If there is already an existing private conversation with the given recipients, it will be reused.

Request Parameters:

Parameter	Type	Description
recipients[]	Required string	An array of recipient ids. These may be user ids or course/group ids prefixed with "course_" or "group_" respectively, e.g. <code>recipients[]=1&amp;recipients=2&amp;recipients[]=course_3</code> . If the course/group has over 100 enrollments, 'bulk_message' and 'group_conversation' must be set to true.
subject	string	The subject of the conversation. This is ignored when reusing a conversation. Maximum length is 255 characters.
body	Required string	The message to be sent
force_new	boolean	Forces a new message to be created, even if there is an existing private conversation.
group_conversation	boolean	Defaults to false. When false, individual private conversations will be created with each recipient. If true, this will be a group conversation (i.e. all recipients may see all messages and replies). Must be set true if the number of recipients is over the set maximum (default is 100).
attachment_ids[]	string	An array of attachments ids. These must be files that have been previously uploaded to the sender's "conversation attachments" folder.
media_comment_id	string	Media comment id of an audio or video file to be associated with this message.
media_comment_type	string	Type of the associated media file Allowed values: audio, video
mode	string	Determines whether the messages will be created/sent synchronously or asynchronously. Defaults to sync, and this option is ignored if this is a group conversation or there is just one recipient (i.e. it must be a bulk private message). When sent async, the response will be an empty array (batch status can be queried via the <a href="#">batches API</a> ) Allowed values: sync, async
scope	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: unread, starred, archived
filter[]	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a>
filter_mode	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: and, or, default or
context_code	string	The course or group that is the context for this conversation. Same format as courses or groups in the recipients argument.

Get running batches ConversationsController#batches

GET /api/v1/conversations/batches

Scope: url:GET|/api/v1/conversations/batches

Returns any currently running conversation batches for the current user. Conversation batches are created when a bulk private message is sent asynchronously (see the mode argument to the [create API action](#)).

Example Response:

```
{
  {
    "id": 1,
    "subject": "conversations api example",
    "workflow_state": "created",
    "completion": 0.1234,
    "tags": [],
    "message": {
      {
        "id": 1,
        "created_at": "2011-09-02T10:00:00Z",
        "body": "Quick reminder, no class tomorrow",
        "author_id": 1,
        "generated": false,
        "media_comment": null,
        "forwarded_messages": [],
        "attachments": []
      }
    }
  }
}
```

Get a single conversation ConversationsController#show

GET /api/v1/conversations/:id

Scope: url:GET|/api/v1/conversations/:id

Returns information for a single conversation for the current user. Response includes all fields that are present in the list/index action as well as messages and extended participant information.

Request Parameters:

Parameter	Type	Description
interleave_submissions	boolean (Obsolete)	Submissions are no longer linked to conversations. This parameter is ignored.
scope	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: unread, starred, archived
filter[]	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a>
filter_mode	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: and, or, default or
auto_mark_as_read	boolean	Default true. If true, unread conversations will be automatically marked as read. This will default to false in a future API release, so clients should explicitly send true if that is the desired behavior.

API response field:

- participants
  - Array of relevant users. Includes current user. If there are forwarded messages in this conversation, the authors of those messages will also be included, even if they are not participating in this conversation. Fields include:
- messages
  - Array of messages, newest first. Fields include:
  - id
    - The unique identifier for the message
  - created\_at
    - The timestamp of the message

body
The actual message body
author_id
The id of the user who sent the message (see audience, participants)
generated
If true, indicates this is a system-generated message (e.g. "Bob added Alice to the conversation")
media_comment
Audio/video comment data for this message (if applicable). Fields include: display_name, content-type, media_id, media_type, url
forwarded_messages
If this message contains forwarded messages, they will be included here (same format as this list). Note that those messages may have forwarded messages of their own, etc.
attachments
Array of attachments for this message. Fields include: display_name, content-type, filename, url
<ul style="list-style-type: none"><li>submissions</li></ul>
(Obsolete) Array of assignment submissions having comments relevant to this conversation. Submissions are no longer linked to conversations. This field will always be nil or empty.

Example Response:

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "unread",
  "last_message": "sure thing, here's the file",
  "last_message_at": "2011-09-02T12:00:00-06:00",
  "message_count": 2,
  "subscribed": true,
  "private": true,
  "starred": true,
  "properties": [{"attachments":}],
  "audience": [2],
  "audience_contexts": {"courses": {"1": []}, "groups": {}},
  "avatar_url": "https://canvas.instructure.com/images/messages/avatar-50.png",
  "participants": [
    {"id": 1, "name": "Joe", "full_name": "Joe TA"},
    {"id": 2, "name": "Jane", "full_name": "Jane Teacher"},
    {"id": 3, "name": "Bob", "full_name": "Bob Student"}
  ],
  "messages": [
    {
      {
        "id": 3,
        "created_at": "2011-09-02T12:00:00Z",
        "body": "sure thing, here's the file",
        "author_id": 2,
        "generated": false,
        "media_comment": null,
        "forwarded_messages": [],
        "attachments": [{"id": 1, "display_name": "notes.doc", "uuid": "abcdefabcdefabcdefabcdef"}]
      },
      {
        "id": 2,
        "created_at": "2011-09-02T11:00:00Z",
        "body": "hey, bob didn't get the notes. do you have a copy i can give him?",
        "author_id": 2,
        "generated": false,
        "media_comment": null,
        "forwarded_messages": [
          {
            {
              "id": 1,
              "created_at": "2011-09-02T10:00:00Z",
              "body": "can i get a copy of the notes? i was out",
              "author_id": 3,
              "generated": false,
              "media_comment": null,
              "forwarded_messages": [],
              "attachments": []
            }
          ]
        },
        "attachments": []
      }
    ]
  ],
  "submissions": []
}
```

Edit a conversation [ConversationsController#update](#)

PUT /api/v1/conversations/:id

Scope: url:PUT /api/v1/conversations/:id

Updates attributes for a single conversation.

Request Parameters:

Parameter	Type	Description
conversation[workflow_state]	string	Change the state of this conversation Allowed values: read, unread, archived
conversation[subscribed]	boolean	Toggle the current user's subscription to the conversation (only valid for group conversations). If unsubscribed, the user will still have access to the latest messages, but the conversation won't be automatically flagged as unread, nor will it jump to the top of the inbox.
conversation[starred]	boolean	Toggle the starred state of the current user's view of the conversation.
scope	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: unread, starred, archived
filter[]	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a>
filter_mode	string	Used when generating "visible" in the API response. See the explanation under the <a href="#">index API action</a> Allowed values: and, or, default or

Example Response:

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "read",
  "last_message": "sure thing, here's the file",
  "last_message_at": "2011-09-02T12:00:00-06:00",
  "message_count": 2,
  "subscribed": true,
  "private": true,
  "starred": false,
  "properties": [{"attachments":}],
  "audience": [2],
  "audience_contexts": {"courses": {"1": []}, "groups": {}},
  "avatar_url": "https://canvas.instructure.com/images/messages/avatar-50.png",
  "participants": [{"id": 1, "name": "Joe", "full_name": "Joe TA"}]
}
```

Mark all as read [ConversationsController#mark\\_all\\_as\\_read](#)

POST /api/v1/conversations/mark\_all\_as\_read

Scope: url:POST /api/v1/conversations/mark\_all\_as\_read

Mark all conversations as read.

Delete a conversation [ConversationsController#destroy](#)

DELETE /api/v1/conversations/:id

**Scope:** url:DELETE/api/v1/conversations/:id

Delete this conversation and its messages. Note that this only deletes this user's view of the conversation.

Response includes same fields as UPDATE action

**Example Response:**

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "read",
  "last_message": null,
  "last_message_at": null,
  "message_count": 0,
  "subscribed": true,
  "private": true,
  "starred": false,
  "properties": []
}
```

**[Add recipients ConversationsController#add\\_recipients](#)**

**POST /api/v1/conversations/:id/add\_recipients**

**Scope:** url:POST/api/v1/conversations/:id/add\_recipients

Add recipients to an existing group conversation. Response is similar to the GET/show action, except that only includes the latest message (e.g. "joe was added to the conversation by bob")

**Request Parameters:**

Parameter	Type	Description
recipients[]	Required string	An array of recipient ids. These may be user ids or course/group ids prefixed with "course_" or "group_" respectively, e.g. <a href="#">recipients[]=1&amp;recipients=2&amp;recipients[]=course_3</a>

**Example Response:**

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "read",
  "last_message": "let's talk this over with jim",
  "last_message_at": "2011-09-02T12:00:00-06:00",
  "message_count": 2,
  "subscribed": true,
  "private": false,
  "starred": null,
  "properties": [],
  "audience": [2, 3, 4],
  "audience_contexts": {"courses": {"1": []}, "groups": {}},
  "avatar_url": "https://canvas.instructure.com/images/messages/avatar-group-50.png",
  "participants": [
    {"id": 1, "name": "Joe", "full_name": "Joe TA"},
    {"id": 2, "name": "Jane", "full_name": "Jane Teacher"},
    {"id": 3, "name": "Bob", "full_name": "Bob Student"},
    {"id": 4, "name": "Jim", "full_name": "Jim Admin"}
  ],
  "messages": [
    {
      "id": 4,
      "created_at": "2011-09-02T12:10:00Z",
      "body": "Jim was added to the conversation by Joe TA",
      "author_id": 1,
      "generated": true,
      "media_comment": null,
      "forwarded_messages": [],
      "attachments": []
    }
  ]
}
```

**[Add a message ConversationsController#add\\_message](#)**

**POST /api/v1/conversations/:id/add\_message**

**Scope:** url:POST/api/v1/conversations/:id/add\_message

Add a message to an existing conversation. Response is similar to the GET/show action, except that only includes the latest message (i.e. what we just sent)

An array of user ids. Defaults to all of the current conversation recipients. To explicitly send a message to no other recipients, this array should consist of the logged-in user id.

An array of message ids from this conversation to send to recipients of the new message. Recipients who already had a copy of included messages will not be affected.

**Request Parameters:**

Parameter	Type	Description
body	Required string	The message to be sent.
attachment_ids[]	string	An array of attachments ids. These must be files that have been previously uploaded to the sender's "conversation attachments" folder.
media_comment_id	string	Media comment id of an audio or video file to be associated with this message.
media_comment_type	string	Type of the associated media file. Allowed values: audio, video
recipients[]	string	no description
included_messages[]	string	no description

**Example Response:**

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "unread",
  "last_message": "let's talk this over with jim",
  "last_message_at": "2011-09-02T12:00:00-06:00",
  "message_count": 2,
  "subscribed": true,
  "private": false,
  "starred": null,
  "properties": [],
  "audience": [2, 3],
  "audience_contexts": {"courses": {"1": []}, "groups": {}},
  "avatar_url": "https://canvas.instructure.com/images/messages/avatar-group-50.png",
  "participants": [
    {"id": 1, "name": "Joe", "full_name": "Joe TA"},
    {"id": 2, "name": "Jane", "full_name": "Jane Teacher"},
    {"id": 3, "name": "Bob", "full_name": "Bob Student"}
  ],
  "messages": [
    {
      "id": 3,
      "created_at": "2011-09-02T12:00:00Z",
      "body": "let's talk this over with jim",
      "author_id": 2,
      "generated": false,
      "media_comment": null,
      "forwarded_messages": [],
      "attachments": []
    }
  ]
}
```

**[Delete a message ConversationsController#remove\\_messages](#)**

**POST /api/v1/conversations/:id/remove\_messages**

**Scope:** url:POST/api/v1/conversations/:id/remove\_messages

Delete messages from this conversation. Note that this only affects this user's view of the conversation. If all messages are deleted, the conversation will be as well (equivalent to DELETE)

**Request Parameters:**

Parameter	Type	Description
remove[]	Required string Array	of message ids to be deleted

Example Response:

```
{
  "id": 2,
  "subject": "conversations api example",
  "workflow_state": "read",
  "last_message": "sure thing, here's the file",
  "last_message_at": "2011-09-02T12:00:00-06:00",
  "message_count": 1,
  "subscribed": true,
  "private": true,
  "starred": null,
  "properties": [{"attachments"}]
}
```

Batch update conversations ConversationsController#batch\_update

PUT /api/v1/conversations

Scope: url:PUT//api/v1/conversations

Perform a change on a set of conversations. Operates asynchronously; use the [progress endpoint](#) to query the status of an operation.

Request Parameters:

Parameter	Type	Description
conversation_ids[]	Required string List	of conversations to update. Limited to 500 conversations.
event	Required string	The action to take on each conversation. Allowed values: mark_as_read, mark_as_unread, star, unstar, archive, destroy

Example Request:

```
curl https://<canvas>/api/v1/conversations \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'event=mark_as_read' \
-d 'conversation_ids[]=1' \
-d 'conversation_ids[]=2'
```

Returns a [Progress](#) object

Find recipients ConversationsController#find\_recipients

GET /api/v1/conversations/find\_recipients

Scope: url:GET//api/v1/conversations/find\_recipients

Deprecated, see the [Find recipients endpoint](#) in the Search API

Unread count ConversationsController#unread\_count

GET /api/v1/conversations/unread\_count

Scope: url:GET//api/v1/conversations/unread\_count

Get the number of unread conversations for the current user

Example Response:

```
{'unread_count': '7'}
```

Query by course. CourseAuditApiController#for\_course

GET /api/v1/audit/course/courses/:course\_id

Scope: url:GET//api/v1/audit/course/courses/:course\_id

List course change events for a given course.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [CourseEvent](#) objects

Query by account. CourseAuditApiController#for\_account

GET /api/v1/audit/course/accounts/:account\_id

Scope: url:GET//api/v1/audit/course/accounts/:account\_id

List course change events for a given account.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [CourseEvent](#) objects

Show a Course pace CoursePacesController#api\_show

GET /api/v1/courses/:course\_id/course\_pacing/:id

Scope: url:GET//api/v1/courses/:course\_id/course\_pacing/:id

Returns a course pace for the course and pace id provided

Request Parameters:

Parameter	Type	Description
course_id	Required integer	The id of the course
course_pace_id	Required integer	The id of the course_pace

Example Request:

```
curl https://<canvas>/api/v1/courses/1/course_pacing/1 \
-H 'Authorization: Bearer <token>'
```

Returns a [CoursePace](#) object

Create a Course pace CoursePacesController#create

POST /api/v1/courses/:course\_id/course\_pacing

Scope: url:POST|/api/v1/courses/:course\_id/course\_pacing

Update a Course pace CoursePacesController#update

PUT /api/v1/courses/:course\_id/course\_pacing/:id

Scope: url:PUT|/api/v1/courses/:course\_id/course\_pacing/:id

Returns the updated course pace

Request Parameters:

Parameter	Type	Description
course_id	Required integer	The id of the course
course_pace_id	Required integer	The id of the course pace
end_date	Datetime	End date of the course pace
exclude_weekends	boolean	Course pace dates excludes weekends if true
selected_days_to_skip	string	Array<String> Course pace dates excludes weekends if true
hard_end_dates	boolean	Course pace uess hard end dates if true
workflow_state	string	The state of the course pace
course_pace_module_item_attributes[]	string	Module Items attributes

Example Request:

```
curl https://<canvas>/api/v1/courses/1/course_pacing/1 \
-X PUT \
-H 'Authorization: Bearer <token>'
```

Returns a CoursePace object

Delete a Course pace CoursePacesController#destroy

DELETE /api/v1/courses/:course\_id/course\_pacing/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/course\_pacing/:id

Returns the updated course pace

Request Parameters:

Parameter	Type	Description
course_id	Required integer	The id of the course
course_pace_id	Required integer	The id of the course_pace

Example Request:

```
curl https://<canvas>/api/v1/courses/1/course_pacing/1 \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns a CoursePace object

Set extensions for student quiz submissions Quizzes::CourseQuizExtensionsController#create

POST /api/v1/courses/:course\_id/quiz\_extensions

Scope: url:POST|/api/v1/courses/:course\_id/quiz\_extensions

Responses

- 200 OK if the request was successful
- 403 Forbidden if you are not allowed to extend quizzes for this course

Request Parameters:

Parameter	Type	Description
user_id	Required integer	The ID of the user we want to add quiz extensions for.
extra_attempts	integer	Number of times the student is allowed to re-take the quiz over the multiple-attempt limit. This is limited to 1000 attempts or less.
extra_time	integer	The number of extra minutes to allow for all attempts. This will add to the existing time limit on the submission. This is limited to 10080 minutes (1 week)
manually_unlocked	boolean	Allow the student to take the quiz even if it's locked for everyone else.
extend_from_now	integer	The number of minutes to extend the quiz from the current time. This is mutually exclusive to extend_from_end_at. This is limited to 1440 minutes (24 hours)
extend_from_end_at	integer	The number of minutes to extend the quiz beyond the quiz's current ending time. This is mutually exclusive to extend_from_now. This is limited to 1440 minutes (24 hours)

Example Request:

```
{
  "quiz_extensions": [{
    "user_id": 3,
    "extra_attempts": 2,
    "extra_time": 20,
    "manually_unlocked": true
  },{
    "user_id": 2,
    "extra_attempts": 2,
    "extra_time": 20,
    "manually_unlocked": false
  }]
}

{
  "quiz_extensions": [{
    "user_id": 3,
    "extend_from_now": 20
  }]
}
```

Example Response:

```
{
  "quiz_extensions": [QuizExtension]
}
```

List your courses CoursesController#index

GET /api/v1/courses

Scope: url::GET//api/v1/courses

Returns the paginated list of active courses for the current user.

Request Parameters:

Parameter	Type	Description
enrollment_type	string	When set, only return courses where the user is enrolled as this type. For example, set to "teacher" to return only courses where the user is enrolled as a Teacher. This argument is ignored if enrollment_role is given. Allowed values: teacher, student, ta, observer, designer
enrollment_role	string	Deprecated When set, only return courses where the user is enrolled with the specified course-level role. This can be a role created with the <a href="#">Add Role API</a> or a base role type of 'StudentEnrollment', 'TeacherEnrollment', 'TaEnrollment', 'ObserverEnrollment', or 'DesignerEnrollment'.
enrollment_role_id	integer	When set, only return courses where the user is enrolled with the specified course-level role. This can be a role created with the <a href="#">Add Role API</a> or a built_in role type of 'StudentEnrollment', 'TeacherEnrollment', 'TaEnrollment', 'ObserverEnrollment', or 'DesignerEnrollment'.
enrollment_state	string	When set, only return courses where the user has an enrollment with the given state. This will respect section/course/term date overrides. Allowed values: active, invited_or_pending, completed
exclude_blueprint_courses	boolean	When set, only return courses that are not configured as blueprint courses.

include[]	string	<ul style="list-style-type: none"><li>"needs_grading_count": Optional information to include with each Course. When needs_grading_count is given, and the current user has grading rights, the total number of submissions needing grading for all assignments is returned.</li><li>"syllabus_body": Optional information to include with each Course. When syllabus_body is given the user-generated html for the course syllabus is returned.</li><li>"public_description": Optional information to include with each Course. When public_description is given the user-generated text for the course public description is returned.</li><li>"total_scores": Optional information to include with each Course. When total_scores is given, any student enrollments will also include the fields 'computed_current_score', 'computed_final_score', 'computed_current_grade', and 'computed_final_grade', as well as (if the user has permission) 'unposted_current_score', 'unposted_final_score', 'unposted_current_grade', and 'unposted_final_grade' (see Enrollment documentation for more information on these fields). This argument is ignored if the course is configured to hide final grades.</li><li>"current_grading_period_scores": Optional information to include with each Course. When current_grading_period_scores is given and total_scores is given, any student enrollments will also include the fields 'has_grading_periods', 'totals_for_all_grading_periods_option', 'current_grading_period_title', 'current_grading_period_id', 'current_period_computed_current_score', 'current_period_computed_final_score', 'current_period_computed_current_grade', and 'current_period_computed_final_grade', as well as (if the user has permission) 'current_period_unposted_current_score', 'current_period_unposted_final_score', 'current_period_unposted_current_grade', and 'current_period_unposted_final_grade' (see Enrollment documentation for more information on these fields). In addition, when this argument is passed, the course will have a 'has_grading_periods' attribute on it. This argument is ignored if the total_scores argument is not included. If the course is configured to hide final grades, the following fields are not returned: 'totals_for_all_grading_periods_option', 'current_period_computed_current_score', 'current_period_computed_final_score', 'current_period_computed_current_grade', 'current_period_computed_final_grade', 'current_period_unposted_current_score', 'current_period_unposted_final_score', 'current_period_unposted_current_grade', and 'current_period_unposted_final_grade'</li><li>"grading_periods": Optional information to include with each Course. When grading_periods is given, a list of the grading periods associated with each course is returned.</li><li>"term": Optional information to include with each Course. When term is given, the information for the enrollment term for each course is returned.</li><li>"account": Optional information to include with each Course. When account is given, the account json for each course is returned.</li><li>"course_progress": Optional information to include with each Course. When course_progress is given, each course will include a 'course_progress' object with the fields: 'requirement_count', an integer specifying the total number of requirements in the course, 'requirement_completed_count', an integer specifying the total number of requirements in this course that have been completed, and 'next_requirement_url', a string url to the next requirement item, and 'completed_at', the date the course was completed (null if incomplete). 'next_requirement_url' will be null if all requirements have been completed or the current module does not require sequential progress. "course_progress" will return an error message if the course is not module based or the user is not enrolled as a student in the course.</li><li>"sections": Section enrollment information to include with each Course. Returns an array of hashes containing the section ID (id), section name (name), start and end dates (start_at, end_at), as well as the enrollment type (enrollment_role, e.g. 'StudentEnrollment').</li><li>"storage_quota_used_mb": The amount of storage space used by the files in this course</li><li>"total_students": Optional information to include with each Course. Returns an integer for the total amount of active and invited students.</li><li>"passback_status": Include the grade passback status</li><li>"favorites": Optional information to include with each Course. Indicates if the user has marked the course as a favorite course.</li><li>"teachers": Teacher information to include with each Course. Returns an array of hashes containing the <a href="#">UserDisplay</a> information for each teacher in the course.</li><li>"observed_users": Optional information to include with each Course. Will include data for observed users if the current user has an observer enrollment.</li><li>"tabs": Optional information to include with each Course. Will include the list of tabs configured for each course. See the <a href="#">List available tabs API</a> for more information.</li><li>"course_image": Optional information to include with each Course. Returns course image url if a course image has been set.</li><li>"banner_image": Optional information to include with each Course. Returns course banner image url if the course is a Canvas for Elementary subject and a banner image has been set.</li><li>"concluded": Optional information to include with each Course. Indicates whether the course has been concluded, taking course and term dates into account.</li><li>"post_manually": Optional information to include with each Course. Returns true if the course post policy is set to Manually post grades. Returns false if the the course post policy is set to Automatically post grades.</li></ul>
		Allowed values: needs_grading_count, syllabus_body, public_description, total_scores, current_grading_period_scores, grading_periods, term, account, course_progress, sections, storage_quota_used_mb, total_students, passback_status, favorites, teachers, observed_users, course_image, banner_image, concluded, post_manually
state[]	string	If set, only return courses that are in the given state(s). By default, "available" is returned for students and observers, and anything except "deleted", for all other enrollment types Allowed values: unpublished, available, completed, deleted

Returns a list of [Course](#) objects

List courses for a user CoursesController#user\_index

GET /api/v1/users/user\_id/courses

Scope: url::GET//api/v1/users/:user\_id/courses

Returns a paginated list of active courses for this user. To view the course list for a user other than yourself, you must be either an observer of that user or an administrator.

Request Parameters:

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"needs_grading_count": Optional information to include with each Course. When needs_grading_count is given, and the current user has grading rights, the total number of submissions needing grading for all assignments is returned.</li><li>"syllabus_body": Optional information to include with each Course. When syllabus_body is given the user-generated html for the course syllabus is returned.</li><li>"public_description": Optional information to include with each Course. When public_description is given the user-generated text for the course public description is returned.</li><li>"total_scores": Optional information to include with each Course. When total_scores is given, any student enrollments will also include the fields 'computed_current_score', 'computed_final_score', 'computed_current_grade', and 'computed_final_grade' (see Enrollment documentation for more information on these fields). This argument is ignored if the course is configured to hide final grades.</li><li>"current_grading_period_scores": Optional information to include with each Course. When current_grading_period_scores is given and total_scores is given, any student enrollments will also include the fields 'has_grading_periods', 'totals_for_all_grading_periods_option', 'current_grading_period_title', 'current_grading_period_id', 'current_period_computed_current_score', 'current_period_computed_final_score', 'current_period_computed_current_grade', and 'current_period_computed_final_grade', as well as (if the user has permission) 'current_period_unposted_current_score', 'current_period_unposted_final_score', 'current_period_unposted_current_grade', and 'current_period_unposted_final_grade' (see Enrollment documentation for more information on these fields). In addition, when this argument is passed, the course will have a 'has_grading_periods' attribute on it. This argument is ignored if the course is configured to hide final grades or if the total_scores argument is not included.</li><li>"grading_periods": Optional information to include with each Course. When grading_periods is given, a list of the grading periods associated with each course is returned.</li><li>"term": Optional information to include with each Course. When term is given, the information for the enrollment term for each course is returned.</li></ul>

Parameter	Type	Description
		<ul style="list-style-type: none"><li>“account”: Optional information to include with each Course. When account is given, the account json for each course is returned.</li><li>“course_progress”: Optional information to include with each Course. When course_progress is given, each course will include a 'course_progress' object with the fields: 'requirement_count', an integer specifying the total number of requirements in the course, 'requirement_completed_count', an integer specifying the total number of requirements in this course that have been completed, and 'next_requirement_url', a string url to the next requirement item, and 'completed_at', the date the course was completed (null if incomplete). 'next_requirement_url' will be null if all requirements have been completed or the current module does not require sequential progress. "course_progress" will return an error message if the course is not module based or the user is not enrolled as a student in the course.</li><li>“sections”: Section enrollment information to include with each Course. Returns an array of hashes containing the section ID (id), section name (name), start and end dates (start_at, end_at), as well as the enrollment type (enrollment_role, e.g. 'StudentEnrollment').</li><li>“storage_quota_used_mb”: The amount of storage space used by the files in this course</li><li>“total_students”: Optional information to include with each Course. Returns an integer for the total amount of active and invited students.</li><li>“passback_status”: Include the grade passback_status</li><li>“favorites”: Optional information to include with each Course. Indicates if the user has marked the course as a favorite course.</li><li>“teachers”: Teacher information to include with each Course. Returns an array of hashes containing the <a href="#">UserDisplay</a> information for each teacher in the course.</li><li>“observed_users”: Optional information to include with each Course. Will include data for observed users if the current user has an observer enrollment.</li><li>“tabs”: Optional information to include with each Course. Will include the list of tabs configured for each course. See the <a href="#">List available tabs API</a> for more information.</li><li>“course_image”: Optional information to include with each Course. Returns course image url if a course image has been set.</li><li>“banner_image”: Optional information to include with each Course. Returns course banner image url if the course is a Canvas for Elementary subject and a banner image has been set.</li><li>“concluded”: Optional information to include with each Course. Indicates whether the course has been concluded, taking course and term dates into account.</li><li>“post_manually”: Optional information to include with each Course. Returns true if the course post policy is set to “Manually”. Returns false if the the course post policy is set to “Automatically”.</li></ul> <p>Allowed values: needs_grading_count, syllabus_body, public_description, total_scores, current_grading_period_scores, grading_periods, term, account, course_progress, sections, storage_quota_used_mb, total_students, passback_status, favorites, teachers, observed_users, course_image, banner_image, concluded, post_manually</p>
state[]	string	If set, only return courses that are in the given state(s). By default, “available” is returned for students and observers, and anything except “deleted”, for all other enrollment types Allowed values: unpublished, available, completed, deleted
enrollment_state	string	When set, only return courses where the user has an enrollment with the given state. This will respect section/course/term date overrides. Allowed values: active, invited_or_pending, completed
homeroom	boolean	If set, only return homeroom courses.
account_id	string	If set, only include courses associated with this account
Returns a list of <a href="#">Course</a> objects		

Get user progress [CoursesController#user\\_progress](#)

GET /api/v1/courses/:course\_id/users/:user\_id/progress

Scope: url: GET|/api/v1/courses/:course\_id/users/:user\_id/progress

Return progress information for the user and course

You can supply self as the user\_id to query your own progress in a course. To query another user's progress, you must be a teacher in the course, an administrator, or a linked observer of the user.

Returns a [CourseProgress](#) object

Create a new course [CoursesController#create](#)

POST /api/v1/accounts/:account\_id/courses

Scope: url: POST|/api/v1/accounts/:account\_id/courses

Create a new course

Request Parameters:

Parameter	Type	Description
course[name]	string	The name of the course. If omitted, the course will be named “Unnamed Course.”
course[course_code]	string	The course code for the course.
course[start_at]	DateTime	Course start date in ISO8601 format, e.g. 2011-01-01T01:00Z This value is ignored unless 'restrict_enrollments_to_course_dates' is set to true.
course[end_at]	DateTime	Course end date in ISO8601 format. e.g. 2011-01-01T01:00Z This value is ignored unless 'restrict_enrollments_to_course_dates' is set to true.
course[license]	string	<p>The name of the licensing. Should be one of the following abbreviations (a descriptive name is included in parenthesis for reference):</p> <ul style="list-style-type: none"><li>‘private’ (Private Copyrighted)</li><li>‘cc_by_nc_nd’ (CC Attribution Non-Commercial No Derivatives)</li><li>‘cc_by_nc_sa’ (CC Attribution Non-Commercial Share Alike)</li><li>‘cc_by_nc’ (CC Attribution Non-Commercial)</li><li>‘cc_by_nd’ (CC Attribution No Derivatives)</li><li>‘cc_by_sa’ (CC Attribution Share Alike)</li><li>‘cc_by’ (CC Attribution)</li><li>‘public_domain’ (Public Domain).</li></ul>
course[is_public]	boolean	Set to true if course is public to both authenticated and unauthenticated users.
course[is_public_to_auth_users]	boolean	Set to true if course is public only to authenticated users.
course[public_syllabus]	boolean	Set to true to make the course syllabus public.
course[public_syllabus_to_auth]	boolean	Set to true to make the course syllabus public for authenticated users.
course[public_description]	string	A publicly visible description of the course.
course[allow_student_wiki_edits]	boolean	If true, students will be able to modify the course wiki.
course[allow_wiki_comments]	boolean	If true, course members will be able to comment on wiki pages.
course[allow_student_forum_attachments]	boolean	If true, students can attach files to forum posts.
course[open_enrollment]	boolean	Set to true if the course is open enrollment.
course[self_enrollment]	boolean	Set to true if the course is self enrollment.



Parameter	Type	Description
course[restrict_enrollments_to_course_dates]	boolean	Set to true to restrict user enrollments to the start and end dates of the course. This value must be set to true in order to specify a course start date and/or end date.
course[term_id]	string	The unique ID of the term to create to course in.
course[sis_course_id]	string	The unique SIS identifier.
course[integration_id]	string	The unique Integration identifier.
course[hide_final_grades]	boolean	If this option is set to true, the totals in student grades summary will be hidden.
course[apply_assignment_group_weights]	boolean	Set to true to weight final grade based on assignment groups percentages.
course[time_zone]	string	The time zone for the course. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
offer	boolean	If this option is set to true, the course will be available to students immediately.
enroll_me	boolean	Set to true to enroll the current user as the teacher.
		The type of page that users will see when they first visit the course <ul style="list-style-type: none"><li>'feed' Recent Activity Dashboard</li><li>'modules' Course Modules/Sections Page</li><li>'assignments' Course Assignments List</li><li>'syllabus' Course Syllabus Page</li></ul> other types may be added in the future Allowed values: feed, wiki, modules, syllabus, assignments
course[default_view]	string	
course[syllabus_body]	string	The syllabus body for the course
course[grading_standard_id]	integer	The grading standard id to set for the course. If no value is provided for this argument the current grading_standard will be un-set from this course.
course[grade_passback_setting]	string	Optional. The grade_passback_setting for the course. Only 'nightly_sync', 'disabled', and " are allowed
course[course_format]	string	Optional. Specifies the format of the course. (Should be 'on_campus', 'online', or 'blended')
course[post_manually]	boolean	Default is false. When true, all grades in the course must be posted manually, and will not be automatically posted. When false, all grades in the course will be automatically posted.
enable_sis_reactivation	boolean	When true, will first try to re-activate a deleted course with matching sis_course_id if possible.

Returns a [Course](#) object

[Upload a file CoursesController#create\\_file](#)

**POST** /api/v1/courses/:course\_id/files

**Scope:** url:POST|/api/v1/courses/:course\_id/files

Upload a file to the course.

This API endpoint is the first step in uploading a file to a course. See the [File Upload Documentation](#) for details on the file upload workflow.

Only those with the “Manage Files” permission on a course can upload files to the course. By default, this is Teachers, TAs and Designers.

[List students CoursesController#students](#)

**GET** /api/v1/courses/:course\_id/students

**Scope:** url:GET|/api/v1/courses/:course\_id/students

Returns the paginated list of students enrolled in this course.

DEPRECATED: Please use the [course\\_users](#) endpoint and pass “student” as the enrollment\_type.

Returns a list of [User](#) objects

[List users in course CoursesController#users](#)

**GET** /api/v1/courses/:course\_id/users

**Scope:** url:GET|/api/v1/courses/:course\_id/users

**GET** /api/v1/courses/:course\_id/search\_users

**Scope:** url:GET|/api/v1/courses/:course\_id/search\_users

Returns the paginated list of users in this course. And optionally the user's enrollments in the course.

**Request Parameters:**

Parameter	Type	Description
search_term	string	The partial name or full ID of the users to match and return in the results list.
sort	string	When set, sort the results of the search based on the given field. Allowed values: username, last_login, email, sis_id
enrollment_type[]	string	When set, only return users where the user is enrolled as this type. “student_view” implies include[]=test_student. This argument is ignored if enrollment_role is given. Allowed values: teacher, student, student_view, ta, observer, designer
enrollment_role	string	Deprecated When set, only return users enrolled with the specified course-level role. This can be a role created with the <a href="#">Add Role API</a> or a base role type of ‘StudentEnrollment’, ‘TeacherEnrollment’, ‘TaEnrollment’, ‘ObserverEnrollment’, or ‘DesignerEnrollment’.
enrollment_role_id	integer	When set, only return courses where the user is enrolled with the specified course-level role. This can be a role created with the <a href="#">Add Role API</a> or a built_in role id with type ‘StudentEnrollment’, ‘TeacherEnrollment’, ‘TaEnrollment’, ‘ObserverEnrollment’, or ‘DesignerEnrollment’.
include[]	string	<ul style="list-style-type: none"><li>“enrollments”: Optionally include with each Course the user's current and invited enrollments. If the user is enrolled as a student, and the account has permission to manage or view all grades, each enrollment will include a 'grades' key with 'current_score', 'final_score', 'current_grade' and 'final_grade' values.</li><li>“locked”: Optionally include whether an enrollment is locked.</li><li>“avatar_url”: Optionally include avatar_url.</li><li>“bio”: Optionally include each user's bio.</li><li>“test_student”: Optionally include the course's Test Student,</li></ul>

Parameter	Type	Description
		if present. Default is to not include Test Student. <ul style="list-style-type: none"><li>“custom_links”: Optionally include plugin-supplied custom links for each student, such as analytics information</li><li>“current_grading_period_scores”: if enrollments is included as</li></ul> well as this directive, the scores returned in the enrollment will be for the current grading period if there is one. A 'grading_period_id' value will also be included with the scores. if grading_period_id is nil there is no current grading period and the score is a total score. <ul style="list-style-type: none"><li>“uuid”: Optionally include the users uuid</li></ul> Allowed values: enrollments, locked, avatar_url, test_student, bio, custom_links, current_grading_period_scores, uuid
user_id	string	If this parameter is given and it corresponds to a user in the course, the page parameter will be ignored and the page containing the specified user will be returned instead.
user_ids[]	integer	If included, the course users set will only include users with IDs specified by the param. Note: this will not work in conjunction with the “user_id” argument but multiple user_ids can be included.
enrollment_state[]	string	When set, only return users where the enrollment workflow state is of one of the given types. “active” and “invited” enrollments are returned by default. Allowed values: active, invited, rejected, completed, inactive

Returns a list of [User](#) objects

[List recently logged in students CoursesController#recent\\_students](#)

**GET /api/v1/courses/:course\_id/recent\_students**

**Scope:** url:GET|/api/v1/courses/:course\_id/recent\_students

Returns the paginated list of users in this course, ordered by how recently they have logged in. The records include the 'last\_login' field which contains a timestamp of the last time that user logged into canvas. The querying user must have the 'View usage reports' permission.

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/<course_id>/recent_users
```

Returns a list of [User](#) objects

[Get single user CoursesController#user](#)

**GET /api/v1/courses/:course\_id/users:id**

**Scope:** url:GET|/api/v1/courses/:course\_id/users/:id

Return information on a single user.

Accepts the same include[] parameters as the :users: action, and returns a single user with the same fields as that action.

Returns an [User](#) object

[Search for content share users CoursesController#content\\_share\\_users](#)

**GET /api/v1/courses/:course\_id/content\_share\_users**

**Scope:** url:GET|/api/v1/courses/:course\_id/content\_share\_users

Returns a paginated list of users you can share content with. Requires the content share feature and the user must have the manage content permission for the course.

**Request Parameters:**

Parameter	Type	Description
search_term	Required string	Term used to find users. Will search available share users with the search term in their name.

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/<course_id>/content_share_users \
-d 'search_term=smith'
```

Returns a list of [User](#) objects

[Preview processed html CoursesController#preview\\_html](#)

**POST /api/v1/courses/:course\_id/preview\_html**

**Scope:** url:POST|/api/v1/courses/:course\_id/preview\_html

Preview html content processed for this course

**Request Parameters:**

Parameter	Type	Description
html	string	The html content to process

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/preview_html \
-F 'html=<p>badhtml</badhtml>processed html</p>' \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "html": "<p>processed html</p>"
}
```

[Course activity stream CoursesController#activity\\_stream](#)

**GET /api/v1/courses/:course\_id/activity\_stream**

**Scope:** url:GET|/api/v1/courses/:course\_id/activity\_stream

Returns the current user's course-specific activity stream, paginated.

For full documentation, see the API documentation for the user activity stream, in the user api.

[Course activity stream summary CoursesController#activity\\_stream\\_summary](#)

**GET /api/v1/courses/:course\_id/activity\_stream/summary**

**Scope:** url:GET|/api/v1/courses/:course\_id/activity\_stream/summary

Returns a summary of the current user's course-specific activity stream.

For full documentation, see the API documentation for the user activity stream summary, in the user api.

[Course TODO items CoursesController#todo\\_items](#)

**GET /api/v1/courses/:course\_id/todo**

**Scope:** url:GET|/api/v1/courses/:course\_id/todo

Returns the current user's course-specific todo items.

For full documentation, see the API documentation for the user todo items, in the user api.

Delete/Conclude a course CoursesController#destroy

DELETE /api/v1/courses/:id

Scope: url:DELETE|/api/v1/courses/:id

Delete or conclude an existing course

Request Parameters:

Parameter	Type	Description
event	Required string	The action to take on the course. Allowed values: delete, conclude

Example Response:

```
{ "delete": "true" }
```

Get course settings CoursesController#api\_settings

GET /api/v1/courses/:course\_id/settings

Scope: url:GET|/api/v1/courses/:course\_id/settings

Returns some of a course's settings.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/settings \
-X GET \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "allow_student_discussion_topics": true,
  "allow_student_forum_attachments": false,
  "allow_student_discussion_editing": true,
  "grading_standard_enabled": true,
  "grading_standard_id": 137,
  "allow_student_organized_groups": true,
  "hide_final_grades": false,
  "hide_distribution_graphs": false,
  "hide_sections_on_course_users_page": false,
  "lock_all_announcements": true,
  "usage_rights_required": false,
  "homeroom_course": false,
  "default_due_time": "23:59:59",
  "conditional_release": false
}
```

Update course settings CoursesController#update\_settings

PUT /api/v1/courses/:course\_id/settings

Scope: url:PUT|/api/v1/courses/:course\_id/settings

Can update the following course settings:

Request Parameters:

Parameter	Type	Description
allow_final_grade_override	boolean	Let student final grades for a grading period or the total grades for the course be overridden
allow_student_discussion_topics	boolean	Let students create discussion topics
allow_student_forum_attachments	boolean	Let students attach files to discussions
allow_student_discussion_editing	boolean	Let students edit or delete their own discussion replies
allow_student_organized_groups	boolean	Let students organize their own groups
allow_student_discussion_reporting	boolean	Let students report offensive discussion content
allow_student_anonymous_discussion_topics	boolean	Let students create anonymous discussion topics
filter_speed_grader_by_student_group	boolean	Filter SpeedGrader to only the selected student group
hide_final_grades	boolean	Hide totals in student grades summary
hide_distribution_graphs	boolean	Hide grade distribution graphs from students
hide_sections_on_course_users_page	boolean	Disallow students from viewing students in sections they do not belong to
lock_all_announcements	boolean	Disable comments on announcements
usage_rights_required	boolean	Copyright and license information must be provided for files before they are published.
restrict_student_past_view	boolean	Restrict students from viewing courses after end date
restrict_student_future_view	boolean	Restrict students from viewing courses before start date
show_announcements_on_home_page	boolean	Show the most recent announcements on the Course home page (if a Wiki, defaults to five announcements, configurable via home_page_announcement_limit). Canvas for Elementary subjects ignore this setting.
home_page_announcement_limit	integer	Limit the number of announcements on the home page if enabled via show_announcements_on_home_page
syllabus_course_summary	boolean	Show the course summary (list of assignments and calendar events) on the syllabus page. Default is true.
default_due_time	string	Set the default due time for assignments. This is the time that will be pre-selected in the Canvas user interface when setting a due date for an assignment. It does not change when any existing assignment is due. It should be given in 24-hour HH:MM:SS format. The default is "23:59:59". Use "inherit" to inherit the account setting.
conditional_release	boolean	Enable or disable individual learning paths for students based on assessment

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/settings \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'allow_student_discussion_topics=false'
```

Return test student for course CoursesController#student\_view\_student

GET /api/v1/courses/:course\_id/student\_view\_student

**Scope:** url:GET|/api/v1/courses/:course\_id/student\_view\_student

Returns information for a test student in this course. Creates a test student if one does not already exist for the course. The caller must have permission to access the course's student view.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/student_view_student \
-X GET \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

[Get a single course CoursesController#show](#)

GET /api/v1/courses/:id

**Scope:** url:GET|/api/v1/courses/:id

GET /api/v1/accounts/:account\_id/courses/:id

**Scope:** url:GET|/api/v1/accounts/:account\_id/courses/:id

Return information on a single course.

Accepts the same include[] parameters as the list action plus:

Request Parameters:

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>“all_courses”: Also search recently deleted courses.</li><li>“permissions”: Include permissions the current user has for the course.</li><li>“observed_users”: Include observed users in the enrollments</li><li>“course_image”: Include course image url if a course image has been set</li><li>“banner_image”: Include course banner image url if the course is a Canvas for Elementary subject and a banner image has been set</li><li>“concluded”: Optional information to include with Course. Indicates whether the course has been concluded, taking course and term dates into account.</li><li>“lti_context_id”: Include course LTI tool id.</li><li>“post_manually”: Include course post policy. If the post policy is manually post grades, the value will be true. If the post policy is automatically post grades, the value will be false.</li></ul>
		Allowed values: needs_grading_count, syllabus_body, public_description, total_scores, current_grading_period_scores, term, account, course_progress, sections, storage_quota_used_mb, total_students, passback_status, favorites, teachers, observed_users, all_courses, permissions, course_image, banner_image, concluded, lti_context_id, post_manually

teacher\_limit integer The maximum number of teacher enrollments to show. If the course contains more teachers than this, instead of giving the teacher enrollments, the count of teachers will be given under a *teacher\_count* key.

Returns a [Course](#) object

[Update a course CoursesController#update](#)

PUT /api/v1/courses/:id

**Scope:** url:PUT|/api/v1/courses/:id

Update an existing course.

Arguments are the same as Courses#create, with a few exceptions (enroll\_me).

If a user has content management rights, but not full course editing rights, the only attribute editable through this endpoint will be “syllabus\_body”

If an account has set prevent\_course\_availability\_editing\_by\_teachers, a teacher cannot change [course](#), [course](#), or [course](#) here.

Request Parameters:

Parameter	Type	Description
course[account_id]	integer	The unique ID of the account to move the course to.
course[name]	string	The name of the course. If omitted, the course will be named “Unnamed Course.”
course[course_code]	string	The course code for the course.
course[start_at]	DateTime	Course start date in ISO8601 format, e.g. 2011-01-01T01:00Z This value is ignored unless ‘restrict_enrollments_to_course_dates’ is set to true, or the course is already published.
course[end_at]	DateTime	Course end date in ISO8601 format, e.g. 2011-01-01T01:00Z This value is ignored unless ‘restrict_enrollments_to_course_dates’ is set to true.
course[license]	string	The name of the licensing. Should be one of the following abbreviations (a descriptive name is included in parenthesis for reference): <ul style="list-style-type: none"><li>‘private’ (Private Copyrighted)</li><li>‘cc_by_nc_nd’ (CC Attribution Non-Commercial No Derivatives)</li><li>‘cc_by_nc_sa’ (CC Attribution Non-Commercial Share Alike)</li><li>‘cc_by_nc’ (CC Attribution Non-Commercial)</li><li>‘cc_by_nd’ (CC Attribution No Derivatives)</li><li>‘cc_by_sa’ (CC Attribution Share Alike)</li><li>‘cc_by’ (CC Attribution)</li><li>‘public_domain’ (Public Domain).</li></ul>
course[is_public]	boolean	Set to true if course is public to both authenticated and unauthenticated users.
course[is_public_to_auth_users]	boolean	Set to true if course is public only to authenticated users.
course[public_syllabus]	boolean	Set to true to make the course syllabus public.
course[public_syllabus_to_auth]	boolean	Set to true to make the course syllabus to public for authenticated users.
course[public_description]	string	A publicly visible description of the course.
course[allow_student_wiki_edits]	boolean	If true, students will be able to modify the course wiki.
course[allow_wiki_comments]	boolean	If true, course members will be able to comment on wiki pages.
course[allow_student_forum_attachments]	boolean	If true, students can attach files to forum posts.
course[open_enrollment]	boolean	Set to true if the course is open enrollment.
course[self_enrollment]	boolean	Set to true if the course is self enrollment.

Parameter	Type	Description
course[restrict_enrollments_to_course_dates]	boolean	Set to true to restrict user enrollments to the start and end dates of the course. Setting this value to false will remove the course end date (if it exists), as well as the course start date (if the course is unpublished).
course[term_id]	integer	The unique ID of the term to create to course in.
course[sis_course_id]	string	The unique SIS identifier.
course[integration_id]	string	The unique Integration identifier.
course[hide_final_grades]	boolean	If this option is set to true, the totals in student grades summary will be hidden.
course[time_zone]	string	The time zone for the course. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
course[apply_assignment_group_weights]	boolean	Set to true to weight final grade based on assignment groups percentages.
course[storage_quota_mb]	integer	Set the storage quota for the course, in megabytes. The caller must have the “Manage storage quotas” account permission.
offer	boolean	If this option is set to true, the course will be available to students immediately.
course[event]	string	The action to take on each course.
		<ul style="list-style-type: none"><li>• ‘claim’ makes a course no longer visible to students. This action is also called “unpublish” on the web site. A course cannot be unpublished if students have received graded submissions.</li><li>• ‘offer’ makes a course visible to students. This action is also called “publish” on the web site.</li></ul>
		<ul style="list-style-type: none"><li>• ‘conclude’ prevents future enrollments and makes a course read-only for all participants. The course still appears in prior-enrollment lists.</li><li>• ‘delete’ completely removes the course from the web site (including course menus and prior-enrollment lists). All enrollments are deleted. Course content may be physically deleted at a future date.</li><li>• ‘undelete’ attempts to recover a course that has been deleted. This action requires account administrative rights. (Recovery is not guaranteed; please conclude rather than delete a course if there is any possibility the course will be used again.) The recovered course will be unpublished. Deleted enrollments will not be recovered.</li></ul>
		Allowed values: claim, offer, conclude, delete, undelete
course[default_view]	string	The type of page that users will see when they first visit the course
		<ul style="list-style-type: none"><li>• ‘feed’ Recent Activity Dashboard</li><li>• ‘wiki’ Wiki Front Page</li><li>• ‘modules’ Course Modules/Sections Page</li><li>• ‘assignments’ Course Assignments List</li><li>• ‘syllabus’ Course Syllabus Page</li></ul>
		other types may be added in the future
		Allowed values: feed, wiki, modules, syllabus, assignments
course[syllabus_body]	string	The syllabus body for the course
course[syllabus_course_summary]	boolean	Optional. Indicates whether the Course Summary (consisting of the course’s assignments and calendar events) is displayed on the syllabus page. Defaults to true.
course[grading_standard_id]	integer	The grading standard id to set for the course. If no value is provided for this argument the current grading_standard will be un-set from this course.
course[grade_passback_setting]	string	Optional. The grade_passback_setting for the course. Only ‘nightly_sync’ and “” are allowed
course[course_format]	string	Optional. Specifies the format of the course. (Should be either ‘on_campus’ or ‘online’)
course[image_id]	integer	This is a file ID corresponding to an image file in the course that will be used as the course image. This will clear the course’s image_url setting if set. If you attempt to provide image_url and image_id in a request it will fail.
course[image_url]	string	This is a URL to an image to be used as the course image. This will clear the course’s image_id setting if set. If you attempt to provide image_url and image_id in a request it will fail.
course[remove_image]	boolean	If this option is set to true, the course image url and course image ID are both set to nil
course[remove_banner_image]	boolean	If this option is set to true, the course banner image url and course banner image ID are both set to nil
course[blueprint]	boolean	Sets the course as a blueprint course.
course[blueprint_restrictions]	BlueprintRestriction	Sets a default set to apply to blueprint course objects when restricted, unless use_blueprint_restrictions_by_object_type is enabled. See the <a href="#">Blueprint Restriction</a> documentation
course[use_blueprint_restrictions_by_object_type]	boolean	When enabled, the <i>blueprint_restrictions</i> parameter will be ignored in favor of the <i>blueprint_restrictions_by_object_type</i> parameter
course[blueprint_restrictions_by_object_type]	multiple BlueprintRestrictions	Allows setting multiple <a href="#">Blueprint Restriction</a> to apply to blueprint course objects of the matching type when restricted. The possible object types are “assignment”, “attachment”, “discussion_topic”, “quiz” and “wiki_page”. Example usage: <pre>course[blueprint_restrictions_by_object_type][assignment][content]=1</pre>
course[homeroom_course]	boolean	Sets the course as a homeroom course. The setting takes effect only when the course is associated with a Canvas for Elementary-enabled account.
course[sync_enrollments_from_homeroom]	string	Syncs enrollments from the homeroom that is set in homeroom_course_id. The setting only takes effect when the course is associated with a Canvas for Elementary-enabled account and sync_enrollments_from_homeroom is enabled.
course[homeroom_course_id]	string	Sets the Homeroom Course id to be used with sync_enrollments_from_homeroom. The setting only takes effect when the course is associated with a Canvas for Elementary-enabled account and sync_enrollments_from_homeroom is enabled.
course[template]	boolean	Enable or disable the course as a template that can be selected by an account
course[course_color]	string	Sets a color in hex code format to be associated with the course. The setting takes effect only when the course is associated with a Canvas for Elementary-enabled account.
course[friendly_name]	string	Set a friendly name for the course. If this is provided and the course is associated with a Canvas for Elementary account, it will be shown instead of the course name. This setting takes priority over course nicknames defined by individual users.
course[enable_course_paces]	boolean	Enable or disable Course Pacing for the course. This setting only has an effect when the Course Pacing feature flag is enabled for the sub-account. Otherwise, Course Pacing are always disabled.  Note: Course Pacing is in active development.
course[conditional_release]	boolean	Enable or disable individual learning paths for students based on assessment

Parameter	Type	Description
course[post_manually]	boolean	When true, all grades in the course will be posted manually. When false, all grades in the course will be automatically posted. Use with caution as this setting will override any assignment level post policy.
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id> \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'course[name]=New course name' \
-d 'course[start_at]=2012-05-05T00:00:00Z'
```

Example Response:

```
{
  "name": "New course name",
  "course_code": "COURSE-001",
  "start_at": "2012-05-05T00:00:00Z",
  "end_at": "2012-08-05T23:59:59Z",
  "sis_course_id": "12345"
}
```

Update courses CoursesController#batch\_update

PUT /api/v1/accounts/:account\_id/courses

Scope: url:PUT|/api/v1/accounts/:account\_id/courses

Update multiple courses in an account. Operates asynchronously; use the [progress endpoint](#) to query the status of an operation.

Request Parameters:

Parameter	Type	Description
course_ids[]	Required string	Required string List of ids of courses to update. At most 500 courses may be updated in one call.
event	Required string	The action to take on each course. Must be one of 'offer', 'conclude', 'delete', or 'undelete'. <ul style="list-style-type: none"><li>'offer' makes a course visible to students. This action is also called “publish” on the web site.</li><li>'conclude' prevents future enrollments and makes a course read-only for all participants. The course still appears in prior-enrollment lists.</li><li>'delete' completely removes the course from the web site (including course menus and prior-enrollment lists). All enrollments are deleted. Course content may be physically deleted at a future date.</li><li>'undelete' attempts to recover a course that has been deleted. (Recovery is not guaranteed; please conclude rather than delete a course if there is any possibility the course will be used again.) The recovered course will be unpublished. Deleted enrollments will not be recovered.</li></ul> Allowed values: offer, conclude, delete, undelete

Example Request:

```
curl https://<canvas>/api/v1/accounts/<account_id>/courses \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'event=offer' \
-d 'course_ids[]=1' \
-d 'course_ids[]=2'
```

Returns a [Progress](#) object

Reset a course CoursesController#reset\_content

POST /api/v1/courses/:course\_id/reset\_content

Scope: url:POST|/api/v1/courses/:course\_id/reset\_content

Deletes the current course, and creates a new equivalent course with no content, but all sections and users moved over.

Returns a [Course](#) object

Get effective due dates CoursesController#effective\_due\_dates

GET /api/v1/courses/:course\_id/effective\_due\_dates

Scope: url:GET|/api/v1/courses/:course\_id/effective\_due\_dates

For each assignment in the course, returns each assigned student's ID and their corresponding due date along with some grading period data. Returns a collection with keys representing assignment IDs and values as a collection containing keys representing student IDs and values representing the student's effective due\_at, the grading\_period\_id of which the due\_at falls in, and whether or not the grading period is closed (in\_closed\_grading\_period)

The list of assignment IDs for which effective student due dates are requested. If not provided, all assignments in the course will be used.

Request Parameters:

Parameter	Type	Description
assignment_ids[]	string	no description

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/effective_due_dates
-X GET \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "1": {
    "14": { "due_at": "2015-09-05", "grading_period_id": null, "in_closed_grading_period": false },
    "15": { due_at: null, "grading_period_id": 3, "in_closed_grading_period": true }
  },
  "2": {
    "14": { "due_at": "2015-08-05", "grading_period_id": 3, "in_closed_grading_period": true }
  }
}
```

Permissions CoursesController#permissions

GET /api/v1/courses/:course\_id/permissions

Scope: url:GET|/api/v1/courses/:course\_id/permissions

Returns permission information for the calling user in the given course. See also the [Account](#) and [Group](#) counterparts.

Request Parameters:

Parameter	Type	Description
permissions[]	string	List of permissions to check against the authenticated user. Permission names are documented in the <a href="#">Create a role</a> endpoint.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/permissions \
-H 'Authorization: Bearer <token>' \
-d 'permissions[]=manage_grades' \
-d 'permissions[]=send_messages'
```

Example Response:

```
{ 'manage_grades': 'false', 'send_messages': 'true' }
```

Get bulk user progress CoursesController#bulk\_user\_progress

GET /api/v1/courses/:course\_id/bulk\_user\_progress

Scope: url:GET|api/v1/courses/:course\_id/bulk\_user\_progress

Returns progress information for all users enrolled in the given course.

You must be a user who has permission to view all grades in the course (such as a teacher or administrator).

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/bulk_user_progress \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  {
    "id": 1,
    "display_name": "Test Student 1",
    "avatar_image_url": "https://<canvas>/images/messages/avatar-50.png",
    "html_url": "https://<canvas>/courses/1/users/1",
    "pronouns": null,
    "progress": {
      "requirement_count": 2,
      "requirement_completed_count": 1,
      "next_requirement_url": "https://<canvas>/courses/<course_id>/modules/items/<item_id>",
      "completed_at": null
    }
  },
  {
    "id": 2,
    "display_name": "Test Student 2",
    "avatar_image_url": "https://<canvas>/images/messages/avatar-50.png",
    "html_url": "https://<canvas>/courses/1/users/2",
    "pronouns": null,
    "progress": {
      "requirement_count": 2,
      "requirement_completed_count": 2,
      "next_requirement_url": null,
      "completed_at": "2021-08-10T16:26:08Z"
    }
  }
}
```

Remove quiz migration alert CoursesController#dismiss\_migration\_limitation\_msg

POST /api/v1/courses/id/dismiss\_migration\_limitation\_message

Scope: url:POST|api/v1/courses/:id/dismiss\_migration\_limitation\_message

Remove alert about the limitations of quiz migrations that is displayed to a user in a course

you must be logged in to use this endpoint

Example Response:

```
{ "success": "true" }
```

Get course copy status ContentImportsController#copy\_course\_status

GET /api/v1/courses/:course\_id/course\_copy:id

Scope: url:GET|api/v1/courses/:course\_id/course\_copy:id

DEPRECATED: Please use the [Content Migrations API](#)

Retrieve the status of a course copy

API response field:

- id  
The unique identifier for the course copy.
- created\_at  
The time that the copy was initiated.
- progress  
The progress of the copy as an integer. It is null before the copying starts, and 100 when finished.
- workflow\_state  
The current status of the course copy. Possible values: "created", "started", "completed", "failed"
- status\_url  
The url for the course copy status API endpoint.

Example Response:

```
{ 'progress':100, 'workflow_state':'completed', 'id':257, 'created_at':'2011-11-17T16:50:06Z', 'status_url':'/api/v1/courses/9457/course_copy/257' }
```

Copy course content ContentImportsController#copy\_course\_content

POST /api/v1/courses/:course\_id/course\_copy

Scope: url:POST|api/v1/courses/:course\_id/course\_copy

DEPRECATED: Please use the [Content Migrations API](#)

Copies content from one course into another. The default is to copy all course content. You can control specific types to copy by using either the 'except' option or the 'only' option.

The response is the same as the course copy status endpoint

Request Parameters:

Parameter	Type	Description
source_course	string	ID or SIS-ID of the course to copy the content from
except[]	string	A list of the course content types to exclude, all areas not listed will be copied. Allowed values: course_settings, assignments, external_tools, files, topics, calendar_events, quizzes, wiki_pages, modules, outcomes
only[]	string	A list of the course content types to copy, all areas not listed will not be copied. Allowed values: course_settings, assignments, external_tools, files, topics, calendar_events, quizzes, wiki_pages, modules, outcomes

List custom gradebook columns CustomGradebookColumnsApiController#index

GET /api/v1/courses/:course\_id/custom\_gradebook\_columns

Scope: url:GET|api/v1/courses/:course\_id/custom\_gradebook\_columns

A paginated list of all custom gradebook columns for a course

Request Parameters:

Parameter	Type	Description
include_hidden	boolean	Include hidden parameters (defaults to false)

Returns a list of [CustomColumn](#) objects

Create a custom gradebook column CustomGradebookColumnsApiController#create

POST /api/v1/courses/:course\_id/custom\_gradebook\_columns

Scope: url:POST|/api/v1/courses/:course\_id/custom\_gradebook\_columns

Create a custom gradebook column

Request Parameters:

Parameter	Type	Description
column[title]	Required string	no description
column[position]	integer	The position of the column relative to other custom columns
column[hidden]	boolean	Hidden columns are not displayed in the gradebook
column[teacher_notes]	boolean	Set this if the column is created by a teacher. The gradebook only supports one teacher_notes column.
column[read_only]	boolean	Set this to prevent the column from being editable in the gradebook ui

Returns a CustomColumn object

Update a custom gradebook column CustomGradebookColumnsApiController#update

PUT /api/v1/courses/:course\_id/custom\_gradebook\_columns/:id

Scope: url:PUT|/api/v1/courses/:course\_id/custom\_gradebook\_columns/:id

Accepts the same parameters as custom gradebook column creation

Returns a CustomColumn object

Delete a custom gradebook column CustomGradebookColumnsApiController#destroy

DELETE /api/v1/courses/:course\_id/custom\_gradebook\_columns/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/custom\_gradebook\_columns/:id

Permanently deletes a custom column and its associated data

Returns a CustomColumn object

Reorder custom columns CustomGradebookColumnsApiController#reorder

POST /api/v1/courses/:course\_id/custom\_gradebook\_columns/reorder

Scope: url:POST|/api/v1/courses/:course\_id/custom\_gradebook\_columns/reorder

Puts the given columns in the specified order

200 OK is returned if successful

Request Parameters:

Parameter	Type	Description
order[]	Required integer	no description

List entries for a column CustomGradebookColumnDataApiController#index

GET /api/v1/courses/:course\_id/custom\_gradebook\_columns/:id/data

Scope: url:GET|/api/v1/courses/:course\_id/custom\_gradebook\_columns/:id/data

This does not list entries for students without associated data.

Request Parameters:

Parameter	Type	Description
include_hidden	boolean	If true, hidden columns will be included in the result. If false or absent, only visible columns will be returned.

Returns a list of ColumnDatum objects

Update column data CustomGradebookColumnDataApiController#update

PUT /api/v1/courses/:course\_id/custom\_gradebook\_columns/:id/data/:user\_id

Scope: url:PUT|/api/v1/courses/:course\_id/custom\_gradebook\_columns/:id/data/:user\_id

Set the content of a custom column

Request Parameters:

Parameter	Type	Description
column_data[content]	Required string	Column content. Setting this to blank will delete the datum object.

Returns a ColumnDatum object

Bulk update column data CustomGradebookColumnDataApiController#bulk\_update

PUT /api/v1/courses/:course\_id/custom\_gradebook\_column\_data

Scope: url:PUT|/api/v1/courses/:course\_id/custom\_gradebook\_column\_data

Set the content of custom columns

```
{
  "column_data": [
    {
      "column_id": example_column_id,
      "user_id": example_student_id,
      "content": example_content
    },
    {
      "column_id": example_column_id,
      "user_id": example_student_id,
      "content": example_content
    }
  ]
}
```

Request Parameters:

Parameter	Type	Description
column_data[]	Required Array	Column content. Setting this to an empty string will delete the data object.

Example Request:

Returns a Progress object



List discussion topics DiscussionTopicsController#index

GET /api/v1/courses/:course\_id/discussion\_topics

Scope: url:GET/api/v1/courses/:course\_id/discussion\_topics

GET /api/v1/groups/:group\_id/discussion\_topics

Scope: url:GET/api/v1/groups/:group\_id/discussion\_topics

Returns the paginated list of discussion topics for this course or group.

Request Parameters:

Parameter	Type	Description
include[]	string	If “all_dates” is passed, all dates associated with graded discussions’ assignments will be included. If “sections” is passed, includes the course sections that are associated with the topic, if the topic is specific to certain sections of the course. If “sections_user_count” is passed, then: (a) If sections were asked for “and” the topic is specific to certain course sections, includes the number of users in each section. (as part of the section json asked for above) (b) Else, includes at the root level the total number of users in the topic’s context (group or course) that the topic applies to.
		If “overrides” is passed, the overrides for the assignment will be included
		Allowed values: all_dates, sections, sections_user_count, overrides
order_by	string	Determines the order of the discussion topic list. Defaults to “position”.
		Allowed values: position, recent_activity, title
scope	string	Only return discussion topics in the given state(s). Defaults to including all topics. Filtering is done after pagination, so pages may be smaller than requested if topics are filtered. Can pass multiple states as comma separated string.
		Allowed values: locked, unlocked, pinned, unpinned
only_announcements	boolean	Return announcements instead of discussion topics. Defaults to false
filter_by	string	The state of the discussion topic to return. Currently only supports unread state.
		Allowed values: all, unread
search_term	string	The partial title of the discussion topics to match and return.
exclude_context_module_locked_topics	boolean	For students, exclude topics that are locked by module progression. Defaults to false.

Example Request:

```
curl https://canvas>/api/v1/courses/<course_id>/discussion_topics \
-H 'Authorization: Bearer <token>'
```

Returns a list of DiscussionTopic objects

Create a new discussion topic DiscussionTopicsController#create

POST /api/v1/courses/:course\_id/discussion\_topics

Scope: url:POST/api/v1/courses/:course\_id/discussion\_topics

POST /api/v1/groups/:group\_id/discussion\_topics

Scope: url:POST/api/v1/groups/:group\_id/discussion\_topics

Create an new discussion topic for the course or group.

Request Parameters:

Parameter	Type	Description
title	string	no description
message	string	no description
discussion_type	string	The type of discussion. Defaults to side_comment or not_threaded if not value is given. Accepted values are ‘side_comment’, ‘not_threaded’ for discussions that only allow one level of nested comments, and ‘threaded’ for fully threaded discussions.
		Allowed values: side_comment, threaded, not_threaded
published	boolean	Whether this topic is published (true) or draft state (false). Only teachers and TAs have the ability to create draft state topics.
delayed_post_at	DateTime	If a timestamp is given, the topic will not be published until that time.
allow_rating	boolean	Whether or not users can rate entries in this topic.
lock_at	DateTime	If a timestamp is given, the topic will be scheduled to lock at the provided timestamp. If the timestamp is in the past, the topic will be locked.
podcast_enabled	boolean	If true, the topic will have an associated podcast feed.
podcast_has_student_posts	boolean	If true, the podcast will include posts from students as well. Implies podcast_enabled.
require_initial_post	boolean	If true then a user may not respond to other replies until that user has made an initial reply. Defaults to false.
assignment	Assignment	To create an assignment discussion, pass the assignment parameters as a sub-object. See the Create an Assignment API for the available parameters. The name parameter will be ignored, as it’s taken from the discussion title. If you want to make a discussion that was an assignment NOT an assignment, pass set_assignment = false as part of the assignment object
is_announcement	boolean	If true, this topic is an announcement. It will appear in the announcement’s section rather than the discussions section. This requires announcement-posting permissions.
pinned	boolean	If true, this topic will be listed in the “Pinned Discussion” section
position_after	string	By default, discussions are sorted chronologically by creation date, you can pass the id of another topic to have this one show up after the other when they are listed.
group_category_id	integer	If present, the topic will become a group discussion assigned to the group.
only_graders_can_rate	boolean	If true, only graders will be allowed to rate entries.
sort_by_rating	boolean	If true, entries will be sorted by rating.
attachment	File	A multipart/form-data form-field-style attachment. Attachments larger than 1 kilobyte are subject to quota restrictions.

Parameter	Type	Description
specific_sections	string	A comma-separated list of sections ids to which the discussion topic should be made specific to. If it is not desired to make the discussion topic specific to sections, then this parameter may be omitted or set to "all". Can only be present only on announcements and only those that are for a course (as opposed to a group).

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/discussion_topics \
-F 'title=my topic' \
-F 'message=initial message' \
-F 'podcast_enabled=1' \
-H 'Authorization: Bearer <token>'
-F 'attachment=@<filename>' \

curl https://<canvas>/api/v1/courses/<course_id>/discussion_topics \
-F 'title=my assignment topic' \
-F 'message=initial message' \
-F 'assignment[points_possible]=15' \
-H 'Authorization: Bearer <token>'
```

Update a topic DiscussionTopicsController#update

PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id

Scope: url:PUT[/api/v1/courses/:course\_id/discussion\_topics/:topic\_id]

PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id

Scope: url:PUT[/api/v1/groups/:group\_id/discussion\_topics/:topic\_id]

Update an existing discussion topic for the course or group.

Request Parameters:

Parameter	Type	Description
title	string	no description
message	string	no description
discussion_type	string	The type of discussion. Defaults to side_comment or not_threaded if not value is given. Accepted values are 'side_comment', 'not_threaded' for discussions that only allow one level of nested comments, and 'threaded' for fully threaded discussions.  Allowed values: side_comment, threaded, not_threaded
published	boolean	Whether this topic is published (true) or draft state (false). Only teachers and TAs have the ability to create draft state topics.
delayed_post_at	DateTime	If a timestamp is given, the topic will not be published until that time.
lock_at	DateTime	If a timestamp is given, the topic will be scheduled to lock at the provided timestamp. If the timestamp is in the past, the topic will be locked.
podcast_enabled	boolean	If true, the topic will have an associated podcast feed.
podcast_has_student_posts	boolean	If true, the podcast will include posts from students as well. Implies podcast_enabled.
require_initial_post	boolean	If true then a user may not respond to other replies until that user has made an initial reply. Defaults to false.
assignment	Assignment	To create an assignment discussion, pass the assignment parameters as a sub-object. See the <a href="#">Create an Assignment API</a> for the available parameters. The name parameter will be ignored, as it's taken from the discussion title. If you want to make a discussion that was an assignment NOT an assignment, pass set_assignment = false as part of the assignment object
is_announcement	boolean	If true, this topic is an announcement. It will appear in the announcement's section rather than the discussions section. This requires announcement-posting permissions.
pinned	boolean	If true, this topic will be listed in the "Pinned Discussion" section
position_after	string	By default, discussions are sorted chronologically by creation date, you can pass the id of another topic to have this one show up after the other when they are listed.
group_category_id	integer	If present, the topic will become a group discussion assigned to the group.
allow_rating	boolean	If true, users will be allowed to rate entries.
only_graders_can_rate	boolean	If true, only graders will be allowed to rate entries.
sort_by_rating	boolean	If true, entries will be sorted by rating.
specific_sections	string	A comma-separated list of sections ids to which the discussion topic should be made specific too. If it is not desired to make the discussion topic specific to sections, then this parameter may be omitted or set to "all". Can only be present only on announcements and only those that are for a course (as opposed to a group).

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id> \
-F 'title=This will be positioned after Topic #1234' \
-F 'position_after=1234' \
-H 'Authorization: Bearer <token>'
```

Delete a topic DiscussionTopicsController#destroy

DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id

Scope: url:DELETE[/api/v1/courses/:course\_id/discussion\_topics/:topic\_id]

DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id

Scope: url:DELETE[/api/v1/groups/:group\_id/discussion\_topics/:topic\_id]

Deletes the discussion topic. This will also delete the assignment, if it's an assignment discussion.

Example Request:

```
curl -X DELETE https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id> \
-H 'Authorization: Bearer <token>'
```

Reorder pinned topics DiscussionTopicsController#reorder

POST /api/v1/courses/:course\_id/discussion\_topics/reorder

Scope: url:POST[/api/v1/courses/:course\_id/discussion\_topics/reorder]

POST /api/v1/groups/:group\_id/discussion\_topics/reorder

Scope: url:POST[/api/v1/groups/:group\_id/discussion\_topics/reorder]

Puts the pinned discussion topics in the specified order. All pinned topics should be included.

Request Parameters:

Parameter	Type	Description
order[]	Required integer	The ids of the pinned discussion topics in the desired order. (For example, "order=104,102,103".)

Update an entry DiscussionEntriesController#update

PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:id

Scope: url:PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:id

PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:id

Scope: url:PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:id

Update an existing discussion entry.

The entry must have been created by the current user, or the current user must have admin rights to the discussion. If the edit is not allowed, a 401 will be returned.

Request Parameters:

Parameter	Type	Description
message	string	The updated body of the entry.

Example Request:

```
curl -X PUT 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>' \
-F 'message=<message>' \
-H 'Authorization: Bearer <token>'
```

Delete an entry DiscussionEntriesController#destroy

DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:id

Scope: url:DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:id

DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:id

Scope: url:DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:id

Delete a discussion entry.

The entry must have been created by the current user, or the current user must have admin rights to the discussion. If the delete is not allowed, a 401 will be returned.

The discussion will be marked deleted, and the user\_id and message will be cleared out.

Example Request:

```
curl -X DELETE 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>' \
-H 'Authorization: Bearer <token>'
```

Get a single topic DiscussionTopicsApiController#show

GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id

Scope: url:GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id

GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id

Scope: url:GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id

Returns data on an individual discussion topic. See the List action for the response formatting.

Request Parameters:

Parameter	Type	Description
		If “all_dates” is passed, all dates associated with graded discussions’ assignments will be included. If “sections” is passed, includes the course sections that are associated with the topic, if the topic is specific to certain sections of the course. If “sections_user_count” is passed, then:
include[]	string	(a) If sections were asked for “and” the topic is specific to certain course sections, includes the number of users in each section. (as part of the section json asked for above) (b) Else, includes at the root level the total number of users in the topic’s context (group or course) that the topic applies to.
		If “overrides” is passed, the overrides for the assignment will be included
		Allowed values: all_dates, sections, sections_user_count, overrides

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id> \
-H 'Authorization: Bearer <token>'
```

Summary DiscussionTopicsApiController#summary

GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries

Scope: url:GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries

GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries

Scope: url:GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries

Generates a summary for a discussion topic.

Request Parameters:

Parameter	Type	Description
userInput	string	Areas or topics for the summary to focus on.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/summaries \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "id": 1,
  "text": "This is a summary of the discussion topic."
}
```

Disable summary DiscussionTopicsApiController#disable\_summary

PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries/disable

Scope: url:PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries/disable

PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries/disable

Scope: url:PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries/disable

Disables the summary for a discussion topic.

Example Request:

```
curl -X PUT https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/disable_summary \
```

Example Response:

```
{
  "success": true
}
```

Summary Feedback DiscussionTopicsApiController#summary\_feedback

POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries/:summary\_id/feedback

Scope: url:POST|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/summaries/:summary\_id/feedback

POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries/:summary\_id/feedback

Scope: url:POST|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/summaries/:summary\_id/feedback

Persists feedback on a discussion topic summary.

Request Parameters:

Parameter	Type	Description
		Required The action to take on the summary. Possible values are:
		<ul style="list-style-type: none"><li>“seen”: Marks the summary as seen. This action saves the feedback if it's not already persisted.</li><li>“like”: Marks the summary as liked.</li><li>“dislike”: Marks the summary as disliked.</li><li>“reset_like”: Resets the like status of the summary.</li><li>“regenerate”: Regenerates the summary feedback.</li><li>“disable_summary”: Disables the summary feedback.</li></ul>
_action	string	Any other value will result in an error response.

Example Request:

```
curl -X POST https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/summaries/<summary_id>/feedback \
-F '_action=like' \
-H 'Authorization: Bearer
```

Example Response:

```
{
  "liked": true,
  "disliked": false
}
```

Get the full topic DiscussionTopicsApiController#view

GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/view

Scope: url:GET|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/view

GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/view

Scope: url:GET|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/view

Return a cached structure of the discussion topic, containing all entries, their authors, and their message bodies.

May require (depending on the topic) that the user has posted in the topic. If it is required, and the user has not posted, will respond with a 403 Forbidden status and the body 'require\_initial\_post'.

In some rare situations, this cached structure may not be available yet. In that case, the server will respond with a 503 error, and the caller should try again soon.

The response is an object containing the following keys:

- “participants”: A list of summary information on users who have posted to the discussion. Each value is an object containing their id, display\_name, and avatar\_url.
- “unread\_entries”: A list of entry ids that are unread by the current user: this implies that any entry not in this list is read.
- “entry\_ratings”: A map of entry ids to ratings by the current user. Entries not in this list have no rating. Only populated if rating is enabled.
- “forced\_entries”: A list of entry ids that have forced\_read\_state set to true. This flag is meant to indicate the entry's read\_state has been manually set to ‘unread’ by the user, so the entry should not be automatically marked as read.
- “view”: A threaded view of all the entries in the discussion, containing the id, user\_id, and message.
- “new\_entries”: Because this view is eventually consistent, it's possible that newly created or updated entries won't yet be reflected in the view. If the application wants to also get a flat list of all entries not yet reflected in the view, pass include\_new\_entries=1 to the request and this array of entries will be returned. These entries are returned in a flat array, in ascending created\_at order.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/view' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "unread_entries": [1,3,4],
  "entry_ratings": {3: 1},
  "forced_entries": [1],
  "participants": [
    { "id": 10, "display_name": "user 1", "avatar_image_url": "https://...", "html_url": "https://..." },
    { "id": 11, "display_name": "user 2", "avatar_image_url": "https://...", "html_url": "https://..." }
  ],
  "view": [
    { "id": 1, "user_id": 10, "parent_id": null, "message": "...html text...", "replies": [
      { "id": 3, "user_id": 11, "parent_id": 1, "message": "...html...", "replies": [...] }
    ] },
    { "id": 2, "user_id": 11, "parent_id": null, "message": "...html..." },
    { "id": 4, "user_id": 10, "parent_id": null, "message": "...html..." }
  ]
}
```

Post an entry DiscussionTopicsApiController#add\_entry

POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries

Scope: url:POST|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries

POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries

Scope: url:POST|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries

Create a new entry in a discussion topic. Returns a json representation of the created entry (see documentation for ‘entries’ method) on success.

Request Parameters:

Parameter	Type	Description
message	string	The body of the entry.
attachment	string	A multipart/form-data form-field-style attachment. Attachments larger than 1 kilobyte are subject to quota restrictions.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries.json' \
-F 'message=message' \
-F 'attachment=@filename' \
-H 'Authorization: Bearer <token>'
```

Duplicate discussion topic DiscussionTopicsApiController#duplicate

POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/duplicate

**Scope:** url:POST|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/duplicate

**POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/duplicate**

**Scope:** url:POST|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/duplicate

Duplicate a discussion topic according to context (Course/Group)

**Example Request:**

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/discussion_topics/123/duplicate

curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/group/456/discussion_topics/456/duplicate
```

Returns a [DiscussionTopic](#) object

**List topic entries [DiscussionTopicsApiController#entries](#)**

**GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries**

**Scope:** url:GET|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries

**GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries**

**Scope:** url:GET|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries

Retrieve the (paginated) top-level entries in a discussion topic.

May require (depending on the topic) that the user has posted in the topic. If it is required, and the user has not posted, will respond with a 403 Forbidden status and the body 'require\_initial\_post'.

Will include the 10 most recent replies, if any, for each entry returned.

If the topic is a root topic with children corresponding to groups of a group assignment, entries from those subtopics for which the user belongs to the corresponding group will be returned.

Ordering of returned entries is newest-first by posting timestamp (reply activity is ignored).

**API response field:**

- id  
The unique identifier for the entry.
- user\_id  
The unique identifier for the author of the entry.
- editor\_id  
The unique user id of the person to last edit the entry, if different than user\_id.
- user\_name  
The name of the author of the entry.
- message  
The content of the entry.
- read\_state  
The read state of the entry, "read" or "unread".
- forced\_read\_state  
Whether the read\_state was forced (was set manually)
- created\_at  
The creation time of the entry, in ISO8601 format.
- updated\_at  
The updated time of the entry, in ISO8601 format.
- attachment  
JSON representation of the attachment for the entry, if any. Present only if there is an attachment.
- attachments  
**Deprecated.** Same as attachment, but returned as a one-element array. Present only if there is an attachment.
- recent\_replies  
The 10 most recent replies for the entry, newest first. Present only if there is at least one reply.
- has\_more\_replies  
True if there are more than 10 replies for the entry (i.e., not all were included in this response). Present only if there is at least one reply.

**Example Response:**

```
[ {
  "id": 1019,
  "user_id": 7086,
  "user_name": "nobody@example.com",
  "message": "Newer entry",
  "read_state": "read",
  "forced_read_state": false,
  "created_at": "2011-11-03T21:33:29Z",
  "attachment": {
    "content_type": "unknown/unknown",
    "url": "http://www.example.com/files/681/download?verifier=J0G10Ru1tv8o6LjG0W1xg0b5513E1zVv9gC8KUT3",
    "filename": "content.txt",
    "display_name": "content.txt" },
  {
    "id": 1016,
    "user_id": 7086,
    "user_name": "nobody@example.com",
    "message": "first top-level entry",
    "read_state": "unread",
    "forced_read_state": false,
    "created_at": "2011-11-03T21:32:29Z",
    "recent_replies": [
      {
        "id": 1017,
        "user_id": 7086,
        "user_name": "nobody@example.com",
        "message": "Reply message",
        "created_at": "2011-11-03T21:32:29Z"
      }
    ],
    "has_more_replies": false } ]
```

**Post a reply [DiscussionTopicsApiController#add\\_reply](#)**

**POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies**

**Scope:** url:POST|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

**POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies**

**Scope:** url:POST|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

Add a reply to an entry in a discussion topic. Returns a json representation of the created reply (see documentation for 'replies' method) on success.

May require (depending on the topic) that the user has posted in the topic. If it is required, and the user has not posted, will respond with a 403 Forbidden status and the body 'require\_initial\_post'.

**Request Parameters:**

Parameter	Type	Description
message	string	The body of the entry.

Parameter	Type	Description
attachment	string	A multipart/form-data form-field-style attachment. Attachments larger than 1 kilobyte are subject to quota restrictions.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>/replies.json' \  
-F 'message=message' \  
-F 'attachment=@<filename>' \  
-H 'Authorization: Bearer <token>'
```

List entry replies DiscussionTopicsApiController#replies

GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

Scope: url:GET|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

Scope: url:GET|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/replies

Retrieve the (paginated) replies to a top-level entry in a discussion topic.

May require (depending on the topic) that the user has posted in the topic. If it is required, and the user has not posted, will respond with a 403 Forbidden status and the body 'require\_initial\_post'.

Ordering of returned entries is newest-first by creation timestamp.

API response field:

- id  
The unique identifier for the reply.
- user\_id  
The unique identifier for the author of the reply.
- editor\_id  
The unique user id of the person to last edit the entry, if different than user\_id.
- user\_name  
The name of the author of the reply.
- message  
The content of the reply.
- read\_state  
The read state of the entry, "read" or "unread".
- forced\_read\_state  
Whether the read\_state was forced (was set manually)
- created\_at  
The creation time of the reply, in ISO8601 format.

Example Response:

```
[ {  
  "id": 1015,  
  "user_id": 7004,  
  "user_name": "nobody@example.com",  
  "message": "Newer message",  
  "read_state": "read",  
  "forced_read_state": false,  
  "created_at": "2011-11-03T21:27:44Z" },  
 {  
  "id": 1014,  
  "user_id": 7004,  
  "user_name": "nobody@example.com",  
  "message": "Older message",  
  "read_state": "unread",  
  "forced_read_state": false,  
  "created_at": "2011-11-03T21:26:44Z" } ]
```

List entries DiscussionTopicsApiController#entry\_list

GET /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entry\_list

Scope: url:GET|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entry\_list

GET /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entry\_list

Scope: url:GET|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entry\_list

Retrieve a paginated list of discussion entries, given a list of ids.

May require (depending on the topic) that the user has posted in the topic. If it is required, and the user has not posted, will respond with a 403 Forbidden status and the body 'require\_initial\_post'.

Request Parameters:

Parameter	Type	Description
ids[]	string	A list of entry ids to retrieve. Entries will be returned in id order, smallest id first.

API response field:

- id  
The unique identifier for the reply.
- user\_id  
The unique identifier for the author of the reply.
- user\_name  
The name of the author of the reply.
- message  
The content of the reply.
- read\_state  
The read state of the entry, "read" or "unread".
- forced\_read\_state  
Whether the read\_state was forced (was set manually)
- created\_at  
The creation time of the reply, in ISO8601 format.
- deleted  
If the entry has been deleted, returns true. The user\_id, user\_name, and message will not be returned for deleted entries.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entry_list?ids[]=1&ids[]=2&ids[]=3' \  
-H 'Authorization: Bearer <token>'
```

Example Response:

```
[
  { ... entry 1 ... },
  { ... entry 2 ... },
  { ... entry 3 ... },
]
```

Mark topic as read DiscussionTopicsApiController#mark\_topic\_read

PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read

Scope: url:PUT|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read

PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read

Scope: url:PUT|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read

Mark the initial text of the discussion topic as read.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/read.json' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"
```

Mark all topic as read DiscussionTopicsApiController#mark\_all\_topic\_read

PUT /api/v1/courses/:course\_id/discussion\_topics/read\_all

Scope: url:PUT|/api/v1/courses/:course\_id/discussion\_topics/read\_all

PUT /api/v1/groups/:group\_id/discussion\_topics/read\_all

Scope: url:PUT|/api/v1/groups/:group\_id/discussion\_topics/read\_all

Mark the initial text of all the discussion topics as read in the context.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/read_all' \
-X POST \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"
```

Mark topic as unread DiscussionTopicsApiController#mark\_topic\_unread

DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read

Scope: url:DELETE|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read

DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read

Scope: url:DELETE|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read

Mark the initial text of the discussion topic as unread.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/read.json' \
-X DELETE \
-H "Authorization: Bearer <token>"
```

Mark all entries as read DiscussionTopicsApiController#mark\_all\_read

PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read\_all

Scope: url:PUT|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read\_all

PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read\_all

Scope: url:PUT|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read\_all

Mark the discussion topic and all its entries as read.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Request Parameters:

Parameter	Type	Description
forced_read_state	boolean	A boolean value to set all of the entries' forced_read_state. No change is made if this argument is not specified.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/read_all.json' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"
```

Mark all entries as unread DiscussionTopicsApiController#mark\_all\_unread

DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read\_all

Scope: url:DELETE|/api/v1/courses/:course\_id/discussion\_topics/:topic\_id/read\_all

DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read\_all

Scope: url:DELETE|/api/v1/groups/:group\_id/discussion\_topics/:topic\_id/read\_all

Mark the discussion topic and all its entries as unread.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Request Parameters:

Parameter	Type	Description
forced_read_state	boolean	A boolean value to set all of the entries' forced_read_state. No change is made if this argument is not specified.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/read_all.json' \
-X DELETE \
-H "Authorization: Bearer <token>"
```

[Mark entry as read DiscussionTopicsApiController#mark\\_entry\\_read](#)

**PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read**

**Scope:** url:PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read

**PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read**

**Scope:** url:PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read

Mark a discussion entry as read.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

**Request Parameters:**

Parameter	Type	Description
forced_read_state	boolean	A boolean value to set the entry's forced_read_state. No change is made if this argument is not specified.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>/read.json' \  
-X PUT \  
-H "Authorization: Bearer <token>" \  
-H "Content-Length: 0"
```

[Mark entry as unread DiscussionTopicsApiController#mark\\_entry\\_unread](#)

**DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read**

**Scope:** url:DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read

**DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read**

**Scope:** url:DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/read

Mark a discussion entry as unread.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

**Request Parameters:**

Parameter	Type	Description
forced_read_state	boolean	A boolean value to set the entry's forced_read_state. No change is made if this argument is not specified.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>/read.json' \  
-X DELETE \  
-H "Authorization: Bearer <token>"
```

[Rate entry DiscussionTopicsApiController#rate\\_entry](#)

**POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/rating**

**Scope:** url:POST /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/entries/:entry\_id/rating

**POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/rating**

**Scope:** url:POST /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/entries/:entry\_id/rating

Rate a discussion entry.

On success, the response will be 204 No Content with an empty body.

**Request Parameters:**

Parameter	Type	Description
rating	integer	A rating to set on this entry. Only 0 and 1 are accepted.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/entries/<entry_id>/rating.json' \  
-X POST \  
-H "Authorization: Bearer <token>"
```

[Subscribe to a topic DiscussionTopicsApiController#subscribe\\_topic](#)

**PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/subscribed**

**Scope:** url:PUT /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/subscribed

**PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/subscribed**

**Scope:** url:PUT /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/subscribed

Subscribe to a topic to receive notifications about new entries

On success, the response will be 204 No Content with an empty body

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/subscribed.json' \  
-X PUT \  
-H "Authorization: Bearer <token>" \  
-H "Content-Length: 0"
```

[Unsubscribe from a topic DiscussionTopicsApiController#unsubscribe\\_topic](#)

**DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/subscribed**

**Scope:** url:DELETE /api/v1/courses/:course\_id/discussion\_topics/:topic\_id/subscribed

**DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/subscribed**

**Scope:** url:DELETE /api/v1/groups/:group\_id/discussion\_topics/:topic\_id/subscribed

Unsubscribe from a topic to stop receiving notifications about new entries

On success, the response will be 204 No Content with an empty body

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/<course_id>/discussion_topics/<topic_id>/subscribed.json' \  
-X DELETE \  
-H "Authorization: Bearer <token>"
```

[Create enrollment term TermsController#create](#)

**POST /api/v1/accounts/:account\_id/terms**

**Scope:** url:POST /api/v1/accounts/:account\_id/terms



Create a new enrollment term for the specified account.

Request Parameters:

Parameter	Type	Description
enrollment_term[name]	string	The name of the term.
enrollment_term[start_at]	DateTime	The day/time the term starts. Accepts times in ISO 8601 format, e.g. 2015-01-10T18:48:00Z.
enrollment_term[end_at]	DateTime	The day/time the term ends. Accepts times in ISO 8601 format, e.g. 2015-01-10T18:48:00Z.
enrollment_term[sis_term_id]	string	The unique SIS identifier for the term.
enrollment_term[overrides][enrollment_type][start_at]	DateTime	The day/time the term starts, overridden for the given enrollment type. <b>enrollment_type</b> can be one of StudentEnrollment, TeacherEnrollment, TaEnrollment, or DesignerEnrollment
enrollment_term[overrides][enrollment_type][end_at]	DateTime	The day/time the term ends, overridden for the given enrollment type. <b>enrollment_type</b> can be one of StudentEnrollment, TeacherEnrollment, TaEnrollment, or DesignerEnrollment

Returns an [EnrollmentTerm](#) object

Update enrollment term TermsController#update

PUT /api/v1/accounts/:account\_id/terms/:id

Scope: url:PUT /api/v1/accounts/:account\_id/terms/:id

Update an existing enrollment term for the specified account.

Request Parameters:

Parameter	Type	Description
enrollment_term[name]	string	The name of the term.
enrollment_term[start_at]	DateTime	The day/time the term starts. Accepts times in ISO 8601 format, e.g. 2015-01-10T18:48:00Z.
enrollment_term[end_at]	DateTime	The day/time the term ends. Accepts times in ISO 8601 format, e.g. 2015-01-10T18:48:00Z.
enrollment_term[sis_term_id]	string	The unique SIS identifier for the term.
enrollment_term[overrides][enrollment_type][start_at]	DateTime	The day/time the term starts, overridden for the given enrollment type. <b>enrollment_type</b> can be one of StudentEnrollment, TeacherEnrollment, TaEnrollment, or DesignerEnrollment
enrollment_term[overrides][enrollment_type][end_at]	DateTime	The day/time the term ends, overridden for the given enrollment type. <b>enrollment_type</b> can be one of StudentEnrollment, TeacherEnrollment, TaEnrollment, or DesignerEnrollment
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Returns an [EnrollmentTerm](#) object

Delete enrollment term TermsController#destroy

DELETE /api/v1/accounts/:account\_id/terms/:id

Scope: url:DELETE /api/v1/accounts/:account\_id/terms/:id

Delete the specified enrollment term.

Returns an [EnrollmentTerm](#) object

List enrollment terms TermsApiController#index

GET /api/v1/accounts/:account\_id/terms

Scope: url:GET /api/v1/accounts/:account\_id/terms

An object with a paginated list of all of the terms in the account.

Request Parameters:

Parameter	Type	Description
workflow_state[]	string	If set, only returns terms that are in the given state. Defaults to 'active'.
		Allowed values: active, deleted, all
		Array of additional information to include.
include[]	string	“overrides”
		term start/end dates overridden for different enrollment types
		“course_count”
		the number of courses in each term
		Allowed values: overrides
term_name	string	If set, only returns terms that match the given search keyword. Search keyword is matched against term name.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/1/terms?include[]=overrides
```

Example Response:

```
{
  "enrollment_terms": [
    {
      "id": 1,
      "name": "Fall 2006",
      "start_at": "2006-09-31T20:00:00Z",
      "end_at": "2006-12-20T20:00:00Z",
      "created_at": "2025-01-02T03:43:11Z",
      "workflow_state": "active",
      "grading_period_group_id": 1,
      "sis_term_id": null,
      "overrides": {
        "StudentEnrollment": {
          "start_at": "2006-09-03T20:00:00Z",
          "end_at": "2006-12-19T20:00:00Z"
        },
        "TeacherEnrollment": {
          "start_at": null,
          "end_at": "2006-12-30T20:00:00Z"
        }
      }
    }
  ]
}
```

Returns an [EnrollmentTermsList](#) object

Retrieve enrollment term TermsApiController#show

GET /api/v1/accounts/:account\_id/terms/:id

Scope: url:GET//api/v1/accounts/:account\_id/terms/:id

Retrieves the details for an enrollment term in the account. Includes overrides by default.

Example Request:

curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/accounts/1/terms/2

Returns an EnrollmentTerm object

List enrollments EnrollmentsApiController#index

GET /api/v1/courses/:course\_id/enrollments

Scope: url:GET//api/v1/courses/:course\_id/enrollments

GET /api/v1/sections/:section\_id/enrollments

Scope: url:GET//api/v1/sections/:section\_id/enrollments

GET /api/v1/users/:user\_id/enrollments

Scope: url:GET//api/v1/users/:user\_id/enrollments

Depending on the URL given, return a paginated list of either (1) all of the enrollments in a course, (2) all of the enrollments in a section or (3) all of a user's enrollments. This includes student, teacher, TA, and observer enrollments.

If a user has multiple enrollments in a context (e.g. as a teacher and a student or in multiple course sections), each enrollment will be listed separately.

note: Currently, only a root level admin user can return other users' enrollments. A user can, however, return his/her own enrollments.

Enrollments scoped to a course context will include inactive states by default if the caller has account admin authorization and the state[] parameter is omitted.

Request Parameters:

Parameter	Type	Description
type[]	string	A list of enrollment types to return. Accepted values are 'StudentEnrollment', 'TeacherEnrollment', 'TaEnrollment', 'DesignerEnrollment', and 'ObserverEnrollment.' If omitted, all enrollment types are returned. This argument is ignored if 'role' is given.
role[]	string	A list of enrollment roles to return. Accepted values include course-level roles created by the Add Role API as well as the base enrollment types accepted by the 'type' argument above.
state[]	string	Filter by enrollment state. If omitted, 'active' and 'invited' enrollments are returned. The following synthetic states are supported only when querying a user's enrollments (either via user_id argument or via user enrollments endpoint): current_and_invited, current_and_future, current_future_and_restricted, current_and_concluded Allowed values: active, invited, creation_pending, deleted, rejected, completed, inactive, current_and_invited, current_and_future, current_future_and_restricted, current_and_concluded
include[]	string	Array of additional information to include on the enrollment or user records. "avatar_url" and "group_ids" will be returned on the user record. If "current_points" is specified, the fields "current_points" and (if the caller has permissions to manage grades) "unposted_current_points" will be included in the "grades" hash for student enrollments. Allowed values: avatar_url, group_ids, locked, observed_users, can_be_removed, uuid, current_points
user_id	string	Filter by user_id (only valid for course or section enrollment queries). If set to the current user's id, this is a way to determine if the user has any enrollments in the course or section, independent of whether the user has permission to view other people on the roster.
grading_period_id	integer	Return grades for the given grading_period. If this parameter is not specified, the returned grades will be for the whole course.
enrollment_term_id	integer	Returns only enrollments for the specified enrollment term. This parameter only applies to the user enrollments path. May pass the ID from the enrollment terms api or the SIS id prepended with 'sis_term_id'.
sis_account_id[]	string	Returns only enrollments for the specified SIS account ID(s). Does not look into sub_accounts. May pass in array or string.
sis_course_id[]	string	Returns only enrollments matching the specified SIS course ID(s). May pass in array or string.
sis_section_id[]	string	Returns only section enrollments matching the specified SIS section ID(s). May pass in array or string.
sis_user_id[]	string	Returns only enrollments for the specified SIS user ID(s). May pass in array or string.
created_for_sis_id[]	boolean	If sis_user_id is present and created_for_sis_id is true, Returns only enrollments for the specified SIS ID(s). If a user has two sis_id's, one enrollment may be created using one of the two ids. This would limit the enrollments returned from the endpoint to enrollments that were created from a sis_import with that sis_user_id

Returns a list of Enrollment objects

Enrollment by ID EnrollmentsApiController#show

GET /api/v1/accounts/:account\_id/enrollments/:id

Scope: url:GET//api/v1/accounts/:account\_id/enrollments/:id

Get an Enrollment object by Enrollment ID

Request Parameters:

Parameter	Type	Description
id	Required integer	The ID of the enrollment object

Returns an Enrollment object

Enroll a user EnrollmentsApiController#create

POST /api/v1/courses/:course\_id/enrollments

Scope: url:POST//api/v1/courses/:course\_id/enrollments

POST /api/v1/sections/:section\_id/enrollments

Scope: url:POST//api/v1/sections/:section\_id/enrollments

Create a new user enrollment for a course or section.

Request Parameters:

Parameter	Type	Description
enrollment[start_at]	DateTime	The start time of the enrollment, in ISO8601 format. e.g. 2012-04-18T23:08:51Z
enrollment[end_at]	DateTime	The end time of the enrollment, in ISO8601 format. e.g. 2012-04-18T23:08:51Z
enrollment[user_id]	Required string	The ID of the user to be enrolled in the course.
enrollment[type]	Required string	Enroll the user as a student, teacher, TA, observer, or designer. If no value is given, the type will be inferred by enrollment if supplied, otherwise 'StudentEnrollment' will be used. Allowed values: StudentEnrollment, TeacherEnrollment, TaEnrollment, ObserverEnrollment, DesignerEnrollment

Parameter	Type	Description
enrollment[role]		Deprecated Assigns a custom course-level role to the user.
enrollment[role_id]	integer	Assigns a custom course-level role to the user.
enrollment[enrollment_state]	string	<p>If set to 'active,' student will be immediately enrolled in the course. Otherwise they will be required to accept a course invitation. Default is 'invited.'</p> <p>If set to 'inactive', student will be listed in the course roster for teachers, but will not be able to participate in the course until their enrollment is activated.</p> <p>Allowed values: active, invited, inactive</p>
enrollment[course_section_id]	integer	<p>The ID of the course section to enroll the student in. If the section-specific URL is used, this argument is redundant and will be ignored.</p> <p>If set, the enrollment will only allow the user to see and interact with users enrolled in the section given by course_section_id.</p> <ul style="list-style-type: none"><li>For teachers and TAs, this includes grading privileges.</li><li>Section-limited students will not see any users (including teachers and TAs) not enrolled in their sections.</li><li>Users may have other enrollments that grant privileges to multiple sections in the same course.</li></ul>
enrollment[limit_privileges_to_course_section]	boolean	
enrollment[notify]	boolean	If true, a notification will be sent to the enrolled user. Notifications are not sent by default.
enrollment[self_enrollment_code]	string	If the current user is not allowed to manage enrollments in this course, but the course allows self-enrollment, the user can self- enroll as a student in the default section by passing in a valid code. When self-enrolling, the user_id must be 'self'. The enrollment_state will be set to 'active' and all other arguments will be ignored.
enrollment[self_enrolled]	boolean	If true, marks the enrollment as a self-enrollment, which gives students the ability to drop the course if desired. Defaults to false.
enrollment[associated_user_id]	integer	For an observer enrollment, the ID of a student to observe. This is a one-off operation; to automatically observe all a student's enrollments (for example, as a parent), please use the <a href="#">User Observes API</a> .
enrollment[sis_user_id]	string	Required if the user is being enrolled from another trusted account. The unique identifier for the user (sis_user_id) must also be accompanied by the root_account parameter. The user_id will be ignored.
enrollment[integration_id]	string	Required if the user is being enrolled from another trusted account. The unique identifier for the user (integration_id) must also be accompanied by the root_account parameter. The user_id will be ignored.
root_account	string	The domain of the account to search for the user. Will be a no-op unless the sis_user_id or integration_id parameter is also included.

Example Request:

```
curl https://<canvas>/api/v1/courses/:course_id/enrollments \
-X POST \
-F 'enrollment[user_id]=1' \
-F 'enrollment[type]=StudentEnrollment' \
-F 'enrollment[enrollment_state]=active' \
-F 'enrollment[course_section_id]=1' \
-F 'enrollment[limit_privileges_to_course_section]=true' \
-F 'enrollment[notify]=false'

curl https://<canvas>/api/v1/courses/:course_id/enrollments \
-X POST \
-F 'enrollment[user_id]=2' \
-F 'enrollment[type]=StudentEnrollment'
```

Returns an [Enrollment](#) object

[Conclude, deactivate, or delete an enrollment EnrollmentsApiController#destroy](#)

DELETE /api/v1/courses/:course\_id/enrollments/id

Scope: url:DELETE/ /api/v1/courses/:course\_id/enrollments/:id

Conclude, deactivate, or delete an enrollment. If the task argument isn't given, the enrollment will be concluded.

Request Parameters:

Parameter	Type	Description
task	string	<p>The action to take on the enrollment. When inactive, a user will still appear in the course roster to admins, but be unable to participate. ("inactivate" and "deactivate" are equivalent tasks)</p> <p>Allowed values: conclude, delete, inactivate, deactivate</p>

Example Request:

```
curl https://<canvas>/api/v1/courses/:course_id/enrollments/enrollment_id \
-X DELETE \
-F 'task=conclude'
```

Returns an [Enrollment](#) object

[Accept Course Invitation EnrollmentsApiController#accept](#)

POST /api/v1/courses/:course\_id/enrollments/id/accept

Scope: url:POST/ /api/v1/courses/:course\_id/enrollments/:id/accept

accepts a pending course invitation for the current user

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/enrollments/:id/accept \
-X POST \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "success": true
}
```

[Reject Course Invitation EnrollmentsApiController#reject](#)

POST /api/v1/courses/:course\_id/enrollments/id/reject

Scope: url:POST/ /api/v1/courses/:course\_id/enrollments/:id/reject

rejects a pending course invitation for the current user

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/enrollments/:id/reject \
-X POST \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "success": true
}
```

[Re-activate an enrollment EnrollmentsApiController#reactivate](#)

PUT /api/v1/courses/:course\_id/enrollments/id/reactivate

Scope: url:PUT/ /api/v1/courses/:course\_id/enrollments/:id/reactivate

Activates an inactive enrollment

Example Request:

```
curl https://<canvas>/api/v1/courses/:course_id/enrollments/:enrollment_id/reactivate \
-X PUT
```

Returns an [Enrollment](#) object

[Add last attended date EnrollmentsApiController#last\\_attended](#)

PUT /api/v1/courses/:course\_id/users/:user\_id/last\_attended

Scope: url:PUT|/api/v1/courses/:course\_id/users/:user\_id/last\_attended

Add last attended date to student enrollment in course

Request Parameters:

Parameter	Type	Description
date	Date	The last attended date of a student enrollment in a course.

Example Request:

```
curl https://<canvas>/api/v1/courses/:course_id/users/:user_id/last_attended"
-X PUT => date="Thu%20Dec%2021%202017%2000:00:00%20GMT-0700%20(MST)"
```

Returns an [Enrollment](#) object

[Show Temporary Enrollment recipient and provider status EnrollmentsApiController#show\\_temporary\\_enrollment\\_status](#)

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

GET /api/v1/users/:user\_id/temporary\_enrollment\_status

Scope: url:GET|/api/v1/users/:user\_id/temporary\_enrollment\_status

Returns a JSON Object containing the temporary enrollment status for a user.

Request Parameters:

Parameter	Type	Description
account_id	string	The ID of the account to check for temporary enrollment status. Defaults to the domain root account if not provided.

Example Response:

```
{
  "is_provider": false, "is_recipient": true, "can_provide": false
}
```

[Get all ePortfolios for a User EportfoliosApiController#index](#)

GET /api/v1/users/:user\_id/eportfolios

Scope: url:GET|/api/v1/users/:user\_id/eportfolios

Get a list of all ePortfolios for the specified user.

Request Parameters:

Parameter	Type	Description
deleted	boolean	Include deleted ePortfolios. Only available to admins who can moderate_user_content.
include[]	string	Allowed values: deleted

Returns a list of [ePortfolio](#) objects

[Get an ePortfolio EportfoliosApiController#show](#)

GET /api/v1/eportfolios/:id

Scope: url:GET|/api/v1/eportfolios/:id

Get details for a single ePortfolio.

Returns an [ePortfolio](#) object

[Delete an ePortfolio EportfoliosApiController#delete](#)

DELETE /api/v1/eportfolios/:id

Scope: url:DELETE|/api/v1/eportfolios/:id

Mark an ePortfolio as deleted.

Returns an [ePortfolio](#) object

[Get ePortfolio Pages EportfoliosApiController#pages](#)

GET /api/v1/eportfolios/:eportfolio\_id/pages

Scope: url:GET|/api/v1/eportfolios/:eportfolio\_id/pages

Get details for the pages of an ePortfolio

Returns a list of [ePortfolioPage](#) objects

[Moderate an ePortfolio EportfoliosApiController#moderate](#)

PUT /api/v1/eportfolios/:eportfolio\_id/moderate

Scope: url:PUT|/api/v1/eportfolios/:eportfolio\_id/moderate

Update the spam\_status of an eportfolio. Only available to admins who can moderate\_user\_content.

Request Parameters:

Parameter	Type	Description
spam_status	string	The spam status for the ePortfolio
		Allowed values: marked_as_spam, marked_as_safe

Returns an [ePortfolio](#) object

[Moderate all ePortfolios for a User EportfoliosApiController#moderate\\_all](#)

PUT /api/v1/users/:user\_id/eportfolios

Scope: url:PUT|/api/v1/users/:user\_id/eportfolios

Update the spam\_status for all active eportfolios of a user. Only available to admins who can moderate\_user\_content.

Request Parameters:

Parameter	Type	Description
spam_status	string	The spam status for all the ePortfolios
		Allowed values: marked_as_spam, marked_as_safe

Restore a deleted ePortfolio EportfoliosApiController#restore

PUT /api/v1/eportfolios/:eportfolio\_id/restore

Scope: url:PUT|/api/v1/eportfolios/:eportfolio\_id/restore

Restore an ePortfolio back to active that was previously deleted. Only available to admins who can moderate\_user\_content.

Returns an ePortfolio object

List courses with their latest ePub export EpubExportsController#index

GET /api/v1/epub\_exports

Scope: url:GET|/api/v1/epub\_exports

A paginated list of all courses a user is actively participating in, and the latest ePub export associated with the user & course.

Returns a list of CourseEpubExport objects

Create ePub Export EpubExportsController#create

POST /api/v1/courses/:course\_id/epub\_exports

Scope: url:POST|/api/v1/courses/:course\_id/epub\_exports

Begin an ePub export for a course.

You can use the Progress API to track the progress of the export. The export's progress is linked to with the progress\_url value.

When the export completes, use the Show content export endpoint to retrieve a download URL for the exported content.

Returns an EpubExport object

Show ePub export EpubExportsController#show

GET /api/v1/courses/:course\_id/epub\_exports/id

Scope: url:GET|/api/v1/courses/:course\_id/epub\_exports/:id

Get information about a single ePub export.

Returns an EpubExport object

Create Error Report ErrorsController#create

POST /api/v1/error\_reports

Scope: url:POST|/api/v1/error\_reports

Create a new error report documenting an experienced problem

Performs the same action as when a user uses the “help -> report a problem” dialog.

Request Parameters:

Parameter	Type	Description
error[subject]	Required string	The summary of the problem
error[url]	string	URL from which the report was issued
error[email]	string	Email address for the reporting user
error[comments]	string	The long version of the story from the user one what they experienced
error[http_env]	SerializedHash	A collection of metadata about the users' environment. If not provided, canvas will collect it based on information found in the request. (Doesn't have to be HTTPENV info, could be anything JSON object that can be serialized as a hash, a mobile app might include relevant metadata for itself)

Example Request:

```
# Create error report
curl 'https://<canvas>/api/v1/error_reports' \
-X POST \
-F 'error[subject]="things are broken"' \
-F 'error[url]=http://<canvas>/courses/1' \
-F 'error[description]="All my thoughts on what I saw"' \
-H 'Authorization: Bearer <token>'
```

List external tools ExternalToolsController#index

GET /api/v1/courses/:course\_id/external\_tools

Scope: url:GET|/api/v1/courses/:course\_id/external\_tools

GET /api/v1/accounts/:account\_id/external\_tools

Scope: url:GET|/api/v1/accounts/:account\_id/external\_tools

GET /api/v1/groups/:group\_id/external\_tools

Scope: url:GET|/api/v1/groups/:group\_id/external\_tools

Returns the paginated list of external tools for the current context. See the get request docs for a single tool for a list of properties on an external tool.

Request Parameters:

Parameter	Type	Description
search_term	string	The partial name of the tools to match and return.
selectable	boolean	If true, then only tools that are meant to be selectable are returned.
include_parents	boolean	If true, then include tools installed in all accounts above the current context
placement	string	The placement type to filter by.

Example Request:

```
Return all tools at the current context as well as all tools from the parent, and filter the tools list to only those with a placement of 'editor_button'
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools?include_parents=true&placement=editor_button' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
[
  {
    "id": 1,
    "domain": "domain.example.com",
    "url": "http://www.example.com/lms/lti",
    "consumer_key": "key",
    "name": "LTI Tool",
    "description": "This is for cool things",
    "created_at": "2037-07-21T13:29:31Z",
    "updated_at": "2037-07-28T19:38:31Z",
    "privacy_level": "anonymous",
    "custom_fields": { "key": "value" },
    "is_rce_favorite": false,
    "is_top_nav_favorite": false,
    "account_navigation": {
      "canvas_icon_class": "icon-lti",
      "icon_url": "...",
      "text": "...",
      "url": "...",
      "label": "...",
      "selection_width": 50,
      "selection_height": 50
    },
    "assignment_selection": null,
    "course_home_sub_navigation": null,
    "course_navigation": {
      "canvas_icon_class": "icon-lti",
      "icon_url": "...",
      "text": "...",
      "url": "...",
      "default": "disabled",
      "enabled": "true",
      "visibility": "public",
      "windowTarget": "_blank"
    },
    "editor_button": {
      "canvas_icon_class": "icon-lti",
      "icon_url": "...",
      "message_type": "ContentItemSelectionRequest",
      "text": "...",
      "url": "...",
      "label": "...",
      "selection_width": 50,
      "selection_height": 50
    },
    "homework_submission": null,
    "link_selection": null,
    "migration_selection": null,
    "resource_selection": null,
    "tool_configuration": null,
    "user_navigation": null,
    "selection_width": 500,
    "selection_height": 500,
    "icon_url": "...",
    "not_selectable": false,
    "deployment_id": null,
    "unified_tool_id": null
  },
  { ... }
]
```

Get a sessionless launch url for an external tool. ExternalToolsController#generate\_sessionless\_launch

GET /api/v1/courses/:course\_id/external\_tools/sessionless\_launch

Scope: url:GET /api/v1/courses/:course\_id/external\_tools/sessionless\_launch

GET /api/v1/accounts/:account\_id/external\_tools/sessionless\_launch

Scope: url:GET /api/v1/accounts/:account\_id/external\_tools/sessionless\_launch

Returns a sessionless launch url for an external tool. Prefers the resource\_link\_lookup\_uuid, but defaults to the other passed parameters id, url, and launch\_type

NOTE: Either the resource\_link\_lookup\_uuid, id, or url must be provided unless launch\_type is assessment or module\_item.

Request Parameters:

Parameter	Type	Description
id	string	The external id of the tool to launch.
url	string	The LTI launch url for the external tool.
assignment_id	string	The assignment id for an assignment launch. Required if launch_type is set to "assessment".
module_item_id	string	The assignment id for a module item launch. Required if launch_type is set to "module_item".
launch_type	string	The type of launch to perform on the external tool. Placement names (eg. "course_navigation") can also be specified to use the custom launch url for that placement; if done, the tool id must be provided. Allowed values: assessment, module_item
resource_link_lookup_uuid	string	The identifier to lookup a resource link.

API response field:

- id  
The id for the external tool to be launched.
- name  
The name of the external tool to be launched.
- url  
The url to load to launch the external tool for the user.

Example Request:

```
Finds the tool by id and returns a sessionless launch url
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/sessionless_launch' \
-H 'Authorization: Bear <token>' \
-F 'id=<external_tool_id>'

Finds the tool by launch url and returns a sessionless launch url
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/sessionless_launch' \
-H 'Authorization: Bearer <token>' \
-F 'url=<lti launch url>'

Finds the tool associated with a specific assignment and returns a sessionless launch url
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/sessionless_launch' \
-H 'Authorization: Bearer <token>' \
-F 'launch_type=assessment' \
-F 'assignment_id=<assignment_id>'

Finds the tool associated with a specific module item and returns a sessionless launch url
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/sessionless_launch' \
-H 'Authorization: Bearer <token>' \
-F 'launch_type=module_item' \
-F 'module_item_id=<module_item_id>'

Finds the tool by id and returns a sessionless launch url for a specific placement
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/sessionless_launch' \
-H 'Authorization: Bearer <token>' \
-F 'id=<external_tool_id>' \
-F 'launch_type=<placement_name>'
```

Get a single external tool ExternalToolsController#show

GET /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

Scope: url:GET /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

GET /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Scope: url:GET /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Returns the specified external tool.

API response field:

- id  
The unique identifier for the tool
- domain  
The domain to match links against
- url  
The url to match links against
- consumer\_key  
The consumer key used by the tool (The associated shared secret is not returned)
- name  
The name of the tool
- description  
A description of the tool
- created\_at  
Timestamp of creation
- updated\_at  
Timestamp of last update
- privacy\_level  
How much user information to send to the external tool: "anonymous", "name\_only", "email\_only", "public"
- custom\_fields  
Custom fields that will be sent to the tool consumer
- is\_rce\_favorite  
Boolean determining whether this tool should be in a preferred location in the RCE.
- is\_top\_nav\_favorite  
Boolean determining whether this tool should have a dedicated button in Top Navigation.
- account\_navigation  
The configuration for account navigation links (see create API for values)
- assignment\_selection  
The configuration for assignment selection links (see create API for values)
- course\_home\_sub\_navigation  
The configuration for course home navigation links (see create API for values)
- course\_navigation  
The configuration for course navigation links (see create API for values)
- editor\_button  
The configuration for a WYSIWYG editor button (see create API for values)
- homework\_submission  
The configuration for homework submission selection (see create API for values)
- link\_selection  
The configuration for link selection (see create API for values)
- migration\_selection  
The configuration for migration selection (see create API for values)
- resource\_selection  
The configuration for a resource selector in modules (see create API for values)
- tool\_configuration  
The configuration for a tool configuration link (see create API for values)
- user\_navigation  
The configuration for user navigation links (see create API for values)
- selection\_width  
The pixel width of the iFrame that the tool will be rendered in
- selection\_height  
The pixel height of the iFrame that the tool will be rendered in
- icon\_url  
The url for the tool icon
- not\_selectable  
whether the tool is not selectable from assignment and modules
- unified\_tool\_id  
The unique identifier for the tool in LearnPlatform
- deployment\_id  
The unique identifier for the deployment of the tool

Example Response:

```
{
  "id": 1,
  "domain": "domain.example.com",
  "url": "http://www.example.com/ims/lti",
  "consumer_key": "key",
  "name": "LTI Tool",
  "description": "This is for cool things",
  "created_at": "2037-07-21T13:29:31Z",
  "updated_at": "2037-07-28T19:38:31Z",
  "privacy_level": "anonymous",
  "custom_fields": {"key": "value"},
  "account_navigation": {
    "canvas_icon_class": "icon-lti",
    "icon_url": "...",
    "text": "...",
    "url": "...",
    "label": "...",
    "selection_width": 50,
    "selection_height": 50
  },
  "assignment_selection": null,
  "course_home_sub_navigation": null,
  "course_navigation": {
    "canvas_icon_class": "icon-lti",
    "icon_url": "...",
    "text": "...",
    "url": "...",
    "default": "disabled",
    "enabled": "true",
    "visibility": "public",
```

```
    "windowTarget": "_blank"
  },
  "editor_button": {
    "canvas_icon_class": "icon-lti",
    "icon_url": "...",
    "message_type": "ContentItemSelectionRequest",
    "text": "...",
    "url": "...",
    "label": "...",
    "selection_width": 50,
    "selection_height": 50
  },
  "homework_submission": null,
  "link_selection": null,
  "migration_selection": null,
  "resource_selection": null,
  "tool_configuration": null,
  "user_navigation": {
    "canvas_icon_class": "icon-lti",
    "icon_url": "...",
    "text": "...",
    "url": "...",
    "default": "disabled",
    "enabled": "true",
    "visibility": "public",
    "windowTarget": "_blank"
  },
  "selection_width": 500,
  "selection_height": 500,
  "icon_url": "...",
  "not_selectable": false
}
```

Create an external tool ExternalToolsController#create

POST /api/v1/courses/:course\_id/external\_tools

Scope: url:POST|/api/v1/courses/:course\_id/external\_tools

POST /api/v1/accounts/:account\_id/external\_tools

Scope: url:POST|/api/v1/accounts/:account\_id/external\_tools

Create an external tool in the specified course/account. The created tool will be returned, see the “show” endpoint for an example. If a client ID is supplied canvas will attempt to create a context external tool using the LTI 1.3 standard.

Request Parameters:

Parameter	Type	Description
client_id	Required string	The client id is attached to the developer key. If supplied all other parameters are unnecessary and will be ignored
name	Required string	The name of the tool
privacy_level	Required string	How much user information to send to the external tool. Allowed values: anonymous, name_only, email_only, public
consumer_key	Required string	The consumer key for the external tool
shared_secret	Required string	The shared secret with the external tool
description	string	A description of the tool
url	string	The url to match links against. Either “url” or “domain” should be set, not both.
domain	string	The domain to match links against. Either “url” or “domain” should be set, not both.
icon_url	string	The url of the icon to show for this tool
text	string	The default text to show for this tool
custom_fields[field_name]	string	Custom fields that will be sent to the tool consumer; can be used multiple times
is_rce_favorite	boolean	(Deprecated in favor of <a href="#">Add tool to RCE Favorites</a> and <a href="#">Remove tool from RCE Favorites</a> ) Whether this tool should appear in a preferred location in the RCE. This only applies to tools in root account contexts that have an editor button placement.
account_navigation[url]	string	The url of the external tool for account navigation
account_navigation[enabled]	boolean	Set this to enable this feature
account_navigation[text]	string	The text that will show on the left-tab in the account navigation
account_navigation[selection_width]	string	The width of the dialog the tool is launched in
account_navigation[selection_height]	string	The height of the dialog the tool is launched in
account_navigation[display_type]	string	The layout type to use when launching the tool. Must be “full_width”, “full_width_in_context”, “in_nav_context”, “borderless”, or “default”
user_navigation[url]	string	The url of the external tool for user navigation
user_navigation[enabled]	boolean	Set this to enable this feature
user_navigation[text]	string	The text that will show on the left-tab in the user navigation
user_navigation[visibility]	string	Who will see the navigation tab. “admins” for admins, “public” or “members” for everyone. Setting this to ‘null’ will remove this configuration and use the default behavior, which is “public”. Allowed values: admins, members, public
course_home_sub_navigation[url]	string	The url of the external tool for right-side course home navigation menu
course_home_sub_navigation[enabled]	boolean	Set this to enable this feature
course_home_sub_navigation[text]	string	The text that will show on the right-side course home navigation menu
course_home_sub_navigation[icon_url]	string	The url of the icon to show in the right-side course home navigation menu
course_navigation[enabled]	boolean	Set this to enable this feature
course_navigation[text]	string	The text that will show on the left-tab in the course navigation
course_navigation[visibility]	string	Who will see the navigation tab. “admins” for course admins, “members” for students, “public” for everyone. Setting this to ‘null’ will remove this configuration and use the default behavior, which is “public”. Allowed values: admins, members, public
course_navigation>windowTarget]	string	Determines how the navigation tab will be opened. “_blank” Launches the external tool in a new window or tab. “_self” (Default) Launches the external tool in an iframe inside of Canvas.



Parameter	Type	Description
		Allowed values: <code>_blank</code> , <code>_self</code>
		If set to “disabled” the tool will not appear in the course navigation until a teacher explicitly enables it.
<code>course_navigation[default]</code>	string	If set to “enabled” the tool will appear in the course navigation without requiring a teacher to explicitly enable it. defaults to “enabled”  Allowed values: <code>disabled</code> , <code>enabled</code>
<code>course_navigation[display_type]</code>	string	The layout type to use when launching the tool. Must be “full_width”, “full_width_in_context”, “in_nav_context”, “borderless”, or “default”
<code>editor_button[url]</code>	string	The url of the external tool
<code>editor_button[enabled]</code>	boolean	Set this to enable this feature
<code>editor_button[icon_url]</code>	string	The url of the icon to show in the WYSIWYG editor
<code>editor_button[selection_width]</code>	string	The width of the dialog the tool is launched in
<code>editor_button[selection_height]</code>	string	The height of the dialog the tool is launched in
<code>editor_button[message_type]</code>	string	Set this to <code>ContentItemSelectionRequest</code> to tell the tool to use content-item; otherwise, omit
<code>homework_submission[url]</code>	string	The url of the external tool
<code>homework_submission[enabled]</code>	boolean	Set this to enable this feature
<code>homework_submission[text]</code>	string	The text that will show on the homework submission tab
<code>homework_submission[message_type]</code>	string	Set this to <code>ContentItemSelectionRequest</code> to tell the tool to use content-item; otherwise, omit
<code>link_selection[url]</code>	string	The url of the external tool
<code>link_selection[enabled]</code>	boolean	Set this to enable this feature
<code>link_selection[text]</code>	string	The text that will show for the link selection text
<code>link_selection[message_type]</code>	string	Set this to <code>ContentItemSelectionRequest</code> to tell the tool to use content-item; otherwise, omit
<code>migration_selection[url]</code>	string	The url of the external tool
<code>migration_selection[enabled]</code>	boolean	Set this to enable this feature
<code>migration_selection[message_type]</code>	string	Set this to <code>ContentItemSelectionRequest</code> to tell the tool to use content-item; otherwise, omit
<code>tool_configuration[url]</code>	string	The url of the external tool
<code>tool_configuration[enabled]</code>	boolean	Set this to enable this feature
<code>tool_configuration[message_type]</code>	string	Set this to <code>ContentItemSelectionRequest</code> to tell the tool to use content-item; otherwise, omit
<code>tool_configuration[prefer_sis_email]</code>	boolean	Set this to default the <code>lis_person_contact_email_primary</code> to prefer provisioned <code>sis_email</code> ; otherwise, omit
<code>resource_selection[url]</code>	string	The url of the external tool
<code>resource_selection[enabled]</code>	boolean	Set this to enable this feature. If set to false, <code>not_selectable</code> must also be set to true in order to hide this tool from the selection UI in modules and assignments.
<code>resource_selection[icon_url]</code>	string	The url of the icon to show in the module external tool list
<code>resource_selection[selection_width]</code>	string	The width of the dialog the tool is launched in
<code>resource_selection[selection_height]</code>	string	The height of the dialog the tool is launched in
<code>config_type</code>	string	Configuration can be passed in as CC xml instead of using query parameters. If this value is “by_url” or “by_xml” then an xml configuration will be expected in either the “config_xml” or “config_url” parameter. Note that the name parameter overrides the tool name provided in the xml
<code>config_xml</code>	string	XML tool configuration, as specified in the CC xml specification. This is required if “config_type” is set to “by_xml”
<code>config_url</code>	string	URL where the server can retrieve an XML tool configuration, as specified in the CC xml specification. This is required if “config_type” is set to “by_url”
<code>not_selectable</code>	boolean	Default: false. If set to true, and if <code>resource_selection</code> is set to false, the tool won’t show up in the external tool selection UI in modules and assignments
<code>oauth_compliant</code>	boolean	Default: false, if set to true LTI query params will not be copied to the post body.
<code>unified_tool_id</code>	string	The unique identifier for the tool in LearnPlatform

Example Request:

```
This would create a tool on this course with two custom fields and a course navigation tab
curl -X POST 'https://<canvas>/api/v1/courses/<course_id>/external_tools' \
-H 'Authorization: Bearer <token>' \
-F 'name=LTI Example' \
-F 'consumer_key=asdfg' \
-F 'shared_secret=lkjh' \
-F 'url=https://example.com/ims/lti' \
-F 'privacy_level=name_only' \
-F 'custom_fields[key1]=value1' \
-F 'custom_fields[key2]=value2' \
-F 'course_navigation[text]=Course Materials' \
-F 'course_navigation[enabled]=true'

This would create a tool on the account with navigation for the user profile page
curl -X POST 'https://<canvas>/api/v1/accounts/<account_id>/external_tools' \
-H 'Authorization: Bearer <token>' \
-F 'name=LTI Example' \
-F 'consumer_key=asdfg' \
-F 'shared_secret=lkjh' \
-F 'url=https://example.com/ims/lti' \
-F 'privacy_level=name_only' \
-F 'user_navigation[url]=https://example.com/ims/lti/user_endpoint' \
-F 'user_navigation[text]=Something Cool' \
-F 'user_navigation[enabled]=true'

This would create a tool on the account with configuration pulled from an external URL
curl -X POST 'https://<canvas>/api/v1/accounts/<account_id>/external_tools' \
-H 'Authorization: Bearer <token>' \
-F 'name=LTI Example' \
-F 'consumer_key=asdfg' \
-F 'shared_secret=lkjh' \
-F 'config_type=by_url' \
-F 'config_url=https://example.com/ims/lti/tool_config.xml'
```

Edit an external tool ExternalToolsController#update

PUT /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

Scope: url:PUT /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

PUT /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Scope: url:PUT /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Update the specified external tool. Uses same parameters as create

Example Request:

This would update the specified keys on this external tool  
curl -X PUT "https://<canvas>/api/v1/courses/<course\_id>/external\_tools/<external\_tool\_id>" \  
-H "Authorization: Bearer <token>" \  
-F "name=Public Example" \  
-F "privacy\_level=public" \

Delete an external tool ExternalToolsController#destroy

DELETE /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

Scope: url:DELETE /api/v1/courses/:course\_id/external\_tools/:external\_tool\_id

DELETE /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Scope: url:DELETE /api/v1/accounts/:account\_id/external\_tools/:external\_tool\_id

Remove the specified external tool

Example Request:

This would delete the specified external tool  
curl -X DELETE "https://<canvas>/api/v1/courses/<course\_id>/external\_tools/<external\_tool\_id>" \  
-H "Authorization: Bearer <token>" \

Add tool to RCE Favorites ExternalToolsController#add\_rce\_favorite

POST /api/v1/accounts/:account\_id/external\_tools/rce\_favorites/:id

Scope: url:POST /api/v1/accounts/:account\_id/external\_tools/rce\_favorites/:id

Add the specified editor\_button external tool to a preferred location in the RCE for courses in the given account and its subaccounts (if the subaccounts haven't set their own RCE Favorites). Cannot set more than 2 RCE Favorites.

Example Request:

curl -X POST "https://<canvas>/api/v1/accounts/<account\_id>/external\_tools/rce\_favorites/<id>" \  
-H "Authorization: Bearer <token>" \

Remove tool from RCE Favorites ExternalToolsController#remove\_rce\_favorite

DELETE /api/v1/accounts/:account\_id/external\_tools/rce\_favorites/:id

Scope: url:DELETE /api/v1/accounts/:account\_id/external\_tools/rce\_favorites/:id

Remove the specified external tool from a preferred location in the RCE for the given account

Example Request:

curl -X DELETE "https://<canvas>/api/v1/accounts/<account\_id>/external\_tools/rce\_favorites/<id>" \  
-H "Authorization: Bearer <token>" \

Add tool to Top Navigation Favorites ExternalToolsController#add\_top\_nav\_favorite

POST /api/v1/accounts/:account\_id/external\_tools/top\_nav\_favorites/:id

Scope: url:POST /api/v1/accounts/:account\_id/external\_tools/top\_nav\_favorites/:id

Adds a dedicated button in Top Navigation for the specified tool for the given account. Cannot set more than 2 top\_navigation Favorites.

Example Request:

curl -X POST "https://<canvas>/api/v1/accounts/<account\_id>/external\_tools/top\_nav\_favorites/<id>" \  
-H "Authorization: Bearer <token>" \

Remove tool from Top Navigation Favorites ExternalToolsController#remove\_top\_nav\_favorite

DELETE /api/v1/accounts/:account\_id/external\_tools/top\_nav\_favorites/:id

Scope: url:DELETE /api/v1/accounts/:account\_id/external\_tools/top\_nav\_favorites/:id

Removes the dedicated button in Top Navigation for the specified tool for the given account.

Example Request:

curl -X DELETE "https://<canvas>/api/v1/accounts/<account\_id>/external\_tools/top\_nav\_favorites/<id>" \  
-H "Authorization: Bearer <token>" \

Get visible course navigation tools ExternalToolsController#all\_visible\_nav\_tools

GET /api/v1/external\_tools/visible\_course\_nav\_tools

Scope: url:GET /api/v1/external\_tools/visible\_course\_nav\_tools

Get a list of external tools with the course\_navigation placement that have not been hidden in course settings and whose visibility settings apply to the requesting user. These tools are the same that appear in the course navigation.

The response format is the same as for List external tools, but with additional context\_id and context\_name fields on each element in the array.

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

context\_codes[] Required string List of context\_codes to retrieve visible course nav tools for (for example, course\_123). Only courses are presently supported.

API response field:

- context\_id  
The unique identifier of the associated context
- context\_name  
The name of the associated context

Example Request:

curl "https://<canvas>/api/v1/external\_tools/visible\_course\_nav\_tools?context\_codes[]=course\_5" \  
-H "Authorization: Bearer <token>" \

Example Response:

```
[[{"id": 1, "domain": "domain.example.com", "url": "http://www.example.com/ims/lti", "context_id": 5, "context_name": "Example Course", ... }, { ... }]]
```

Get visible course navigation tools for a single course ExternalToolsController#visible\_course\_nav\_tools

GET /api/v1/courses/:course\_id/external\_tools/visible\_course\_nav\_tools

**Scope:** url:GET[/api/v1/courses/:course\_id/external\_tools/visible\_course\_nav\_tools]

Get a list of external tools with the course\_navigation placement that have not been hidden in course settings and whose visibility settings apply to the requesting user. These tools are the same that appear in the course navigation. The response format is the same as Get visible course navigation tools.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/external_tools/visible_course_nav_tools' \
-H 'Authorization: Bearer <token>'
```

List favorite courses FavoritesController#list\_favorite\_courses

GET /api/v1/users/self/favorites/courses

**Scope:** url:GET[/api/v1/users/self/favorites/courses]

Retrieve the paginated list of favorite courses for the current user. If the user has not chosen any favorites, then a selection of currently enrolled courses will be returned.

See the [List courses API](#) for details on accepted include[] parameters.

Request Parameters:

Parameter	Type	Description
exclude_blueprint_courses	boolean	When set, only return courses that are not configured as blueprint courses.

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/courses \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a list of [Course](#) objects

List favorite groups FavoritesController#list\_favorite\_groups

GET /api/v1/users/self/favorites/groups

**Scope:** url:GET[/api/v1/users/self/favorites/groups]

Retrieve the paginated list of favorite groups for the current user. If the user has not chosen any favorites, then a selection of groups that the user is a member of will be returned.

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/groups \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a list of [Group](#) objects

Add course to favorites FavoritesController#add\_favorite\_course

POST /api/v1/users/self/favorites/courses/:id

**Scope:** url:POST[/api/v1/users/self/favorites/courses/:id]

Add a course to the current user's favorites. If the course is already in the user's favorites, nothing happens. Canvas for Elementary subject and homeroom courses can be added to favorites, but this has no effect in the UI.

Request Parameters:

Parameter	Type	Description
id	Required string	The ID or SIS ID of the course to add. The current user must be registered in the course.

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/courses/1170 \
-X POST \
-H 'Authorization: Bearer <ACCESS_TOKEN>' \
-H 'Content-Length: 0'
```

Returns a [Favorite](#) object

Add group to favorites FavoritesController#add\_favorite\_groups

POST /api/v1/users/self/favorites/groups/:id

**Scope:** url:POST[/api/v1/users/self/favorites/groups/:id]

Add a group to the current user's favorites. If the group is already in the user's favorites, nothing happens.

Request Parameters:

Parameter	Type	Description
id	Required string	The ID or SIS ID of the group to add. The current user must be a member of the group.

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/group/1170 \
-X POST \
-H 'Authorization: Bearer <ACCESS_TOKEN>' \
-H 'Content-Length: 0'
```

Returns a [Favorite](#) object

Remove course from favorites FavoritesController#remove\_favorite\_course

DELETE /api/v1/users/self/favorites/courses/:id

**Scope:** url:DELETE[/api/v1/users/self/favorites/courses/:id]

Remove a course from the current user's favorites.

Request Parameters:

Parameter	Type	Description
id	Required string	the ID or SIS ID of the course to remove

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/courses/1170 \
-X DELETE \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a [Favorite](#) object

Remove group from favorites FavoritesController#remove\_favorite\_groups

DELETE /api/v1/users/self/favorites/groups/:id

**Scope:** url:DELETE[/api/v1/users/self/favorites/groups/:id]

Remove a group from the current user's favorites.

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
id	Required string	the ID or SIS ID of the group to remove

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/groups/1170 \
-X DELETE \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

Returns a [Favorite](#) object

[Reset course favorites FavoritesController#reset\\_course\\_favorites](#)

**DELETE /api/v1/users/self/favorites/courses**

**Scope:** url:DELETE|/api/v1/users/self/favorites/courses

Reset the current user's course favorites to the default automatically generated list of enrolled courses

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/courses \
-X DELETE \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

[Reset group favorites FavoritesController#reset\\_groups\\_favorites](#)

**DELETE /api/v1/users/self/favorites/groups**

**Scope:** url:DELETE|/api/v1/users/self/favorites/groups

Reset the current user's group favorites to the default automatically generated list of enrolled group

Example Request:

```
curl https://<canvas>/api/v1/users/self/favorites/group \
-X DELETE \
-H 'Authorization: Bearer <ACCESS_TOKEN>'
```

[List features FeatureFlagsController#index](#)

**GET /api/v1/courses/:course\_id/features**

**Scope:** url:GET|/api/v1/courses/:course\_id/features

**GET /api/v1/accounts/:account\_id/features**

**Scope:** url:GET|/api/v1/accounts/:account\_id/features

**GET /api/v1/users/:user\_id/features**

**Scope:** url:GET|/api/v1/users/:user\_id/features

A paginated list of all features that apply to a given Account, Course, or User.

Example Request:

```
curl 'http://<canvas>/api/v1/courses/1/features' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Feature](#) objects

[List enabled features FeatureFlagsController#enabled\\_features](#)

**GET /api/v1/courses/:course\_id/features/enabled**

**Scope:** url:GET|/api/v1/courses/:course\_id/features/enabled

**GET /api/v1/accounts/:account\_id/features/enabled**

**Scope:** url:GET|/api/v1/accounts/:account\_id/features/enabled

**GET /api/v1/users/:user\_id/features/enabled**

**Scope:** url:GET|/api/v1/users/:user\_id/features/enabled

A paginated list of all features that are enabled on a given Account, Course, or User. Only the feature names are returned.

Example Request:

```
curl 'http://<canvas>/api/v1/courses/1/features/enabled' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
[ "fancy_wickets", "automatic_essay_grading", "telepathic_navigation" ]
```

[List environment features FeatureFlagsController#environment](#)

**GET /api/v1/features/environment**

**Scope:** url:GET|/api/v1/features/environment

Return a hash of global feature options that pertain to the Canvas user interface. This is the same information supplied to the web interface as ENV.FEATURES.

Example Request:

```
curl 'http://<canvas>/api/v1/features/environment' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{ "telepathic_navigation": true, "fancy_wickets": true, "automatic_essay_grading": false }
```

[Get feature flag FeatureFlagsController#show](#)

**GET /api/v1/courses/:course\_id/features/flags/:feature**

**Scope:** url:GET|/api/v1/courses/:course\_id/features/flags/:feature

**GET /api/v1/accounts/:account\_id/features/flags/:feature**

**Scope:** url:GET|/api/v1/accounts/:account\_id/features/flags/:feature

**GET /api/v1/users/:user\_id/features/flags/:feature**

**Scope:** url:GET|/api/v1/users/:user\_id/features/flags/:feature

Get the feature flag that applies to a given Account, Course, or User. The flag may be defined on the object, or it may be inherited from a parent account. You can look at the context\_id and context\_type of the returned object to determine which is the case. If these fields are missing, then the object is the global Canvas default.

Example Request:

```
curl 'http://<canvas>/api/v1/courses/1/features/flags/fancy_wickets' \
-H 'Authorization: Bearer <token>'
```

Returns a [FeatureFlag](#) object

[Set feature flag FeatureFlagsController#update](#)

**PUT /api/v1/courses/:course\_id/features/flags/:feature**

**Scope:** url:PUT /api/v1/courses/:course\_id/features/flags/:feature

**PUT /api/v1/accounts/:account\_id/features/flags/:feature**

**Scope:** url:PUT /api/v1/accounts/:account\_id/features/flags/:feature

**PUT /api/v1/users/:user\_id/features/flags/:feature**

**Scope:** url:PUT /api/v1/users/:user\_id/features/flags/:feature

Set a feature flag for a given Account, Course, or User. This call will fail if a parent account sets a feature flag for the same feature in any state other than “allowed”.

**Request Parameters:**

Parameter	Type	Description
		“off”
		The feature is not available for the course, user, or account and sub-accounts.
		“allowed”
state	string	(valid only on accounts) The feature is off in the account, but may be enabled in sub-accounts and courses by setting a feature flag on the sub-account or course.
		“on”
		The feature is turned on unconditionally for the user, course, or account and sub-accounts.
		Allowed values: off, allowed, on

**Example Request:**

```
curl -X PUT 'http://<canvas>/api/v1/courses/1/features/flags/fancy_wickets' \
-H "Authorization: Bearer " \
-F "state=on"
```

Returns a [FeatureFlag](#) object

**[Remove feature flag FeatureFlagsController#delete](#)**

**DELETE /api/v1/courses/:course\_id/features/flags/:feature**

**Scope:** url:DELETE /api/v1/courses/:course\_id/features/flags/:feature

**DELETE /api/v1/accounts/:account\_id/features/flags/:feature**

**Scope:** url:DELETE /api/v1/accounts/:account\_id/features/flags/:feature

**DELETE /api/v1/users/:user\_id/features/flags/:feature**

**Scope:** url:DELETE /api/v1/users/:user\_id/features/flags/:feature

Remove feature flag for a given Account, Course, or User. (Note that the flag must be defined on the Account, Course, or User directly.) The object will then inherit the feature flags from a higher account, if any exist. If this flag was ‘on’ or ‘off’, then lower-level account flags that were masked by this one will apply again.

**Example Request:**

```
curl -X DELETE 'http://<canvas>/api/v1/courses/1/features/flags/fancy_wickets' \
-H "Authorization: Bearer <token>"
```

Returns a [FeatureFlag](#) object

**[Get quota information FilesController#api\\_quota](#)**

**GET /api/v1/courses/:course\_id/files/quota**

**Scope:** url:GET /api/v1/courses/:course\_id/files/quota

**GET /api/v1/groups/:group\_id/files/quota**

**Scope:** url:GET /api/v1/groups/:group\_id/files/quota

**GET /api/v1/users/:user\_id/files/quota**

**Scope:** url:GET /api/v1/users/:user\_id/files/quota

Returns the total and used storage quota for the course, group, or user.

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/1/files/quota' \
-H "Authorization: Bearer <token>"
```

**Example Response:**

```
{ "quota": 524288000, "quota_used": 402653184 }
```

**[List files FilesController#api\\_index](#)**

**GET /api/v1/courses/:course\_id/files**

**Scope:** url:GET /api/v1/courses/:course\_id/files

**GET /api/v1/users/:user\_id/files**

**Scope:** url:GET /api/v1/users/:user\_id/files

**GET /api/v1/groups/:group\_id/files**

**Scope:** url:GET /api/v1/groups/:group\_id/files

**GET /api/v1/folders/:id/files**

**Scope:** url:GET /api/v1/folders/:id/files

Returns the paginated list of files for the folder or course.

**Request Parameters:**

Parameter	Type	Description
content_types[]	string	Filter results by content-type. You can specify type/subtype pairs (e.g., ‘image/jpeg’), or simply types (e.g., ‘image’, which will match ‘image/gif’, ‘image/jpeg’, etc.).
exclude_content_types[]	string	Exclude given content-types from your results. You can specify type/subtype pairs (e.g., ‘image/jpeg’), or simply types (e.g., ‘image’, which will match ‘image/gif’, ‘image/jpeg’, etc.).
search_term	string	The partial name of the files to match and return.
		Array of additional information to include.
		“user”
		the user who uploaded the file or last edited its content
include[]	string	“usage_rights”
		copyright and license information for the file (see UsageRights)
		Allowed values: user

Parameter	Type	Description
only[]	Array	Array of information to restrict to. Overrides include[]
		"names"
		only returns file name information
sort	string	Sort results by this field. Defaults to 'name'. Note that 'sort=user' implies 'include[]=user'.
		Allowed values: name, size, created_at, updated_at, content_type, user
order	string	The sorting order. Defaults to 'asc'.
		Allowed values: asc, desc

Example Request:

```
curl 'https://<canvas>/api/v1/folders/<folder_id>/files?content_types[]=image&content_types[]=text/plain' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [File](#) objects

[Get public inline preview url FilesController#public\\_url](#)

GET /api/v1/files/:id/public\_url

Scope: url:GET|/api/v1/files/:id/public\_url

Determine the URL that should be used for inline preview of the file.

Request Parameters:

Parameter	Type	Description
submission_id	integer	The id of the submission the file is associated with. Provide this argument to gain access to a file that has been submitted to an assignment (Canvas will verify that the file belongs to the submission and the calling user has rights to view the submission).

Example Request:

```
curl 'https://<canvas>/api/v1/files/:id/public_url' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{ "public_url": "https://example-bucket.s3.amazonaws.com/example-namespaces/attachments/:id/example-filename?AWSAccessKeyId=example-key&Expires=1400000000&Signature=example-signature" }
```

[Get file FilesController#api\\_show](#)

GET /api/v1/files/:id

Scope: url:GET|/api/v1/files/:id

POST /api/v1/files/:id

Scope: url:POST|/api/v1/files/:id

GET /api/v1/courses/:course\_id/files/:id

Scope: url:GET|/api/v1/courses/:course\_id/files/:id

GET /api/v1/groups/:group\_id/files/:id

Scope: url:GET|/api/v1/groups/:group\_id/files/:id

GET /api/v1/users/:user\_id/files/:id

Scope: url:GET|/api/v1/users/:user\_id/files/:id

Returns the standard attachment json object

Request Parameters:

Parameter	Type	Description
include[]	string	Array of additional information to include.
		"user"
		the user who uploaded the file or last edited its content
		"usage_rights"
		copyright and license information for the file (see UsageRights)
replacement_chain_context_type	string	Allowed values: user
		When a user replaces a file during upload, Canvas keeps track of the "replacement chain."
		Include this parameter if you wish Canvas to follow the replacement chain if the requested file was deleted and replaced by another.
replacement_chain_context_id	integer	Must be set to 'course' or 'account'. The "replacement_chain_context_id" parameter must also be included.
		When a user replaces a file during upload, Canvas keeps track of the "replacement chain."
		Include this parameter if you wish Canvas to follow the replacement chain if the requested file was deleted and replaced by another.
replacement_chain_context_id	integer	Indicates the context ID Canvas should use when following the "replacement chain." The "replacement_chain_context_type" parameter must also be included.

Example Request:

```
curl 'https://<canvas>/api/v1/files/<file_id>' \
-H 'Authorization: Bearer <token>'

curl 'https://<canvas>/api/v1/courses/<course_id>/files/<file_id>' \
-H 'Authorization: Bearer <token>'
```

Returns a [File](#) object

[Translate file reference FilesController#file\\_ref](#)

GET /api/v1/courses/:course\_id/files/file\_ref/:migration\_id

Scope: url:GET|/api/v1/courses/:course\_id/files/file\_ref/:migration\_id

Get information about a file from a course copy file reference

Example Request:

```
curl https://<canvas>/api/v1/courses/:id/files/file_ref/:id?file_ref=:id \
-H 'Authorization: Bearer <token>'
```

Returns a [File](#) object

[Update file FilesController#api\\_update](#)

PUT /api/v1/files/:id

**Scope:** url:PUT|/api/v1/files/:id

Update some settings on the specified file

**Request Parameters:**

Parameter	Type	Description
name	string	The new display name of the file, with a limit of 255 characters.
parent_folder_id	string	The id of the folder to move this file into. The new folder must be in the same context as the original parent folder. If the file is in a context without folders this does not apply.  If the file is moved to a folder containing a file with the same name, or renamed to a name matching an existing file, the API call will fail unless this parameter is supplied.  "overwrite"  Replace the existing file with the same name  "rename"  Add a qualifier to make the new filename unique  Allowed values: overwrite, rename
on_duplicate	string	
lock_at	DateTime	The datetime to lock the file at
unlock_at	DateTime	The datetime to unlock the file at
locked	boolean	Flag the file as locked
hidden	boolean	Flag the file as hidden
visibility_level	string	Configure which roles can access this file

**Example Request:**

```
curl -X PUT 'https://<canvas>/api/v1/files/<file_id>' \  
-F 'name=<new_name>' \  
-F 'locked=true' \  
-H 'Authorization: Bearer <token>'
```

Returns a [File](#) object

[Delete file FilesController#destroy](#)

**DELETE** /api/v1/files/:id

**Scope:** url:DELETE|/api/v1/files/:id

Remove the specified file. Unlike most other DELETE endpoints, using this endpoint will result in comprehensive, irretrievable destruction of the file. It should be used with the 'replace' parameter set to true in cases where the file preview also needs to be destroyed (such as to remove files that violate privacy laws).

**Request Parameters:**

Parameter	Type	Description
replace	boolean	This action is irreversible. If replace is set to true the file contents will be replaced with a generic "file has been removed" file. This also destroys any previews that have been generated for the file. Must have manage files and become other users permissions

**Example Request:**

```
curl -X DELETE 'https://<canvas>/api/v1/files/<file_id>' \  
-H 'Authorization: Bearer <token>'
```

Returns a [File](#) object

[Get icon metadata FilesController#icon\\_metadata](#)

**GET** /api/v1/files/:id/icon\_metadata

**Scope:** url:GET|/api/v1/files/:id/icon\_metadata

Returns the icon maker file attachment metadata

**Example Request:**

```
curl 'https://<canvas>/api/v1/courses/1/files/1/metadata' \  
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{  
  "type": "image/svg+xml-icon-maker-icons",  
  "alt": "",  
  "shape": "square",  
  "size": "small",  
  "color": "#FFFFFF",  
  "outlineColor": "#65499D",  
  "outlineSize": "large",  
  "text": "Hello",  
  "textSize": "x-large",  
  "textColor": "#65499D",  
  "textBackgroundColor": "#FFFFFF",  
  "textPosition": "bottom-third",  
  "encodedImage": " ",  
  "encodedImageType": "SingleColor",  
  "encodedImageName": "Health Icon",  
  "x": "50%",  
  "y": "50%",  
  "translateX": -54,  
  "translateY": -54,  
  "width": 108,  
  "height": 108,  
  "transform": "translate(-54,-54)"  
}
```

[Reset link verifier FilesController#reset\\_verifier](#)

**POST** /api/v1/files/:id/reset\_verifier

**Scope:** url:POST|/api/v1/files/:id/reset\_verifier

Resets the link verifier. Any existing links to the file using the previous hard-coded "verifier" parameter will no longer automatically grant access.

Must have manage files and become other users permissions

**Example Request:**

```
curl -X POST 'https://<canvas>/api/v1/files/<file_id>/reset_verifier' \  
-H 'Authorization: Bearer <token>'
```

Returns a [File](#) object

[List folders FoldersController#api\\_index](#)

**GET** /api/v1/folders/:id/folders

**Scope:** url:GET|/api/v1/folders/:id/folders

Returns the paginated list of folders in the folder.

**Example Request:**

```
curl 'https://<canvas>/api/v1/folders/<folder_id>/folders' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Folder](#) objects

List folders and files [FoldersController#list\\_folders\\_and\\_files](#)

GET /api/v1/folders/:id/all

Scope: url:GET|/api/v1/folders/:id/all

Returns the paginated list of folders in the folder and files.

Example Request:

```
curl 'https://<canvas>/api/v1/folders/<folder_id>/all' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Folder](#) objects

List all folders [FoldersController#list\\_all\\_folders](#)

GET /api/v1/courses/:course\_id/folders

Scope: url:GET|/api/v1/courses/:course\_id/folders

GET /api/v1/users/:user\_id/folders

Scope: url:GET|/api/v1/users/:user\_id/folders

GET /api/v1/groups/:group\_id/folders

Scope: url:GET|/api/v1/groups/:group\_id/folders

Returns the paginated list of all folders for the given context. This will be returned as a flat list containing all subfolders as well.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/folders' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Folder](#) objects

Resolve path [FoldersController#resolve\\_path](#)

GET /api/v1/courses/:course\_id/folders/by\_path/\*full\_path

Scope: url:GET|/api/v1/courses/:course\_id/folders/by\_path/\*full\_path

GET /api/v1/courses/:course\_id/folders/by\_path

Scope: url:GET|/api/v1/courses/:course\_id/folders/by\_path

GET /api/v1/users/:user\_id/folders/by\_path/\*full\_path

Scope: url:GET|/api/v1/users/:user\_id/folders/by\_path/\*full\_path

GET /api/v1/users/:user\_id/folders/by\_path

Scope: url:GET|/api/v1/users/:user\_id/folders/by\_path

GET /api/v1/groups/:group\_id/folders/by\_path/\*full\_path

Scope: url:GET|/api/v1/groups/:group\_id/folders/by\_path/\*full\_path

GET /api/v1/groups/:group\_id/folders/by\_path

Scope: url:GET|/api/v1/groups/:group\_id/folders/by\_path

Given the full path to a folder, returns a list of all Folders in the path hierarchy, starting at the root folder, and ending at the requested folder. The given path is relative to the context's root folder and does not include the root folder's name (e.g., "course files"). If an empty path is given, the context's root folder alone is returned. Otherwise, if no folder exists with the given full path, a Not Found error is returned.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/folders/by_path/foo/bar/baz' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Folder](#) objects

Get folder [FoldersController#show](#)

GET /api/v1/courses/:course\_id/folders/:id

Scope: url:GET|/api/v1/courses/:course\_id/folders/:id

GET /api/v1/users/:user\_id/folders/:id

Scope: url:GET|/api/v1/users/:user\_id/folders/:id

GET /api/v1/groups/:group\_id/folders/:id

Scope: url:GET|/api/v1/groups/:group\_id/folders/:id

GET /api/v1/folders/:id

Scope: url:GET|/api/v1/folders/:id

Returns the details for a folder

You can get the root folder from a context by using 'root' as the :id. For example, you could get the root folder for a course like:

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1337/folders/root' \
-H 'Authorization: Bearer <token>'
```

```
curl 'https://<canvas>/api/v1/folders/<folder_id>' \
-H 'Authorization: Bearer <token>'
```

Returns a [Folder](#) object

Update folder [FoldersController#update](#)

PUT /api/v1/folders/:id

Scope: url:PUT|/api/v1/folders/:id

Updates a folder

Request Parameters:

Parameter	Type	Description
name	string	The new name of the folder
parent_folder_id	string	The id of the folder to move this folder into. The new folder must be in the same context as the original parent folder.
lock_at	DateTime	The datetime to lock the folder at



Parameter	Type	Description
unlock_at	DateTime	The datetime to unlock the folder at
locked	boolean	Flag the folder as locked
hidden	boolean	Flag the folder as hidden
position	integer	Set an explicit sort position for the folder

Example Request:

```
curl -XPUT 'https://<canvas>/api/v1/folders/<folder_id>' \  
-F 'name=<new_name>' \  
-F 'locked=true' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Folder](#) object

Create folder [FoldersController#create](#)

POST /api/v1/courses/:course\_id/folders

Scope: url:POST /api/v1/courses/:course\_id/folders

POST /api/v1/users/:user\_id/folders

Scope: url:POST /api/v1/users/:user\_id/folders

POST /api/v1/groups/:group\_id/folders

Scope: url:POST /api/v1/groups/:group\_id/folders

POST /api/v1/folders/:folder\_id/folders

Scope: url:POST /api/v1/folders/:folder\_id/folders

POST /api/v1/accounts/:account\_id/folders

Scope: url:POST /api/v1/accounts/:account\_id/folders

Creates a folder in the specified context

Request Parameters:

Parameter	Type	Description
name	Required string	The name of the folder
parent_folder_id	string	The id of the folder to store the new folder in. An error will be returned if this does not correspond to an existing folder. If this and parent_folder_path are sent an error will be returned. If neither is given, a default folder will be used.
parent_folder_path	string	The path of the folder to store the new folder in. The path separator is the forward slash '/', never a back slash. The parent folder will be created if it does not already exist. This parameter only applies to new folders in a context that has folders, such as a user, a course, or a group. If this and parent_folder_id are sent an error will be returned. If neither is given, a default folder will be used.
lock_at	DateTime	The datetime to lock the folder at
unlock_at	DateTime	The datetime to unlock the folder at
locked	boolean	Flag the folder as locked
hidden	boolean	Flag the folder as hidden
position	integer	Set an explicit sort position for the folder

Example Request:

```
curl 'https://<canvas>/api/v1/folders/<folder_id>/folders' \  
-F 'name=<new_name>' \  
-F 'locked=true' \  
-H 'Authorization: Bearer <token>'
```

```
curl 'https://<canvas>/api/v1/courses/<course_id>/folders' \  
-F 'name=<new_name>' \  
-F 'locked=true' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Folder](#) object

Delete folder [FoldersController#api\\_destroy](#)

DELETE /api/v1/folders/id

Scope: url:DELETE /api/v1/folders:id

Remove the specified folder. You can only delete empty folders unless you set the ‘force’ flag

Request Parameters:

Parameter	Type	Description
force	boolean	Set to ‘true’ to allow deleting a non-empty folder

Example Request:

```
curl -X DELETE 'https://<canvas>/api/v1/folders/<folder_id>' \  
-H 'Authorization: Bearer <token>'
```

Upload a file [FoldersController#create\\_file](#)

POST /api/v1/folders/:folder\_id/files

Scope: url:POST /api/v1/folders/:folder\_id/files

Upload a file to a folder.

This API endpoint is the first step in uploading a file. See the [File Upload Documentation](#) for details on the file upload workflow.

Only those with the “Manage Files” permission on a course or group can upload files to a folder in that course or group.

Copy a file [FoldersController#copy\\_file](#)

POST /api/v1/folders/:dest\_folder\_id/copy\_file

Scope: url:POST /api/v1/folders/:dest\_folder\_id/copy\_file

Copy a file from elsewhere in Canvas into a folder.

Copying a file across contexts (between courses and users) is permitted, but the source and destination must belong to the same institution.

Request Parameters:

Parameter	Type	Description
source_file_id	Required string	The id of the source file
		What to do if a file with the same name already exists at the destination. If such a file exists and this parameter is not given, the call will fail.
		“overwrite”
on_duplicate	string	Replace an existing file with the same name
		“rename”
		Add a qualifier to make the new filename unique
		Allowed values: overwrite, rename

Example Request:

```
curl 'https://<canvas>/api/v1/folders/123/copy_file' \  
-H 'Authorization: Bearer <token>' \  
-F 'source_file_id=456'
```

Returns a [File](#) object

[Copy a folder FoldersController#copy\\_folder](#)

**POST /api/v1/folders/:dest\_folder\_id/copy\_folder**

**Scope:** url:POST /api/v1/folders/:dest\_folder\_id/copy\_folder

Copy a folder (and its contents) from elsewhere in Canvas into a folder.

Copying a folder across contexts (between courses and users) is permitted, but the source and destination must belong to the same institution. If the source and destination folders are in the same context, the source folder may not contain the destination folder. A folder will be renamed at its destination if another folder with the same name already exists.

Request Parameters:

Parameter	Type	Description
source_folder_id	Required string	The id of the source folder

Example Request:

```
curl 'https://<canvas>/api/v1/folders/123/copy_folder' \  
-H 'Authorization: Bearer <token>' \  
-F 'source_file_id=789'
```

Returns a [Folder](#) object

[Get uploaded media folder for user FoldersController#media\\_folder](#)

**GET /api/v1/courses/:course\_id/folders/media**

**Scope:** url:GET /api/v1/courses/:course\_id/folders/media

**GET /api/v1/groups/:group\_id/folders/media**

**Scope:** url:GET /api/v1/groups/:group\_id/folders/media

Returns the details for a designated upload folder that the user has rights to upload to, and creates it if it doesn't exist.

If the current user does not have the permissions to manage files in the course or group, the folder will belong to the current user directly.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1337/folders/media' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Folder](#) object

[Set usage rights UsageRightsController#set\\_usage\\_rights](#)

**PUT /api/v1/courses/:course\_id/usage\_rights**

**Scope:** url:PUT /api/v1/courses/:course\_id/usage\_rights

**PUT /api/v1/groups/:group\_id/usage\_rights**

**Scope:** url:PUT /api/v1/groups/:group\_id/usage\_rights

**PUT /api/v1/users/:user\_id/usage\_rights**

**Scope:** url:PUT /api/v1/users/:user\_id/usage\_rights

Sets copyright and license information for one or more files

Request Parameters:

Parameter	Type	Description
file_ids[]	Required string	List of ids of files to set usage rights for.
folder_ids[]	string	List of ids of folders to search for files to set usage rights for. Note that new files uploaded to these folders do not automatically inherit these rights.
publish	boolean	Whether the file(s) or folder(s) should be published on save, provided that usage rights have been specified (set to 'true' to publish on save).
usage_rights[use_justification]	Required string	The intellectual property justification for using the files in Canvas Allowed values: own_copyright, used_by_permission, fair_use, public_domain, creative_commons
usage_rights[legal_copyright]	string	The legal copyright line for the files
usage_rights[license]	string	The license that applies to the files. See the <a href="#">List licenses endpoint</a> for the supported license types.

Returns an [UsageRights](#) object

[Remove usage rights UsageRightsController#remove\\_usage\\_rights](#)

**DELETE /api/v1/courses/:course\_id/usage\_rights**

**Scope:** url:DELETE /api/v1/courses/:course\_id/usage\_rights

**DELETE /api/v1/groups/:group\_id/usage\_rights**

**Scope:** url:DELETE /api/v1/groups/:group\_id/usage\_rights

**DELETE /api/v1/users/:user\_id/usage\_rights**

**Scope:** url:DELETE /api/v1/users/:user\_id/usage\_rights

Removes copyright and license information associated with one or more files

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
file_ids[]	Required string	List of ids of files to remove associated usage rights from.
folder_ids[]	string	List of ids of folders. Usage rights will be removed from all files in these folders.

List licenses [UsageRightsController#licenses](#)

GET /api/v1/courses/:course\_id/content\_licenses

Scope: url:GET|/api/v1/courses/:course\_id/content\_licenses

GET /api/v1/groups/:group\_id/content\_licenses

Scope: url:GET|/api/v1/groups/:group\_id/content\_licenses

GET /api/v1/users/:user\_id/content\_licenses

Scope: url:GET|/api/v1/users/:user\_id/content\_licenses

A paginated list of licenses that can be applied

Returns a list of [License](#) objects

Query by assignment [GradeChangeAuditApiController#for\\_assignment](#)

GET /api/v1/audit/grade\_change/assignments/:assignment\_id

Scope: url:GET|/api/v1/audit/grade\_change/assignments/:assignment\_id

List grade change events for a given assignment.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [GradeChangeEvent](#) objects

Query by course [GradeChangeAuditApiController#for\\_course](#)

GET /api/v1/audit/grade\_change/courses/:course\_id

Scope: url:GET|/api/v1/audit/grade\_change/courses/:course\_id

List grade change events for a given course.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [GradeChangeEvent](#) objects

Query by student [GradeChangeAuditApiController#for\\_student](#)

GET /api/v1/audit/grade\_change/students/:student\_id

Scope: url:GET|/api/v1/audit/grade\_change/students/:student\_id

List grade change events for a given student.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [GradeChangeEvent](#) objects

Query by grader [GradeChangeAuditApiController#for\\_grader](#)

GET /api/v1/audit/grade\_change/graders/:grader\_id

Scope: url:GET|/api/v1/audit/grade\_change/graders/:grader\_id

List grade change events for a given grader.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [GradeChangeEvent](#) objects

Advanced query [GradeChangeAuditApiController#query](#)

GET /api/v1/audit/grade\_change

Scope: url:GET|/api/v1/audit/grade\_change

List grade change events satisfying all given parameters. Teachers may query for events in courses they teach. Queries without course\_id require account administrator rights.

At least one of course\_id, assignment\_id, student\_id, or grader\_id must be specified.

Request Parameters:

Parameter	Type	Description
course_id	integer	Restrict query to events in the specified course.
assignment_id	integer	Restrict query to the given assignment. If “override” is given, query the course final grade override instead.
student_id	integer	User id of a student to search grading events for.
grader_id	integer	User id of a grader to search grading events for.

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want events.
end_time	DateTime	The end of the time range from which you want events.

Returns a list of [GradeChangeEvent](#) objects

[Days in gradebook history for this course GradebookHistoryApiController#days](#)

**GET /api/v1/courses/:course\_id/gradebook\_history/days**

**Scope:** url:GET|/api/v1/courses/:course\_id/gradebook\_history/days

Returns a map of dates to grader/assignment groups

**Request Parameters:**

Parameter	Type	Description
course_id	Required integer	The id of the contextual course for this API call

Returns a list of [Day](#) objects

[Details for a given date in gradebook history for this course GradebookHistoryApiController#day\\_details](#)

**GET /api/v1/courses/:course\_id/gradebook\_history/:date**

**Scope:** url:GET|/api/v1/courses/:course\_id/gradebook\_history/:date

Returns the graders who worked on this day, along with the assignments they worked on. More details can be obtained by selecting a grader and assignment and calling the 'submissions' api endpoint for a given date.

**Request Parameters:**

Parameter	Type	Description
course_id	Required integer	The id of the contextual course for this API call
date	Required string	The date for which you would like to see detailed information

Returns a list of [Grader](#) objects

[Lists submissions GradebookHistoryApiController#submissions](#)

**GET /api/v1/courses/:course\_id/gradebook\_history/:date/graders/:grader\_id/assignments/:assignment\_id/submissions**

**Scope:** url:GET|/api/v1/courses/:course\_id/gradebook\_history/:date/graders/:grader\_id/assignments/:assignment\_id/submissions

Gives a nested list of submission versions

**Request Parameters:**

Parameter	Type	Description
course_id	Required integer	The id of the contextual course for this API call
date	Required string	The date for which you would like to see submissions
grader_id	Required integer	The ID of the grader for which you want to see submissions
assignment_id	Required integer	The ID of the assignment for which you want to see submissions

Returns a list of [SubmissionHistory](#) objects

[List uncollated submission versions GradebookHistoryApiController#feed](#)

**GET /api/v1/courses/:course\_id/gradebook\_history/feed**

**Scope:** url:GET|/api/v1/courses/:course\_id/gradebook\_history/feed

Gives a paginated, uncollated list of submission versions for all matching submissions in the context. This SubmissionVersion objects will not include the new\_grade OR previous\_grade keys, only the grade; same for graded\_at and grader.

**Request Parameters:**

Parameter	Type	Description
course_id	Required integer	The id of the contextual course for this API call
assignment_id	integer	The ID of the assignment for which you want to see submissions. If absent, versions of submissions from any assignment in the course are included.
user_id	integer	The ID of the user for which you want to see submissions. If absent, versions of submissions from any user in the course are included.
ascending	boolean	Returns submission versions in ascending date order (oldest first). If absent, returns submission versions in descending date order (newest first).

Returns a list of [SubmissionVersion](#) objects

[List grading period sets GradingPeriodSetsController#index](#)

**GET /api/v1/accounts/:account\_id/grading\_period\_sets**

**Scope:** url:GET|/api/v1/accounts/:account\_id/grading\_period\_sets

Returns the paginated list of grading period sets

**Example Response:**

```
{
  "grading_period_set": [GradingPeriodGroup]
}
```

[Create a grading period set GradingPeriodSetsController#create](#)

**POST /api/v1/accounts/:account\_id/grading\_period\_sets**

**Scope:** url:POST|/api/v1/accounts/:account\_id/grading\_period\_sets

Create and return a new grading period set

**Request Parameters:**

Parameter	Type	Description
enrollment_term_ids[]	Array	A list of associated term ids for the grading period set
grading_period_set[title]	Required string	The title of the grading period set
grading_period_set[weighted]	boolean	A boolean to determine whether the grading periods in the set are weighted

Parameter	Type	Description
grading_period_set[display_totals_for_all_grading_periods]	boolean	A boolean to determine whether the totals for all grading periods in the set are displayed

Example Response:

```
{
  "grading_period_set": [GradingPeriodGroup]
}
```

Update a grading period set [GradingPeriodSetsController#update](#)

PATCH /api/v1/accounts/:account\_id/grading\_period\_sets/:id

Scope: url:PATCH|/api/v1/accounts/:account\_id/grading\_period\_sets/:id

Update an existing grading period set

204 No Content response code is returned if the update was successful.

Request Parameters:

Parameter	Type	Description
enrollment_term_ids[]	Array	A list of associated term ids for the grading period set
grading_period_set[title]	Required string	The title of the grading period set
grading_period_set[weighted]	boolean	A boolean to determine whether the grading periods in the set are weighted
grading_period_set[display_totals_for_all_grading_periods]	boolean	A boolean to determine whether the totals for all grading periods in the set are displayed

Delete a grading period set [GradingPeriodSetsController#destroy](#)

DELETE /api/v1/accounts/:account\_id/grading\_period\_sets/:id

Scope: url:DELETE|/api/v1/accounts/:account\_id/grading\_period\_sets/:id

204 No Content response code is returned if the deletion was successful.

List grading periods [GradingPeriodsController#index](#)

GET /api/v1/accounts/:account\_id/grading\_periods

Scope: url:GET|/api/v1/accounts/:account\_id/grading\_periods

GET /api/v1/courses/:course\_id/grading\_periods

Scope: url:GET|/api/v1/courses/:course\_id/grading\_periods

Returns the paginated list of grading periods for the current course.

Example Response:

```
{
  "grading_periods": [GradingPeriod]
}
```

Get a single grading period [GradingPeriodsController#show](#)

GET /api/v1/courses/:course\_id/grading\_periods/:id

Scope: url:GET|/api/v1/courses/:course\_id/grading\_periods/:id

Returns the grading period with the given id

Example Response:

```
{
  "grading_periods": [GradingPeriod]
}
```

Update a single grading period [GradingPeriodsController#update](#)

PUT /api/v1/courses/:course\_id/grading\_periods/:id

Scope: url:PUT|/api/v1/courses/:course\_id/grading\_periods/:id

Update an existing grading period.

Request Parameters:

Parameter	Type	Description
grading_periods[[start_date]	Required Date	The date the grading period starts.
grading_periods[[end_date]	Required Date	no description
grading_periods[[weight]	number	A weight value that contributes to the overall weight of a grading period set which is used to calculate how much assignments in this period contribute to the total grade

Example Response:

```
{
  "grading_periods": [GradingPeriod]
}
```

Delete a grading period [GradingPeriodsController#destroy](#)

DELETE /api/v1/courses/:course\_id/grading\_periods/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/grading\_periods/:id

DELETE /api/v1/accounts/:account\_id/grading\_periods/:id

Scope: url:DELETE|/api/v1/accounts/:account\_id/grading\_periods/:id

204 No Content response code is returned if the deletion was successful.

Batch update grading periods [GradingPeriodsController#batch\\_update](#)

PATCH /api/v1/courses/:course\_id/grading\_periods/batch\_update

Scope: url:PATCH|/api/v1/courses/:course\_id/grading\_periods/batch\_update

PATCH /api/v1/grading\_period\_sets/:set\_id/grading\_periods/batch\_update

Scope: url:PATCH|/api/v1/grading\_period\_sets/:set\_id/grading\_periods/batch\_update

Update multiple grading periods

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
set_id	Required string	The id of the grading period set.
grading_periods[][id]	string	The id of the grading period. If the id parameter does not exist, a new grading period will be created.
grading_periods[][title]	Required string	The title of the grading period. The title is required for creating a new grading period, but not for updating an existing grading period.
grading_periods[][start_date]	Required Date	The date the grading period starts. The start_date is required for creating a new grading period, but not for updating an existing grading period.
grading_periods[][end_date]	Required Date	The date the grading period ends. The end_date is required for creating a new grading period, but not for updating an existing grading period.
grading_periods[][close_date]	Required Date	The date after which grades can no longer be changed for a grading period. The close_date is required for creating a new grading period, but not for updating an existing grading period.
grading_periods[][weight]	number	A weight value that contributes to the overall weight of a grading period set which is used to calculate how much assignments in this period contribute to the total grade

Example Response:

```
{
  "grading_periods": [GradingPeriod]
}
```

Create a new grading standard [GradingStandardsApiController#create](#)

POST /api/v1/accounts/:account\_id/grading\_standards

Scope: url: POST /api/v1/accounts/:account\_id/grading\_standards

POST /api/v1/courses/:course\_id/grading\_standards

Scope: url: POST /api/v1/courses/:course\_id/grading\_standards

Create a new grading standard

Request Parameters:

Parameter	Type	Description
title	Required string	The title for the Grading Standard.
points_based	boolean	Whether or not a grading scheme is points based. Defaults to false.
scaling_factor	integer	The factor by which to scale a percentage into a points based scheme grade. This is the maximum number of points possible in the grading scheme. Defaults to 1. Not required for percentage based grading schemes.
grading_scheme_entry[[name]	Required string	The name for an entry value within a GradingStandard that describes the range of the value e.g. A-
grading_scheme_entry[[value]	Required integer	The value for the name of the entry within a GradingStandard. The entry represents the lower bound of the range for the entry. This range includes the value up to the next entry in the GradingStandard, or 100 if there is no upper bound. The lowest value will have a lower bound range of 0. e.g. 93

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/grading_standards \
-X POST \
-H 'Authorization: Bearer <token>' \
-d 'title=New standard name' \
-d 'points_based=false' \
-d 'scaling_factor=1.0' \
-d 'grading_scheme_entry[[name]=A' \
-d 'grading_scheme_entry[[value]=94' \
-d 'grading_scheme_entry[[name]=A-' \
-d 'grading_scheme_entry[[value]=99' \
-d 'grading_scheme_entry[[name]=B+' \
-d 'grading_scheme_entry[[value]=87' \
-d 'grading_scheme_entry[[name]=B' \
-d 'grading_scheme_entry[[value]=84' \
-d 'grading_scheme_entry[[name]=B-' \
-d 'grading_scheme_entry[[value]=80' \
-d 'grading_scheme_entry[[name]=C+' \
-d 'grading_scheme_entry[[value]=77' \
-d 'grading_scheme_entry[[name]=C' \
-d 'grading_scheme_entry[[value]=74' \
-d 'grading_scheme_entry[[name]=C-' \
-d 'grading_scheme_entry[[value]=70' \
-d 'grading_scheme_entry[[name]=D+' \
-d 'grading_scheme_entry[[value]=67' \
-d 'grading_scheme_entry[[name]=D' \
-d 'grading_scheme_entry[[value]=64' \
-d 'grading_scheme_entry[[name]=D-' \
-d 'grading_scheme_entry[[value]=60' \
-d 'grading_scheme_entry[[name]=F' \
-d 'grading_scheme_entry[[value]=0'
```

Example Response:

```
{
  "title": "New standard name",
  "id": 1,
  "context_id": 1,
  "context_type": "Course",
  "grading_scheme": [
    { "name": "A", "value": 0.9 },
    { "name": "B", "value": 0.8 }
  ]
}
```

Returns a [GradingStandard](#) object

List the grading standards available in a context. [GradingStandardsApiController#context\\_index](#)

GET /api/v1/courses/:course\_id/grading\_standards

Scope: url: GET /api/v1/courses/:course\_id/grading\_standards

GET /api/v1/accounts/:account\_id/grading\_standards

Scope: url: GET /api/v1/accounts/:account\_id/grading\_standards

Returns the paginated list of grading standards for the given context that are visible to the user.

Example Request:

```
curl https://<canvas>/api/v1/courses/1/grading_standards \
-H 'Authorization: Bearer <token>'
```

Returns a list of [GradingStandard](#) objects

Get a single grading standard in a context. [GradingStandardsApiController#context\\_show](#)

GET /api/v1/courses/:course\_id/grading\_standards/:grading\_standard\_id

Scope: url: GET /api/v1/courses/:course\_id/grading\_standards/:grading\_standard\_id

GET /api/v1/accounts/:account\_id/grading\_standards/:grading\_standard\_id

Scope: url: GET /api/v1/accounts/:account\_id/grading\_standards/:grading\_standard\_id

Returns a grading standard for the given context that is visible to the user.

Example Request:

```
curl https://<canvas>/api/v1/courses/1/grading_standards/5 \
-H 'Authorization: Bearer <token>'
```

Returns a [GradingStandard](#) object

[List group categories for a context GroupCategoriesController#index](#)

**GET /api/v1/accounts/:account\_id/group\_categories**

**Scope:** url:GET//api/v1/accounts/:account\_id/group\_categories

**GET /api/v1/courses/:course\_id/group\_categories**

**Scope:** url:GET//api/v1/courses/:course\_id/group\_categories

Returns a paginated list of group categories in a context. The list returned depends on the permissions of the current user. If the user has group management permissions ('GRANULAR\_MANAGE\_GROUPS\_PERMISSIONS'), the response will include only collaborative group categories. If the user has tag management permissions ('GRANULAR\_MANAGE\_TAGS\_PERMISSIONS'), the response will include only non-collaborative group categories.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/group_categories \
-H 'Authorization: Bearer <token>'
```

Returns a list of [GroupCategory](#) objects

[Get a single group category GroupCategoriesController#show](#)

**GET /api/v1/group\_categories/:group\_category\_id**

**Scope:** url:GET//api/v1/group\_categories/:group\_category\_id

Returns the data for a single group category, or a 401 if the caller doesn't have the rights to see it.

**Example Request:**

```
curl https://<canvas>/api/v1/group_categories/<group_category_id> \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupCategory](#) object

[Create a Group Category GroupCategoriesController#create](#)

**POST /api/v1/accounts/:account\_id/group\_categories**

**Scope:** url:POST//api/v1/accounts/:account\_id/group\_categories

**POST /api/v1/courses/:course\_id/group\_categories**

**Scope:** url:POST//api/v1/courses/:course\_id/group\_categories

Create a new group category

**Request Parameters:**

Parameter	Type	Description
name	Required string	Name of the group category
non_collaborative	boolean	Can only be set by users with the Differentiated Tag Add permission If set to true, groups in this category will be only be visible to users with the Differentiated Tag Manage permission.
self_signup	string	Allow students to sign up for a group themselves (Course Only). valid values are: "enabled" allows students to self sign up for any group in course "restricted" allows students to self sign up only for groups in the same section null disallows self sign up Allowed values: enabled, restricted
auto_leader	string	Assigns group leaders automatically when generating and allocating students to groups Valid values are: "first" the first student to be allocated to a group is the leader "random" a random student from all members is chosen as the leader Allowed values: first, random
group_limit	integer	Limit the maximum number of users in each group (Course Only). Requires self signup.
sis_group_category_id	string	The unique SIS identifier.
create_group_count	integer	Create this number of groups (Course Only).
split_group_count	string	(Deprecated) Create this number of groups, and evenly distribute students among them. not allowed with "enable_self_signup". because the group assignment happens synchronously, it's recommended that you instead use the assign_unassigned_members endpoint. (Course Only)

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/group_categories \
-F 'name=Project Groups' \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupCategory](#) object

[Import category groups GroupCategoriesController#import](#)

**POST /api/v1/group\_categories/:group\_category\_id/import**

**Scope:** url:POST//api/v1/group\_categories/:group\_category\_id/import

Create Groups in a Group Category through a CSV import

For more information on the format that's expected here, please see the "Group Category CSV" section in the API docs.

**Request Parameters:**

Parameter	Type	Description
attachment	string	There are two ways to post group category import data - either via a multipart/form-data form-field-style attachment, or via a non-multipart raw post request.  "attachment" is required for multipart/form-data style posts. Assumed to be outcome data from a file upload form field named 'attachment'.  Examples:  curl -F attachment=@<filename> -H "Authorization: Bearer <token>" \ 'https://<canvas>/api/v1/group_categories/<category_id>/import'  If you decide to do a raw post, you can skip the 'attachment' argument, but you will then be required to provide a suitable Content-Type header. You are encouraged to also provide the 'extension' argument.  Examples:  curl -H 'Content-Type: text/csv' --data-binary @<filename>.csv \ -H 'Authorization: Bearer <token>' \

Parameter	Type	Description
-----------	------	-------------

'https://<canvas>/api/v1/group\_categories/<category\_id>/import'

Example Response:

```
# Progress (default)
{
  "completion": 0,
  "context_id": 20,
  "context_type": "GroupCategory",
  "created_at": "2013-07-05T10:57:48-06:00",
  "id": 2,
  "message": null,
  "tag": "course_group_import",
  "updated_at": "2013-07-05T10:57:48-06:00",
  "user_id": null,
  "workflow_state": "running",
  "url": "http://localhost:3000/api/v1/progress/2"
}
```

Returns a [Progress](#) object

[Update a Group Category GroupCategoriesController#update](#)

**PUT /api/v1/group\_categories/:group\_category\_id**

**Scope:** url:PUT /api/v1/group\_categories/:group\_category\_id

Modifies an existing group category.

Request Parameters:

Parameter	Type	Description
name	string	Name of the group category
		Allow students to sign up for a group themselves (Course Only). Valid values are: "enabled"
self_signup	string	allows students to self sign up for any group in course "restricted"
		allows students to self sign up only for groups in the same section null disallows self sign up Allowed values: enabled, restricted
		Assigns group leaders automatically when generating and allocating students to groups Valid values are: "first"
auto_leader	string	the first student to be allocated to a group is the leader "random"
		a random student from all members is chosen as the leader Allowed values: first, random
group_limit	integer	Limit the maximum number of users in each group (Course Only). Requires self signup.
sis_group_category_id	string	The unique SIS identifier.
create_group_count	integer	Create this number of groups (Course Only).
split_group_count	string	(Deprecated) Create this number of groups, and evenly distribute students among them. not allowed with "enable_self_signup". because the group assignment happens synchronously, it's recommended that you instead use the assign_unassigned_members endpoint. (Course Only)

Example Request:

```
curl https://<canvas>/api/v1/group_categories/<group_category_id> \
-X PUT \
-F 'name=Project Groups' \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupCategory](#) object

[Delete a Group Category GroupCategoriesController#destroy](#)

**DELETE /api/v1/group\_categories/:group\_category\_id**

**Scope:** url:DELETE /api/v1/group\_categories/:group\_category\_id

Deletes a group category and all groups under it. Protected group categories can not be deleted, i.e. "communities" and "student\_organized".

Example Request:

```
curl https://<canvas>/api/v1/group_categories/<group_category_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

[List groups in group category GroupCategoriesController#groups](#)

**GET /api/v1/group\_categories/:group\_category\_id/groups**

**Scope:** url:GET /api/v1/group\_categories/:group\_category\_id/groups

Returns a paginated list of groups in a group category

Example Request:

```
curl https://<canvas>/api/v1/group_categories/<group_category_id>/groups \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Group](#) objects

[export groups in and users in category GroupCategoriesController#export](#)

BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.

**GET /api/v1/group\_categories/:group\_category\_id/export**

**Scope:** url:GET /api/v1/group\_categories/:group\_category\_id/export

Returns a csv file of users in format ready to import.

Example Request:

```
curl https://<canvas>/api/v1/group_categories/<group_category_id>/export \
-H 'Authorization: Bearer <token>'
```

[List users in group category GroupCategoriesController#users](#)

**GET /api/v1/group\_categories/:group\_category\_id/users**

**Scope:** url:GET /api/v1/group\_categories/:group\_category\_id/users

Returns a paginated list of users in the group category.

Request Parameters:



Parameter	Type	Description
search_term	string	The partial name or full ID of the users to match and return in the results list. Must be at least 3 characters.
unassigned	boolean	Set this value to true if you wish only to search unassigned users in the group category.

Example Request:

```
curl https://<canvas>/api/v1/group_categories/1/users \
-H 'Authorization: Bearer <token>'
```

Returns a list of [User](#) objects

[Assign unassigned members GroupCategoriesController#assign\\_unassigned\\_members](#)

POST /api/v1/group\_categories/:group\_category\_id/assign\_unassigned\_members

Scope: url:POST /api/v1/group\_categories/:group\_category\_id/assign\_unassigned\_members

Assign all unassigned members as evenly as possible among the existing student groups.

Request Parameters:

Parameter	Type	Description
sync	boolean	The assigning is done asynchronously by default. If you would like to override this and have the assigning done synchronously, set this value to true.

Example Request:

```
curl https://<canvas>/api/v1/group_categories/1/assign_unassigned_members \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
# Progress (default)
{
  "completion": 0,
  "context_id": 20,
  "context_type": "GroupCategory",
  "created_at": "2013-07-05T10:57:48-06:00",
  "id": 2,
  "message": null,
  "tag": "assign_unassigned_members",
  "updated_at": "2013-07-05T10:57:48-06:00",
  "user_id": null,
  "workflow_state": "running",
  "url": "http://localhost:3000/api/v1/progress/2"
}

# New Group Memberships (when sync = true)
[
  {
    "id": 65,
    "new_members": [
      {
        "user_id": 2,
        "name": "Sam",
        "display_name": "Sam",
        "sections": [
          {
            "section_id": 1,
            "section_code": "Section 1"
          }
        ]
      },
      {
        "user_id": 3,
        "name": "Sue",
        "display_name": "Sue",
        "sections": [
          {
            "section_id": 2,
            "section_code": "Section 2"
          }
        ]
      }
    ]
  },
  {
    "id": 66,
    "new_members": [
      {
        "user_id": 5,
        "name": "Joe",
        "display_name": "Joe",
        "sections": [
          {
            "section_id": 2,
            "section_code": "Section 2"
          }
        ]
      },
      {
        "user_id": 11,
        "name": "Cecil",
        "display_name": "Cecil",
        "sections": [
          {
            "section_id": 3,
            "section_code": "Section 3"
          }
        ]
      }
    ]
  }
]
]
```

[List your groups GroupsController#index](#)

GET /api/v1/users/self/groups

Scope: url:GET /api/v1/users/self/groups

Returns a paginated list of active groups for the current user.

Request Parameters:

Parameter	Type	Description
context_type	string	Only include groups that are in this type of context. Allowed values: Account, Course
include[]	string	<ul style="list-style-type: none"><li>“tabs”: Include the list of tabs configured for each group. See the <a href="#">List available tabs API</a> for more information.</li></ul> Allowed values: tabs

Example Request:

```
curl https://<canvas>/api/v1/users/self/groups?context_type=Account \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Group](#) objects

[List the groups available in a context. GroupsController#context\\_index](#)

GET /api/v1/accounts/:account\_id/groups

Scope: url:GET /api/v1/accounts/:account\_id/groups

GET /api/v1/courses/:course\_id/groups

Scope: url:GET /api/v1/courses/:course\_id/groups

Returns the paginated list of active groups in the given context that are visible to user.

Request Parameters:

Parameter	Type	Description
only_own_groups	boolean	Will only include groups that the user belongs to if this is set
include[]	string	<ul style="list-style-type: none"><li>“tabs”: Include the list of tabs configured for each group. See the <a href="#">List available tabs API</a> for more information.</li></ul> Allowed values: tabs

Example Request:

```
curl https://<canvas>/api/v1/courses/1/groups \
-H 'Authorization: Bearer <token>'
```

Returns a list of [Group](#) objects

[Get a single group GroupsController#show](#)

GET /api/v1/groups/:group\_id

Scope: url:GET//api/v1/groups/:group\_id

Returns the data for a single group, or a 401 if the caller doesn't have the rights to see it.

Request Parameters:

Parameter	Type	Description
		<ul style="list-style-type: none"><li>“permissions”: Include permissions the current user has for the group.</li></ul>
include[]	string	<ul style="list-style-type: none"><li>“tabs”: Include the list of tabs configured for each group. See the <a href="#">List available tabs API</a> for more information.</li></ul> Allowed values: permissions, tabs

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id> \
-H 'Authorization: Bearer <token>'
```

Returns a [Group](#) object

[Create a group GroupsController#create](#)

POST /api/v1/groups

Scope: url:POST//api/v1/groups

POST /api/v1/group\_categories/:group\_category\_id/groups

Scope: url:POST//api/v1/group\_categories/:group\_category\_id/groups

Creates a new group. Groups created using the “/api/v1/groups/” endpoint will be community groups.

Request Parameters:

Parameter	Type	Description
name	string	The name of the group
description	string	A description of the group
is_public	boolean	whether the group is public (applies only to community groups)
join_level	string	<p>no description</p> Allowed values: parent_context_auto_join, parent_context_request, invitation_only
storage_quota_mb	integer	The allowed file storage for the group, in megabytes. This parameter is ignored if the caller does not have the manage_storage_quotas permission.
sis_group_id	string	The sis ID of the group. Must have manage_sis permission to set.

Example Request:

```
curl https://<canvas>/api/v1/groups \
-F 'name=Math Teachers' \
-F 'description=A place to gather resources for our classes.' \
-F 'is_public=true' \
-F 'join_level=parent_context_auto_join' \
-H 'Authorization: Bearer <token>'
```

Returns a [Group](#) object

[Edit a group GroupsController#update](#)

PUT /api/v1/groups/:group\_id

Scope: url:PUT//api/v1/groups/:group\_id

Modifies an existing group. Note that to set an avatar image for the group, you must first upload the image file to the group, and the use the id in the response as the argument to this function. See the [File Upload Documentation](#) for details on the file upload workflow.

Request Parameters:

Parameter	Type	Description
name	string	The name of the group
description	string	A description of the group
is_public	boolean	Whether the group is public (applies only to community groups). Currently you cannot set a group back to private once it has been made public.
join_level	string	<p>no description</p> Allowed values: parent_context_auto_join, parent_context_request, invitation_only
avatar_id	integer	The id of the attachment previously uploaded to the group that you would like to use as the avatar image for this group.
storage_quota_mb	integer	The allowed file storage for the group, in megabytes. This parameter is ignored if the caller does not have the manage_storage_quotas permission.
members[]	string	An array of user ids for users you would like in the group. Users not in the group will be sent invitations. Existing group members who aren't in the list will be removed from the group.
sis_group_id	string	The sis ID of the group. Must have manage_sis permission to set.
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id> \
-X PUT \
-F 'name=Algebra Teachers' \
-F 'join_level=parent_context_request' \
```

-H 'Authorization: Bearer <token>'

Returns a [Group](#) object

Delete a group [GroupsController#destroy](#)

DELETE /api/v1/groups/:group\_id

Scope: url::DELETE|/api/v1/groups/:group\_id

Deletes a group and removes all members.

Example Request:

curl https://<canvas>/api/v1/groups/<group\_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'

Returns a [Group](#) object

Invite others to a group [GroupsController#invite](#)

POST /api/v1/groups/:group\_id/invite

Scope: url::POST|/api/v1/groups/:group\_id/invite

Sends an invitation to all supplied email addresses which will allow the receivers to join the group.

Request Parameters:

Parameter	Type	Description
invitees[]	Required string	An array of email addresses to be sent invitations.

Example Request:

curl https://<canvas>/api/v1/groups/<group\_id>/invite \
-F 'invitees[]=leonard@example.com' \
-F 'invitees[]=sheldon@example.com' \
-H 'Authorization: Bearer <token>'

List group's users [GroupsController#users](#)

List group's users [GroupsController#users](#)

GET /api/v1/groups/:group\_id/users

Scope: url::GET|/api/v1/groups/:group\_id/users

Returns a paginated list of users in the group.

Request Parameters:

Parameter	Type	Description
search_term	string	The partial name or full ID of the users to match and return in the results list. Must be at least 3 characters.
include[]	string	“avatar_url”: Include users' avatar_urls. Allowed values: avatar_url

exclude\_inactive boolean Whether to filter out inactive users from the results. Defaults to false unless explicitly provided.

Example Request:

curl https://<canvas>/api/v1/groups/1/users \
-H 'Authorization: Bearer <token>'

Returns a list of [User](#) objects

Upload a file [GroupsController#create\\_file](#)

POST /api/v1/groups/:group\_id/files

Scope: url::POST|/api/v1/groups/:group\_id/files

Upload a file to the group.

This API endpoint is the first step in uploading a file to a group. See the [File Upload Documentation](#) for details on the file upload workflow.

Only those with the “Manage Files” permission on a group can upload files to the group. By default, this is anybody participating in the group, or any admin over the group.

Preview processed html [GroupsController#preview\\_html](#)

POST /api/v1/groups/:group\_id/preview\_html

Scope: url::POST|/api/v1/groups/:group\_id/preview\_html

Preview html content processed for this group

Request Parameters:

Parameter	Type	Description
html	string	The html content to process

Example Request:

curl https://<canvas>/api/v1/groups/<group\_id>/preview\_html \
-F 'html=<p>badhtml</badhtml>processed html</p>' \
-H 'Authorization: Bearer <token>'

Example Response:

{
 "html": "<p>processed html</p>"
}

Group activity stream [GroupsController#activity\\_stream](#)

GET /api/v1/groups/:group\_id/activity\_stream

Scope: url::GET|/api/v1/groups/:group\_id/activity\_stream

Returns the current user's group-specific activity stream, paginated.

For full documentation, see the API documentation for the user activity stream, in the user api.

Group activity stream summary [GroupsController#activity\\_stream\\_summary](#)

GET /api/v1/groups/:group\_id/activity\_stream/summary

Scope: url::GET|/api/v1/groups/:group\_id/activity\_stream/summary

Returns a summary of the current user's group-specific activity stream.

For full documentation, see the API documentation for the user activity stream summary, in the user api.

Permissions [GroupsController#permissions](#)

GET /api/v1/groups/:group\_id/permissions

Scope: url:GET[/api/v1/groups/:group\_id/permissions]

Returns permission information for the calling user in the given group. See also the [Account](#) and [Course](#) counterparts.

Request Parameters:

Parameter	Type	Description
permissions[]	string	List of permissions to check against the authenticated user. Permission names are documented in the <a href="#">Create a role</a> endpoint.

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/permissions \
-H 'Authorization: Bearer <token>' \
-d 'permissions[]=read_roster' \
-d 'permissions[]=send_messages_all'
```

Example Response:

```
{'read_roster': 'true', 'send_messages_all': 'false'}
```

List group memberships [GroupMembershipsController#index](#)

GET /api/v1/groups/:group\_id/memberships

Scope: url:GET[/api/v1/groups/:group\_id/memberships]

A paginated list of the members of a group.

Request Parameters:

Parameter	Type	Description
filter_states[]	string	Only list memberships with the given workflow_states. By default it will return all memberships. Allowed values: accepted, invited, requested

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/memberships \
-F 'filter_states[]=invited&filter_states[]=requested' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [GroupMembership](#) objects

Get a single group membership [GroupMembershipsController#show](#)

GET /api/v1/groups/:group\_id/memberships/:membership\_id

Scope: url:GET[/api/v1/groups/:group\_id/memberships/:membership\_id]

GET /api/v1/groups/:group\_id/users/:user\_id

Scope: url:GET[/api/v1/groups/:group\_id/users/:user\_id]

Returns the group membership with the given membership id or user id.

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/memberships/<membership_id> \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/groups/<group_id>/users/<user_id> \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupMembership](#) object

Create a membership [GroupMembershipsController#create](#)

POST /api/v1/groups/:group\_id/memberships

Scope: url:POST[/api/v1/groups/:group\_id/memberships]

Join, or request to join, a group, depending on the join\_level of the group. If the membership or join request already exists, then it is simply returned

Request Parameters:

Parameter	Type	Description
user_id	string	no description

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/memberships \
-F 'user_idself' \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupMembership](#) object

Update a membership [GroupMembershipsController#update](#)

PUT /api/v1/groups/:group\_id/memberships/:membership\_id

Scope: url:PUT[/api/v1/groups/:group\_id/memberships/:membership\_id]

PUT /api/v1/groups/:group\_id/users/:user\_id

Scope: url:PUT[/api/v1/groups/:group\_id/users/:user\_id]

Accept a membership request, or add/remove moderator rights.

Request Parameters:

Parameter	Type	Description
workflow_state	string	Currently, the only allowed value is “accepted” Allowed values: accepted
moderator	string	no description

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/memberships/<membership_id> \
-F 'moderator=true' \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/groups/<group_id>/users/<user_id> \
-F 'moderator=true' \
-H 'Authorization: Bearer <token>'
```

Returns a [GroupMembership](#) object

Leave a group [GroupMembershipsController#destroy](#)

DELETE /api/v1/groups/:group\_id/memberships/:membership\_id

Scope: url:DELETE[/api/v1/groups/:group\_id/memberships/:membership\_id]

DELETE /api/v1/groups/:group\_id/users/:user\_id

**Scope:** url:DELETE|/api/v1/groups/:group\_id/users/:user\_id

Leave a group if you are allowed to leave (some groups, such as sets of course groups created by teachers, cannot be left). You may also use ‘self’ in place of a membership\_id.

Example Request:

```
curl https://<canvas>/api/v1/groups/<group_id>/memberships/<membership_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/groups/<group_id>/users/<user_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

List recent history for a user HistoryController#index

GET /api/v1/users/:user\_id/history

**Scope:** url:GET|/api/v1/users/:user\_id/history

Return a paginated list of the user's recent history. History entries are returned in descending order, newest to oldest. You may list history entries for yourself (use self as the user\_id), for a student you observe, or for a user you manage as an administrator. Note that the per\_page pagination argument is not supported and the number of history entries returned per page will vary.

Returns a list of HistoryEntry objects

Create InstAccess token InstAccessTokensController#create

POST /api/v1/inst\_access\_tokens

**Scope:** url:POST|/api/v1/inst\_access\_tokens

Create a unique, encrypted InstAccess token.

Generates a different InstAccess token each time it's called, each one expires after a short window (1 hour).

Example Request:

```
curl 'https://<canvas>/api/v1/inst_access_tokens' \
-X POST \
-H 'Accept: application/json' \
-H 'Authorization: Bearer <token>'
```

Returns an InstAccessToken object

Create JWT JwtController#create

POST /api/v1/jwts

**Scope:** url:POST|/api/v1/jwts

Create a unique jwt for using with other Canvas services

Generates a different JWT each time it's called, each one expires after a short window (1 hour)

Request Parameters:

Parameter	Type	Description
workflows[]	string	Adds additional data to the JWT to be used by the consuming service workflow
context_type	string	The type of the context in case a specified workflow uses it to consuming the service. Case insensitive. Allowed values: Course, User, Account
context_id	integer	The id of the context in case a specified workflow uses it to consuming the service.
context_uuid	string	The uuid of the context in case a specified workflow uses it to consuming the service.
canvas_audience	boolean	Defaults to true. If false, the JWT will be signed, but not encrypted, for use in downstream services. The default encrypted behaviour can be used to talk to canvas itself.

Example Request:

```
curl 'https://<canvas>/api/v1/jwts' \
-X POST \
-H 'Accept: application/json' \
-H 'Authorization: Bearer <token>'
```

Returns a JWT object

Refresh JWT JwtController#refresh

POST /api/v1/jwts/refresh

**Scope:** url:POST|/api/v1/jwts/refresh

Refresh a JWT for use with other canvas services

Generates a different JWT each time it's called, each one expires after a short window (1 hour).

Request Parameters:

Parameter	Type	Description
jwt	string	Required string An existing JWT token to be refreshed. The new token will have the same context and workflows as the existing token.

Example Request:

```
curl 'https://<canvas>/api/v1/jwts/refresh' \
-X POST \
-H 'Accept: application/json' \
-H 'Authorization: Bearer <token>' \
-d 'jwt=<jwt>'
```

Returns a JWT object

Get a late policy LatePolicyController#show

GET /api/v1/courses/:id/late\_policy

**Scope:** url:GET|/api/v1/courses/:id/late\_policy

Returns the late policy for a course.

Example Response:

```
{
  "late_policy": LatePolicy
}
```

Create a late policy LatePolicyController#create

POST /api/v1/courses/:id/late\_policy

**Scope:** url:POST|/api/v1/courses/:id/late\_policy

Create a late policy. If the course already has a late policy, a bad\_request is returned since there can only be one late policy per course.

Request Parameters:

Parameter	Type	Description
late_policy[missing_submission_deduction_enabled]	boolean	Whether to enable the missing submission deduction late policy.
late_policy[missing_submission_deduction]	number	How many percentage points to deduct from a missing submission.
late_policy[late_submission_deduction_enabled]	boolean	Whether to enable the late submission deduction late policy.
late_policy[late_submission_deduction]	number	How many percentage points to deduct per the late submission interval.
late_policy[late_submission_interval]	string	The interval for late policies.
late_policy[late_submission_minimum_percent_enabled]	boolean	Whether to enable the late submission minimum percent for a late policy.
late_policy[late_submission_minimum_percent]	number	The minimum grade a submissions can have in percentage points.

Example Response:

```
{
  "late_policy": LatePolicy
}
```

Patch a late policy [LatePolicyController#update](#)

PATCH /api/v1/courses/:id/late\_policy

Scope: url:PATCH|/api/v1/courses/:id/late\_policy

Patch a late policy. No body is returned upon success.

Request Parameters:

Parameter	Type	Description
late_policy[missing_submission_deduction_enabled]	boolean	Whether to enable the missing submission deduction late policy.
late_policy[missing_submission_deduction]	number	How many percentage points to deduct from a missing submission.
late_policy[late_submission_deduction_enabled]	boolean	Whether to enable the late submission deduction late policy.
late_policy[late_submission_deduction]	number	How many percentage points to deduct per the late submission interval.
late_policy[late_submission_interval]	string	The interval for late policies.
late_policy[late_submission_minimum_percent_enabled]	boolean	Whether to enable the late submission minimum percent for a late policy.
late_policy[late_submission_minimum_percent]	number	The minimum grade a submissions can have in percentage points.

Get a learning object's date information [LearningObjectDatesController#show](#)

GET /api/v1/courses/:course\_id/modules/:context\_module\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/modules/:context\_module\_id/date\_details

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/date\_details

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/date\_details

GET /api/v1/courses/:course\_id/discussion\_topics/:discussion\_topic\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/discussion\_topics/:discussion\_topic\_id/date\_details

GET /api/v1/courses/:course\_id/pages/:url\_or\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/pages/:url\_or\_id/date\_details

GET /api/v1/courses/:course\_id/files/:attachment\_id/date\_details

Scope: url:GET|/api/v1/courses/:course\_id/files/:attachment\_id/date\_details

Get a learning object's date-related information, including due date, availability dates, override status, and a paginated list of all assignment overrides for the item.

Returns a [LearningObjectDates](#) object

Update a learning object's date information [LearningObjectDatesController#update](#)

PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/date\_details

Scope: url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/date\_details

PUT /api/v1/courses/:course\_id/quizzes/:quiz\_id/date\_details

Scope: url:PUT|/api/v1/courses/:course\_id/quizzes/:quiz\_id/date\_details

PUT /api/v1/courses/:course\_id/discussion\_topics/:discussion\_topic\_id/date\_details

Scope: url:PUT|/api/v1/courses/:course\_id/discussion\_topics/:discussion\_topic\_id/date\_details

PUT /api/v1/courses/:course\_id/pages/:url\_or\_id/date\_details

Scope: url:PUT|/api/v1/courses/:course\_id/pages/:url\_or\_id/date\_details

PUT /api/v1/courses/:course\_id/files/:attachment\_id/date\_details

Scope: url:PUT|/api/v1/courses/:course\_id/files/:attachment\_id/date\_details

Updates date-related information for learning objects, including due date, availability dates, override status, and assignment overrides.

Returns 204 No Content response code if successful.

Request Parameters:

Parameter	Type	Description
due_at	DateTime	The learning object's due date. Not applicable for ungraded discussions, pages, and files.
unlock_at	DateTime	The learning object's unlock date. Must be before the due date if there is one.
lock_at	DateTime	The learning object's lock date. Must be after the due date if there is one.
only_visible_to_overrides	boolean	Whether the learning object is only assigned to students who are targeted by an override.

Parameter	Type	Description
assignment_overrides[]	Array	List of overrides to apply to the learning object. Overrides that already exist should include an ID and will be updated if needed. New overrides will be created for overrides in the list without an ID. Overrides not included in the list will be deleted. Providing an empty list will delete all of the object's overrides. Keys for each override object can include: 'id', 'title', 'due_at', 'unlock_at', 'lock_at', 'student_ids', and 'course_section_id', 'course_id', 'noop_id', and 'unassign_item'.

Example Request:

```
curl https://<canvas>/api/v1/courses/:course_id/assignments/:assignment_id/date_details \
-X PUT \
-H 'Authorization: Bearer <token>' \
-H 'Content-Type: application/json' \
-d '{
  "due_at": "2012-07-01T23:59:00-06:00",
  "unlock_at": "2012-06-01T00:00:00-06:00",
  "lock_at": "2012-08-01T00:00:00-06:00",
  "only_visible_to_overrides": true,
  "assignment_overrides": [
    {
      "id": 212,
      "course_section_id": 3564
    },
    {
      "title": "an assignment override",
      "student_ids": [1, 2, 3]
    }
  ]
},'
```

Create a Line Item Lti::Ims::LineItemsController#create

POST /api/lti/courses/:course\_id/line\_items

Scope: url:POST|/api/lti/courses/:course\_id/line\_items

Create a new Line Item

Request Parameters:

Parameter	Type	Description
scoreMaximum	number	Required number The maximum score for the line item. Scores created for the Line Item may exceed this value.
label	string	Required string The label for the Line Item. If no resourceLinkId is specified this value will also be used as the name of the placeholder assignment.
resourceId	string	A Tool Provider specified id for the Line Item. Multiple line items may share the same resourceId within a given context.
tag	string	A value used to qualify a line Item beyond its ids. Line Items may be queried by this value in the List endpoint. Multiple line items can share the same tag within a given context.
resourceLinkId	string	The resource link id the Line Item should be attached to. This value should match the LTI id of the Canvas assignment associated with the tool.
startDateTime	string	The ISO8601 date and time when the line item is made available. Corresponds to the assignment's unlock_at date.
endDateTime	string	The ISO8601 date and time when the line item stops receiving submissions. Corresponds to the assignment's due_at date.
https://canvas.instructure.com/lti/submission_type	object	(EXTENSION) - Optional block to set Assignment Submission Type when creating a new assignment is created. type - 'none' or 'external_tool' external_tool_url - Submission URL only used when type: 'external_tool'

Example Request:

```
{
  "scoreMaximum": 100.0,
  "label": "LineItemLabel1",
  "resourceId": 1,
  "tag": "MyTag",
  "resourceLinkId": "1",
  "startDateTime": "2022-01-31T22:23:11+0000",
  "endDateTime": "2022-02-07T22:23:11+0000",
  "https://canvas.instructure.com/lti/submission_type": {
    "type": "external_tool",
    "external_tool_url": "https://my.launch.url"
  }
}
```

Returns a [LineItem](#) object

Update a Line Item Lti::Ims::LineItemsController#update

PUT /api/lti/courses/:course\_id/line\_items/:id

Scope: url:PUT|/api/lti/courses/:course\_id/line\_items/:id

Update new Line Item

Request Parameters:

Parameter	Type	Description
scoreMaximum	number	The maximum score for the line item. Scores created for the Line Item may exceed this value.
label	string	The label for the Line Item. If no resourceLinkId is specified this value will also be used as the name of the placeholder assignment.
resourceId	string	A Tool Provider specified id for the Line Item. Multiple line items may share the same resourceId within a given context.
tag	string	A value used to qualify a line Item beyond its ids. Line Items may be queried by this value in the List endpoint. Multiple line items can share the same tag within a given context.
startDateTime	string	The ISO8601 date and time when the line item is made available. Corresponds to the assignment's unlock_at date.
endDateTime	string	The ISO8601 date and time when the line item stops receiving submissions. Corresponds to the assignment's due_at date.

Returns a [LineItem](#) object

Show a Line Item Lti::Ims::LineItemsController#show

GET /api/lti/courses/:course\_id/line\_items/:id

Scope: url:GET|/api/lti/courses/:course\_id/line\_items/:id

Show existing Line Item

Request Parameters:

Parameter	Type	Description
	Array	Array of additional information to include.
include[]	string	"launch_url" includes the launch URL for this line item using the "https://canvas.instructure.com/lti/launch_url" extension Allowed values: launch_url

Returns a [LineItem](#) object

List line Items Lti::Ims::LineItemsController#index

GET /api/lti/courses/:course\_id/line\_items

Scope: url:GET|/api/lti/courses/:course\_id/line\_items

List all Line Items for a course

Request Parameters:

Parameter	Type	Description
tag	string	If specified only Line Items with this tag will be included.
resource_id	string	If specified only Line Items with this resource_id will be included.
resource_link_id	string	If specified only Line Items attached to the specified resource_link_id will be included.
limit	string	May be used to limit the number of Line Items returned in a page
	Array	Array of additional information to include.
include[]	string	"launch_url" includes the launch URL for each line item using the "https://canvas.instructure.com/lti/launch_url" extension Allowed values: launch_url

Returns a [LineItem](#) object

Delete a Line Item Lti::Ims::LineItemsController#destroy

DELETE /api/lti/courses/:course\_id/line\_items/:id

Scope: url:DELETE|/api/lti/courses/:course\_id/line\_items/:id

Delete an existing Line Item

Returns a [LineItem](#) object

Create live assessment results LiveAssessments::ResultsController#create

POST /api/v1/courses/:course\_id/live\_assessments/:assessment\_id/results

Scope: url:POST|/api/v1/courses/:course\_id/live\_assessments/:assessment\_id/results

Creates live assessment results and adds them to a live assessment

Example Request:

```
{
  "results": [
    {
      "passed": false,
      "assessed_at": "2014-05-26T14:57:23-07:00",
      "links": {
        "user": "15"
      }
    },
    {
      "passed": true,
      "assessed_at": "2014-05-26T13:05:40-07:00",
      "links": {
        "user": "16"
      }
    }
  ]
}
```

Example Response:

```
{
  "results": [Result]
}
```

List live assessment results LiveAssessments::ResultsController#index

GET /api/v1/courses/:course\_id/live\_assessments/:assessment\_id/results

Scope: url:GET|/api/v1/courses/:course\_id/live\_assessments/:assessment\_id/results

Returns a paginated list of live assessment results

Request Parameters:

Parameter	Type	Description
user_id	integer	If set, restrict results to those for this user

Example Response:

```
{
  "results": [Result]
}
```

Create or find a live assessment LiveAssessments::AssessmentsController#create

POST /api/v1/courses/:course\_id/live\_assessments

Scope: url:POST|/api/v1/courses/:course\_id/live\_assessments

Creates or finds an existing live assessment with the given key and aligns it with the linked outcome

Example Request:

```
{
  "assessments": [
    {
      "key": "2014-05-27-Outcome-52",
      "title": "Tuesday's LiveAssessment",
      "links": {
        "outcome": "1"
      }
    }
  ]
}
```

Example Response:

```
{
  "links": {
    "assessments.results": "http://example.com/courses/1/live_assessments/5/results"
  },
  "assessments": [Assessment]
}
```

List live assessments LiveAssessments::AssessmentsController#index

GET /api/v1/courses/:course\_id/live\_assessments

Scope: url:GET|/api/v1/courses/:course\_id/live\_assessments

Returns a paginated list of live assessments.

Example Response:

```
{
  "links": {
    "assessments.results": "http://example.com/courses/1/live_assessments/(assessments.id)/results"
  },
  "assessments": [Assessment]
}
```



List user logins PseudonymsController#index

GET /api/v1/accounts/:account\_id/logins

Scope: url:GET|/api/v1/accounts/:account\_id/logins

GET /api/v1/users/:user\_id/logins

Scope: url:GET|/api/v1/users/:user\_id/logins

Given a user ID, return a paginated list of that user's logins for the given account.

API response field:

- account\_id  
The ID of the login's account.
- id  
The unique, numeric ID for the login.
- sis\_user\_id  
The login's unique SIS ID.
- integration\_id  
The login's unique integration ID.
- unique\_id  
The unique ID for the login.
- user\_id  
The unique ID of the login's user.
- authentication\_provider\_id  
The ID of the authentication provider that this login is associated with
- authentication\_provider\_type  
The type of the authentication provider that this login is associated with
- workflow\_state  
The current status of the login
- declared\_user\_type  
The declared intention for this user's role

Example Response:

```
{
  "account_id": 1,
  "id": 2,
  "sis_user_id": null,
  "unique_id": "belieber@example.com",
  "user_id": 2,
  "authentication_provider_id": 1,
  "authentication_provider_type": "facebook",
  "workflow_state": "active",
  "declared_user_type": null,
}
```

Kickoff password recovery flow PseudonymsController#forgot\_password

POST /api/v1/users/reset\_password

Scope: url:POST|/api/v1/users/reset\_password

Given a user email, generate a nonce and email it to the user

API response field:

- requested  
The recovery request status

Example Response:

```
{
  "requested": true
}
```

Create a user login PseudonymsController#create

POST /api/v1/accounts/:account\_id/logins

Scope: url:POST|/api/v1/accounts/:account\_id/logins

Create a new login for an existing user in the given account.

Request Parameters:

Parameter	Type	Description
user[id]	Required string	The ID of the user to create the login for.
login[unique_id]	Required string	The unique ID for the new login.
login[password]	string	The new login's password.
login[sis_user_id]	string	SIS ID for the login. To set this parameter, the caller must be able to manage SIS permissions on the account.
login[integration_id]	string	Integration ID for the login. To set this parameter, the caller must be able to manage SIS permissions on the account. The Integration ID is a secondary identifier useful for more complex SIS integrations.
login[authentication_provider_id]	string	The authentication provider this login is associated with. Logins associated with a specific provider can only be used with that provider. Legacy providers (LDAP, CAS, SAML) will search for logins string associated with them, or unassociated logins. New providers will only search for logins explicitly associated with them. This can be the integer ID of the provider, or the type of the provider (in which case, it will find the first matching provider).
login[declared_user_type]	string	The declared intention of the user type. This can be set, but does not change any Canvas functionality with respect to their access. A user can still be a teacher, admin, student, etc. in any particular context without regard to this setting. This can be used for administrative purposes for integrations to be able to more easily identify why the user was created. Valid values are: * administrative * observer * staff * student * student_other * teacher
user[existing_user_id]	string	A Canvas User ID to identify a user in a trusted account (alternative to 'id', 'existing_sis_user_id', or 'existing_integration_id'). This parameter is not available in OSS Canvas.
user[existing_integration_id]	string	An Integration ID to identify a user in a trusted account (alternative to 'id', 'existing_user_id', or 'existing_sis_user_id'). This parameter is not available in OSS Canvas.

Parameter	Type	Description
user[existing_sis_user_id]	string	An SIS User ID to identify a user in a trusted account (alternative to 'id', 'existing_integration_id', or 'existing_user_id'). This parameter is not available in OSS Canvas.
user[trusted_account]	string	The domain of the account to search for the user. This field is required when identifying a user in a trusted account. This parameter is not available in OSS Canvas.

Example Request:

```
#create a facebook login for user with ID 123
curl 'https://<canvas>/api/v1/accounts/<account_id>/logins' \
-F 'user[id]=123' \
-F 'login[unique_id]=112233445566' \
-F 'login[authentication_provider_id]=facebook' \
-H 'Authorization: Bearer <token>'

#create a login for user in another trusted account:
curl 'https://<canvas>/api/v1/accounts/<account_id>/logins' \
-F 'user[existing_user_sis_id]=51542' \
-F 'user[trusted_account]=canvas.example.edu' \
-F 'login[unique_id]=112233445566' \
-H 'Authorization: Bearer <token>'
```

Edit a user login PseudonymsController#update

PUT /api/v1/accounts/:account\_id/logins/:id

Scope: url:PUT /api/v1/accounts/:account\_id/logins/:id

Update an existing login for a user in the given account.

Request Parameters:

Parameter	Type	Description
login[unique_id]	string	The new unique ID for the login.
login[password]	string	The new password for the login. Can only be set by an admin user if admins are allowed to change passwords for the account.
login[sis_user_id]	string	SIS ID for the login. To set this parameter, the caller must be able to manage SIS permissions on the account.
login[integration_id]	string	Integration ID for the login. To set this parameter, the caller must be able to manage SIS permissions on the account. The Integration ID is a secondary identifier useful for more complex SIS integrations.
login[authentication_provider_id]	string	The authentication provider this login is associated with. Logins associated with a specific provider can only be used with that provider. Legacy providers (LDAP, CAS, SAML) will search for logins associated with them, or unassociated logins. New providers will only search for logins explicitly associated with them. This can be the integer ID of the provider, or the type of the provider (in which case, it will find the first matching provider).
login[workflow_state]	string	Used to suspend or re-activate a login. Allowed values: active, suspended
login[declared_user_type]	string	The declared intention of the user type. This can be set, but does not change any Canvas functionality with respect to their access. A user can still be a teacher, admin, student, etc. in any particular context without regard to this setting. This can be used for administrative purposes for integrations to be able to more easily identify why the user was created. Valid values are: * administrative * observer * staff * student * student_other * teacher
override_sis_stickiness	boolean	Default is true. If false, any fields containing "sticky" changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Example Request:

```
curl https://<canvas>/api/v1/accounts/:account_id/logins/:login_id \
-H "Authorization: Bearer <ACCESS-TOKEN>" \
-X PUT
```

Example Response:

```
{
  "id": 1,
  "user_id": 2,
  "account_id": 3,
  "unique_id": "bieber@example.com",
  "created_at": "2020-01-29T19:33:35Z",
  "sis_user_id": null,
  "integration_id": null,
  "authentication_provider_id": null,
  "workflow_state": "active",
  "declared_user_type": "teacher"
}
```

Delete a user login PseudonymsController#destroy

DELETE /api/v1/users/:user\_id/logins/:id

Scope: url:DELETE /api/v1/users/:user\_id/logins/:id

Delete an existing login.

Example Request:

```
curl https://<canvas>/api/v1/users/:user_id/logins/:login_id \
-H "Authorization: Bearer <ACCESS-TOKEN>" \
-X DELETE
```

Example Response:

```
{
  "unique_id": "bieber@example.com",
  "sis_user_id": null,
  "account_id": 1,
  "id": 12345,
  "user_id": 2
}
```

List LTI Resource Links Lti::ResourceLinksController#index

GET /api/v1/courses/:course\_id/lti\_resource\_links

Scope: url:GET /api/v1/courses/:course\_id/lti\_resource\_links

Returns all Resource Links in the specified course. This includes links that are associated with Assignments, Module Items, Collaborations, and that are embedded in rich content. This endpoint is paginated, and will return 50 links per page by default. Links are sorted by the order in which they were created.

Request Parameters:

Parameter	Type	Description
include_deleted	boolean	Include deleted resource links and links associated with deleted content in response. Default is false.
per_page	integer	The number of registrations to return per page. Defaults to 50.

Example Request:

```
This would return the first 50 LTI resource links for the course, with a Link header pointing to the next page
curl -X GET 'https://<canvas>/api/v1/courses/1/lti_resource_links' \
-H 'Authorization: Bearer <token>' \
```

Returns a list of [Lti::ResourceLink](#) objects

Show an LTI Resource Link Lti::ResourceLinksController#show

GET /api/v1/courses/:course\_id/lti\_resource\_links/:id

**Scope:** url:GET|/api/v1/courses/:course\_id/lti\_resource\_links/:id

Return details about the specified resource link. The ID can be in the standard Canvas format ("1"), or in these special formats:

- resource\_link\_uuid:<uuid> - Find the resource link by its resource\_link\_uuid
- lookup\_uuid:<uuid> - Find the resource link by its lookup\_uuid

Request Parameters:

Parameter	Type	Description
include_deleted	boolean	Include deleted resource links in search. Default is false.

Example Request:

This would return the specified LTI resource link  
curl -X GET 'https://<canvas>/api/v1/courses/1/lti\_resource\_links/lookup\_uuid:c522554a-d4be-49ef-b163-9c87fdc6a06f' \  
-H 'Authorization: Bearer <token>'

Returns a [Lti::ResourceLink](#) object

Create an LTI Resource Link [Lti::ResourceLinksController#create](#)

POST /api/v1/courses/:course\_id/lti\_resource\_links

**Scope:** url:POST|/api/v1/courses/:course\_id/lti\_resource\_links

Create a new LTI Resource Link in the specified course with the provided parameters.

**Caution!** Resource Links are usually created by the tool via LTI Deep Linking. The tool has no knowledge of links created via this API, and may not be able to handle or launch them.

Links created using this API cannot be associated with a specific piece of Canvas content, like an Assignment, Module Item, or Collaboration. Links created using this API are only suitable for embedding in rich content using the 'canvas\_launch\_url' provided in the API response.

This link will be associated with the ContextExternalTool available in this context that matches the provided url. If a matching tool is not found, the link will not be created and this will return an error.

Request Parameters:

Parameter	Type	Description
url	Required string	The launch URL for this resource link.
title	string	The title of the resource link.
custom	Hash	Custom parameters to be sent to the tool when launching this link.

Example Request:

This would create a new LTI resource link in the specified course  
curl -X POST 'https://<canvas>/api/v1/courses/1/lti\_resource\_links' \  
-H 'Authorization: Bearer <token>' \  
-d 'url=https://example.com/lti/launch/new\_content\_item/456' \  
-d 'title=New Content Item' \  
-d 'custom[hello]=world' \

Returns a [Lti::ResourceLink](#) object

Bulk Create LTI Resource Links [Lti::ResourceLinksController#bulk\\_create](#)

POST /api/v1/courses/:course\_id/lti\_resource\_links/bulk

**Scope:** url:POST|/api/v1/courses/:course\_id/lti\_resource\_links/bulk

Create up to 100 new LTI Resource Links in the specified course with the provided parameters.

**Caution!** Resource Links are usually created by the tool via LTI Deep Linking. The tool has no knowledge of links created via this API, and may not be able to handle or launch them.

Links created using this API cannot be associated with a specific piece of Canvas content, like an Assignment, Module Item, or Collaboration. Links created using this API are only suitable for embedding in rich content using the 'canvas\_launch\_url' provided in the API response.

Each link will be associated with the ContextExternalTool available in this context that matches the provided url. If a matching tool is not found, or any parameters are invalid, no links will be created and this will return an error.

Request Parameters:

Parameter	Type	Description
POST	string body [Required, Array]	The POST body should be a JSON array of objects containing the parameters for each link to create.
[url]	Required string	Each object must contain a launch URL.
[title]	string	Each object may contain a title.
[custom]	Hash	Custom parameters to be sent to the tool when launching this link.

Example Request:

This would create a new LTI resource link in the specified course  
curl -X POST 'https://<canvas>/api/v1/courses/1/lti\_resource\_links/bulk' \  
-H 'Authorization: Bearer <token>' \  
--json '[{"url":"https://example.com/lti/launch/new\_content\_item/456","title":"New Content Item","custom":{"hello":"world"}}]'

Returns a [Lti::ResourceLink](#) object

Update an LTI Resource Link [Lti::ResourceLinksController#update](#)

PUT /api/v1/courses/:course\_id/lti\_resource\_links/:id

**Scope:** url:PUT|/api/v1/courses/:course\_id/lti\_resource\_links/:id

Update the specified resource link with the provided parameters.

**Caution!** Changing existing links may result in launch errors.

Request Parameters:

Parameter	Type	Description
url	string	The launch URL for this resource link. <b>Caution!</b> Updating this to a URL that doesn't match the tool could result in errors when launching this link!
custom	Hash	Custom parameters to be sent to the tool when launching this link. <b>Caution!</b> Changing these from what the tool provided could result in errors if the tool doesn't see what it's expecting.
include_deleted	boolean	Update link even if it is deleted. Default is false.

Example Request:

This would update the specified LTI resource link  
curl -X PUT 'https://<canvas>/api/v1/courses/1/lti\_resource\_links/1' \  
-H 'Authorization: Bearer <token>' \  
-d 'url=https://example.com/lti/launch/new\_content\_item/456' \  
-d 'custom[hello]=world'

Returns a [Lti::ResourceLink](#) object

Delete an LTI Resource Link [Lti::ResourceLinksController#destroy](#)

DELETE /api/v1/courses/:course\_id/lti\_resource\_links/:id

**Scope:** url:DELETE|/api/v1/courses/:course\_id/lti\_resource\_links/:id

Delete the specified resource link. The ID can be in the standard Canvas format (“1”), or in these special formats:

- resource\_link\_uuid:<uuid> - Find the resource link by its resource\_link\_uuid
- lookup\_uuid:<uuid> - Find the resource link by its lookup\_uuid

Only links that are not associated with Assignments, Module Items, or Collaborations can be deleted.

Example Request:

```
This would return the specified LTI resource link
curl -X DELETE 'https://<canvas>/api/v1/courses/1/lti_resource_links/lookup_uuid:c522554a-d4be-49ef-b163-9c87fdc6ad6f' \
-H 'Authorization: Bearer <token>'
```

Returns a [LtiResourceLink](#) object

List media tracks for a Media Object or Attachment [MediaTracksController#index](#)

GET /api/v1/media\_objects/:media\_object\_id/media\_tracks

**Scope:** url:GET|/api/v1/media\_objects/:media\_object\_id/media\_tracks

GET /api/v1/media\_attachments/:attachment\_id/media\_tracks

**Scope:** url:GET|/api/v1/media\_attachments/:attachment\_id/media\_tracks

List the media tracks associated with a media object or attachment

Request Parameters:

Parameter	Type	Description
include[]	string	By default, index returns id, locale, kind, media_object_id, and user_id for each of the result MediaTracks. Use include[] to add additional fields. For example include[]=content Allowed values: content, webvtt_content, updated_at, created_at

Example Request:

```
curl https://<canvas>/api/v1/media_objects/<media_object_id>/media_tracks?include[]=content \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/media_attachments/<attachment_id>/media_tracks?include[]=content \
-H 'Authorization: Bearer <token>'
```

Returns a list of [MediaTrack](#) objects

Update Media Tracks [MediaTracksController#update](#)

PUT /api/v1/media\_objects/:media\_object\_id/media\_tracks

**Scope:** url:PUT|/api/v1/media\_objects/:media\_object\_id/media\_tracks

PUT /api/v1/media\_attachments/:attachment\_id/media\_tracks

**Scope:** url:PUT|/api/v1/media\_attachments/:attachment\_id/media\_tracks

Replace the media tracks associated with a media object or attachment with the array of tracks provided in the body. Update will delete any existing tracks not listed, leave untouched any tracks with no content field, and update or create tracks with a content field.

Request Parameters:

Parameter	Type	Description
include[]	string	By default, an update returns id, locale, kind, media_object_id, and user_id for each of the result MediaTracks. Use include[] to add additional fields. For example include[]=content Allowed values: content, webvtt_content, updated_at, created_at

Example Request:

```
curl -X PUT https://<canvas>/api/v1/media_objects/<media_object_id>/media_tracks?include[]=content \
-H 'Authorization: Bearer <token>' \
-d '{"locale": "en"}, {"locale": "af", "content": "1\r\n00:00:00,000 --> 00:00:01,251\r\nThis is the content\r\n"}'

curl -X PUT https://<canvas>/api/v1/media_attachments/<attachment_id>/media_tracks?include[]=content \
-H 'Authorization: Bearer <token>' \
-d '{"locale": "en"}, {"locale": "af", "content": "1\r\n00:00:00,000 --> 00:00:01,251\r\nThis is the content\r\n"}'
```

Returns a list of [MediaTrack](#) objects

List Media Objects [MediaObjectsController#index](#)

GET /api/v1/media\_objects

**Scope:** url:GET|/api/v1/media\_objects

GET /api/v1/courses/:course\_id/media\_objects

**Scope:** url:GET|/api/v1/courses/:course\_id/media\_objects

GET /api/v1/groups/:group\_id/media\_objects

**Scope:** url:GET|/api/v1/groups/:group\_id/media\_objects

GET /api/v1/media\_attachments

**Scope:** url:GET|/api/v1/media\_attachments

GET /api/v1/courses/:course\_id/media\_attachments

**Scope:** url:GET|/api/v1/courses/:course\_id/media\_attachments

GET /api/v1/groups/:group\_id/media\_attachments

**Scope:** url:GET|/api/v1/groups/:group\_id/media\_attachments

Returns media objects created by the user making the request. When using the second version, returns media objects associated with the given course.

Request Parameters:

Parameter	Type	Description
		Field to sort on. Default is “title”
		title
		sorts on user_entered_title if available, title if not.
sort	string	created_at
		sorts on the object’s creation time.
		Allowed values: title, created_at
		Sort direction. Default is “asc”
order	string	Allowed values: asc, desc

Parameter	Type	Description
		Array of data to exclude. By excluding "sources" and "tracks", the api will not need to query kaltura, which greatly speeds up its response.
		sources
exclude[]	string	Do not query kaltura for media_sources
		tracks
		Do not query kaltura for media_tracks
		Allowed values: sources, tracks

Example Request:

```
curl https://<canvas>/api/v1/media_objects?exclude[]=sources&exclude[]=tracks \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/courses/17/media_objects?exclude[]=sources&exclude[]=tracks \
-H 'Authorization: Bearer <token>'
```

Returns a list of [MediaObject](#) objects

[Update Media Object MediaObjectsController#update\\_media\\_object](#)

PUT /api/v1/media\_objects/:media\_object\_id

Scope: url:PUT|/api/v1/media\_objects/:media\_object\_id

PUT /api/v1/media\_attachments/:attachment\_id

Scope: url:PUT|/api/v1/media\_attachments/:attachment\_id

[List students selected for moderation ModerationSetController#index](#)

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/moderated\_students

Scope: url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/moderated\_students

Returns a paginated list of students selected for moderation

Returns a list of [User](#) objects

[Select students for moderation ModerationSetController#create](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/moderated\_students

Scope: url:POST|/api/v1/courses/:course\_id/assignments/:assignment\_id/moderated\_students

Returns an array of users that were selected for moderation

Request Parameters:

Parameter	Type	Description
student_ids[]	number	user ids for students to select for moderation

Returns a list of [User](#) objects

[Bulk select provisional grades ProvisionalGradesController#bulk\\_select](#)

PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/bulk\_select

Scope: url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/bulk\_select

Choose which provisional grades will be received by associated students for an assignment. The caller must be the final grader for the assignment or an admin with :select\_final\_grade rights.

Example Response:

```
{
  "assignment_id": 867,
  "student_id": 5309,
  "selected_provisional_grade_id": 53669
}
```

[Show provisional grade status for a student ProvisionalGradesController#status](#)

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/status

Scope: url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/status

Tell whether the student's submission needs one or more provisional grades.

Request Parameters:

Parameter	Type	Description
student_id	integer	The id of the student to show the status for

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1/assignments/2/provisional_grades/status?student_id=1'
```

Example Response:

```
{ "needs_provisional_grade": false }
```

[Select provisional grade ProvisionalGradesController#select](#)

PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/:provisional\_grade\_id/select

Scope: url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/:provisional\_grade\_id/select

Choose which provisional grade the student should receive for a submission. The caller must be the final grader for the assignment or an admin with :select\_final\_grade rights.

Example Response:

```
{
  "assignment_id": 867,
  "student_id": 5309,
  "selected_provisional_grade_id": 53669
}
```

[Publish provisional grades for an assignment ProvisionalGradesController#publish](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/publish

Scope: url:POST|/api/v1/courses/:course\_id/assignments/:assignment\_id/provisional\_grades/publish

Publish the selected provisional grade for all submissions to an assignment. Use the "Select provisional grade" endpoint to choose which provisional grade to publish for a particular submission.

Students not in the moderation set will have their one and only provisional grade published.

WARNING: This is irreversible. This will overwrite existing grades in the gradebook.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1/assignments/2/provisional_grades/publish' \
-X POST
```

Show provisional grade status for a student AnonymousProvisionalGradesController#status

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_provisional\_grades/status

Scope: url:GET[/api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_provisional\_grades/status]

Determine whether or not the student's submission needs one or more provisional grades.

Request Parameters:

Parameter	Type	Description
anonymous_id	string	The id of the student to show the status for

Example Request:

curl 'https://<canvas>/api/v1/courses/1/assignments/2/anonymous\_provisional\_grades/status?anonymous\_id=1'

Example Response:

{ "needs\_provisional\_grade": false }

List modules ContextModulesApiController#index

GET /api/v1/courses/:course\_id/modules

Scope: url:GET[/api/v1/courses/:course\_id/modules]

A paginated list of the modules in a course

Request Parameters:

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"items": Return module items inline if possible. This parameter suggests that Canvas return module items directly in the Module object JSON, to avoid having to make separate API requests for each module when enumerating modules and items. Canvas is free to omit 'items' for any particular module if it deems them too numerous to return inline. Callers must be prepared to use the <a href="#">List Module Items API</a> if items are not returned.</li><li>"content_details": Requires 'items'. Returns additional details with module items specific to their associated content items. Includes standard lock information for each item.</li></ul> <p>Allowed values: items, content_details</p>
search_term	string	The partial name of the modules (and module items, if 'items' is specified with include[]) to match and return.
student_id	string	Returns module completion information for the student with this id.

Example Request:

curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/222/modules

Returns a list of [Module](#) objects

Show module ContextModulesApiController#show

GET /api/v1/courses/:course\_id/modules/:id

Scope: url:GET[/api/v1/courses/:course\_id/modules/:id]

Get information about a single module

Request Parameters:

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"items": Return module items inline if possible. This parameter suggests that Canvas return module items directly in the Module object JSON, to avoid having to make separate API requests for each module when enumerating modules and items. Canvas is free to omit 'items' for any particular module if it deems them too numerous to return inline. Callers must be prepared to use the <a href="#">List Module Items API</a> if items are not returned.</li><li>"content_details": Requires 'items'. Returns additional details with module items specific to their associated content items. Includes standard lock information for each item.</li></ul> <p>Allowed values: items, content_details</p>
student_id	string	Returns module completion information for the student with this id.

Example Request:

curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/222/modules/123

Returns a [Module](#) object

Create a module ContextModulesApiController#create

POST /api/v1/courses/:course\_id/modules

Scope: url:POST[/api/v1/courses/:course\_id/modules]

Create and return a new module

Request Parameters:

Parameter	Type	Description
module[name]	Required string	The name of the module
module[unlock_at]	DateTime	The date the module will unlock
module[position]	integer	The position of this module in the course (1-based)
module[require_sequential_progress]	boolean	Whether module items must be unlocked in order
module[prerequisite_module_ids[]]	string	IDs of Modules that must be completed before this one is unlocked. Prerequisite modules must precede this module (i.e. have a lower position value), otherwise they will be ignored
module[publish_final_grade]	boolean	Whether to publish the student's final grade for the course upon completion of this module.

Example Request:

curl https://<canvas>/api/v1/courses/<course\_id>/modules \
-X POST \
-H 'Authorization: Bearer <token>' \
-d 'module[name]=module' \
-d 'module[position]=2' \
-d 'module[prerequisite\_module\_ids[]]=121' \
-d 'module[prerequisite\_module\_ids[]]=122'

Returns a [Module](#) object

Update a module ContextModulesApiController#update

PUT /api/v1/courses/:course\_id/modules/:id

Scope: url:PUT[/api/v1/courses/:course\_id/modules/:id]

Update and return an existing module

Request Parameters:

Parameter	Type	Description
module[name]	string	The name of the module
module[unlock_at]	DateTime	The date the module will unlock
module[position]	integer	The position of the module in the course (1-based)
module[require_sequential_progress]	boolean	Whether module items must be unlocked in order
module[prerequisite_module_ids[]]	string	IDs of Modules that must be completed before this one is unlocked Prerequisite modules must precede this module (i.e. have a lower position value), otherwise they will be ignored
module[publish_final_grade]	boolean	Whether to publish the student's final grade for the course upon completion of this module.
module[published]	boolean	Whether the module is published and visible to students

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id> \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'module[name]=module' \
-d 'module[position]=2' \
-d 'module[prerequisite_module_ids][]=121' \
-d 'module[prerequisite_module_ids][]=122'
```

Returns a [Module](#) object

Delete module ContextModulesApiController#destroy

DELETE /api/v1/courses/:course\_id/modules/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/modules/:id

Delete a module

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id> \
-X Delete \
-H 'Authorization: Bearer <token>'
```

Returns a [Module](#) object

Re-lock module progressions ContextModulesApiController#relock

PUT /api/v1/courses/:course\_id/modules/:id/relock

Scope: url:PUT|/api/v1/courses/:course\_id/modules/:id/relock

Resets module progressions to their default locked state and recalculates them based on the current requirements.

Adding progression requirements to an active course will not lock students out of modules they have already unlocked unless this action is called.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/relock \
-X PUT \
-H 'Authorization: Bearer <token>'
```

Returns a [Module](#) object

List module items ContextModuleItemsApiController#index

GET /api/v1/courses/:course\_id/modules/:module\_id/items

Scope: url:GET|/api/v1/courses/:course\_id/modules/:module\_id/items

A paginated list of the items in a module

Request Parameters:

Parameter	Type	Description
include[]	string	If included, will return additional details specific to the content associated with each item. Refer to the <a href="#">Module Item specification</a> for more details. Includes standard lock information for each item. Allowed values: content_details
search_term	string	The partial title of the items to match and return.
student_id	string	Returns module completion information for the student with this id.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/222/modules/123/items
```

Returns a list of [ModuleItem](#) objects

Show module item ContextModuleItemsApiController#show

GET /api/v1/courses/:course\_id/modules/:module\_id/items/:id

Scope: url:GET|/api/v1/courses/:course\_id/modules/:module\_id/items/:id

Get information about a single module item

Request Parameters:

Parameter	Type	Description
include[]	string	If included, will return additional details specific to the content associated with this item. Refer to the <a href="#">Module Item specification</a> for more details. Includes standard lock information for each item. Allowed values: content_details
student_id	string	Returns module completion information for the student with this id.

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/222/modules/123/items/768
```

Returns a [ModuleItem](#) object

Create a module item ContextModuleItemsApiController#create

POST /api/v1/courses/:course\_id/modules/:module\_id/items

Scope: url:POST|/api/v1/courses/:course\_id/modules/:module\_id/items

Create and return a new module item

Request Parameters:

Parameter	Type	Description
module_item[title]	string	The name of the module item and associated content
module_item[type]	Required string	The type of content linked to the item Allowed values: File, Page, Discussion, Assignment, Quiz, SubHeader, ExternalUrl, ExternalTool
module_item[content_id]	Required string	The id of the content to link to the module item. Required, except for 'ExternalUrl', 'Page', and 'SubHeader' types.
module_item[position]	integer	The position of this item in the module (1-based).
module_item[indent]	integer	0-based indent level; module items may be indented to show a hierarchy
module_item[page_url]	string	Suffix for the linked wiki page (e.g. 'front-page'). Required for 'Page' type.
module_item[external_url]	string	External url that the item points to. [Required for 'ExternalUrl' and 'ExternalTool' types.
module_item[new_tab]	boolean	Whether the external tool opens in a new tab. Only applies to 'ExternalTool' type.
module_item[completion_requirement][type]	string	Completion requirement for this module item. "must_view": Applies to all item types "must_contribute": Only applies to "Assignment", "Discussion", and "Page" types "must_submit", "min_score": Only apply to "Assignment" and "Quiz" types "must_mark_done": Only applies to "Assignment" and "Page" types Inapplicable types will be ignored Allowed values: must_view, must_contribute, must_submit, must_mark_done
module_item[completion_requirement][min_score]	integer	Minimum score required to complete. Required for completion_requirement type 'min_score'.
module_item[iframe][width]	integer	Width of the ExternalTool on launch
module_item[iframe][height]	integer	Height of the ExternalTool on launch

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items \
-X POST \
-H 'Authorization: Bearer <token>' \
-d 'module_item[title]=module_item' \
-d 'module_item[type]=ExternalTool' \
-d 'module_item[content_id]=10' \
-d 'module_item[position]=2' \
-d 'module_item[indent]=1' \
-d 'module_item[new_tab]=true' \
-d 'module_item[iframe][width]=300' \
-d 'module_item[iframe][height]=200'
```

Returns a [ModuleItem](#) object

Update a module item [ContextModuleItemsApiController#update](#)

PUT /api/v1/courses/:course\_id/modules/:module\_id/items/:id

Scope: url: PUT /api/v1/courses/:course\_id/modules/:module\_id/items/:id

Update and return an existing module item

Request Parameters:

Parameter	Type	Description
module_item[title]	string	The name of the module item
module_item[position]	integer	The position of this item in the module (1-based)
module_item[indent]	integer	0-based indent level; module items may be indented to show a hierarchy
module_item[external_url]	string	External url that the item points to. Only applies to 'ExternalUrl' type.
module_item[new_tab]	boolean	Whether the external tool opens in a new tab. Only applies to 'ExternalTool' type.
module_item[completion_requirement][type]	string	Completion requirement for this module item. "must_view": Applies to all item types "must_contribute": Only applies to "Assignment", "Discussion", and "Page" types "must_submit", "min_score": Only apply to "Assignment" and "Quiz" types "must_mark_done": Only applies to "Assignment" and "Page" types Inapplicable types will be ignored Allowed values: must_view, must_contribute, must_submit, must_mark_done
module_item[completion_requirement][min_score]	integer	Minimum score required to complete, Required for completion_requirement type 'min_score'.
module_item[published]	boolean	Whether the module item is published and visible to students.
module_item[module_id]	string	Move this item to another module by specifying the target module id here. The target module must be in the same course.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items/<item_id> \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'module_item[position]=2' \
-d 'module_item[indent]=1' \
-d 'module_item[new_tab]=true'
```

Returns a [ModuleItem](#) object

Select a mastery path [ContextModuleItemsApiController#select\\_mastery\\_path](#)

POST /api/v1/courses/:course\_id/modules/:module\_id/items/:id/select\_mastery\_path

Scope: url: POST /api/v1/courses/:course\_id/modules/:module\_id/items/:id/select\_mastery\_path

Select a mastery path when module item includes several possible paths. Requires Mastery Paths feature to be enabled. Returns a compound document with the assignments included in the given path and any module items related to those assignments

Request Parameters:

Parameter	Type	Description
assignment_set_id	string	Assignment set chosen, as specified in the mastery_paths portion of the context module item response
student_id	string	Which student the selection applies to. If not specified, current user is implied.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items/<item_id>/select_master_path \
-X POST \
-H 'Authorization: Bearer <token>' \
-d 'assignment_set_id=2992'
```

Delete module item [ContextModuleItemsApiController#destroy](#)

DELETE /api/v1/courses/:course\_id/modules/:module\_id/items/:id



**Scope:** url:DELETE[/api/v1/courses/:course\_id/modules/:module\_id/items/:id

Delete a module item

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items/<item_id> \
-X Delete \
-H 'Authorization: Bearer <token>'
```

Returns a [ModuleItem](#) object

[Mark module item as done/not done ContextModuleItemsApiController#mark\\_as\\_done](#)

**PUT /api/v1/courses/:course\_id/modules:module\_id/items:id/done**

**Scope:** url:PUT[/api/v1/courses/:course\_id/modules/:module\_id/items/:id/done

Mark a module item as done/not done. Use HTTP method PUT to mark as done, and DELETE to mark as not done.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items/<item_id>/done \
-X Put \
-H 'Authorization: Bearer <token>'
```

[Get module item sequence ContextModuleItemsApiController#item\\_sequence](#)

**GET /api/v1/courses/:course\_id/module\_item\_sequence**

**Scope:** url:GET[/api/v1/courses/:course\_id/module\_item\_sequence

Given an asset in a course, find the ModuleItem it belongs to, the previous and next Module Items in the course sequence, and also any applicable mastery path rules

**Request Parameters:**

Parameter	Type	Description
asset_type	string	The type of asset to find module sequence information for. Use the ModuleItem if it is known (e.g., the user navigated from a module item), since this will avoid ambiguity if the asset appears more than once in the module sequence.  Allowed values: ModuleItem, File, Page, Discussion, Assignment, Quiz, ExternalTool
asset_id	integer	The id of the asset (or the url in the case of a Page)

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/module_item_sequence?asset_type=Assignment&asset_id=123 \
-H 'Authorization: Bearer <token>'
```

Returns a [ModuleItemSequence](#) object

[Mark module item read ContextModuleItemsApiController#mark\\_item\\_read](#)

**POST /api/v1/courses/:course\_id/modules:module\_id/items:id/mark\_read**

**Scope:** url:POST[/api/v1/courses/:course\_id/modules:module\_id/items:id/mark\_read

Fulfills “must view” requirement for a module item. It is generally not necessary to do this explicitly, but it is provided for applications that need to access external content directly (bypassing the html\_url redirect that normally allows Canvas to fulfill “must view” requirements).

This endpoint cannot be used to complete requirements on locked or unpublished module items.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/modules/<module_id>/items/<item_id>/mark_read \
-X POST \
-H 'Authorization: Bearer <token>'
```

[List a module's overrides ModuleAssignmentOverridesController#index](#)

**GET /api/v1/courses/:course\_id/modules:context\_module\_id/assignment\_overrides**

**Scope:** url:GET[/api/v1/courses/:course\_id/modules/:context\_module\_id/assignment\_overrides

Returns a paginated list of AssignmentOverrides that apply to the ContextModule.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/:course_id/modules/:context_module_id/assignment_overrides \
-H 'Authorization: Bearer <token>'
```

Returns a list of [ModuleAssignmentOverride](#) objects

[Update a module's overrides ModuleAssignmentOverridesController#bulk\\_update](#)

**PUT /api/v1/courses/:course\_id/modules:context\_module\_id/assignment\_overrides**

**Scope:** url:PUT[/api/v1/courses/:course\_id/modules/:context\_module\_id/assignment\_overrides

Accepts a list of overrides and applies them to the ContextModule. Returns 204 No Content response code if successful.

**Request Parameters:**

Parameter	Type	Description
overrides[]	Required Array	List of overrides to apply to the module. Overrides that already exist should include an ID and will be updated if needed. New overrides will be created for overrides in the list without an ID. Overrides not included in the list will be deleted. Providing an empty list will delete all of the module's overrides. Keys for each override object can include: 'id', 'title', 'student_ids', and 'course_section_id'.

**Example Request:**

```
curl https://<canvas>/api/v1/courses/:course_id/modules/:context_module_id/assignment_overrides \
-X PUT \
-H 'Authorization: Bearer <token>' \
-H 'Content-Type: application/json' \
-d '{
  "overrides": [
    {
      "id": 212,
      "course_section_id": 3564
    },
    {
      "title": "an assignment override",
      "student_ids": [1, 2, 3]
    }
  ]
}'
```

[List Course Memberships Lti::Ims::NamesAndRolesController#course\\_index](#)

**GET /api/lti/courses/:course\_id/names\_and\_roles**

**Scope:** url:GET[/api/lti/courses/:course\_id/names\_and\_roles

Return active NamesAndRoleMemberships in the given course.

**Request Parameters:**

Parameter	Type	Description
rliid	string	If specified only NamesAndRoleMemberships with access to the LTI link references by this ‘rliid’ will be included. Also causes the member array to be included for each returned NamesAndRoleMembership. If the ‘role’ parameter is also present, it will be ‘and-ed’ together with this parameter

Parameter	Type	Description
role	string parameter	If specified only NamesAndRoleMemberships having this role in the given Course will be included. Value must be a fully-qualified LTI/LIS role URN. If the 'rliid' parameter is also present, it will be 'and-ed' together with this parameter
limit	string	May be used to limit the number of NamesAndRoleMemberships returned in a page. Defaults to 50.
Returns a <a href="#">NamesAndRoleMemberships</a> object		

List Group Memberships Lti::Ims::NamesAndRolesController#group\_index

GET /api/lti/groups/:group\_id/names\_and\_roles

Scope: url:GET|/api/lti/groups/:group\_id/names\_and\_roles

Return active NamesAndRoleMemberships in the given group.

Request Parameters:

Parameter	Type	Description
'rliid'	string	If specified only NamesAndRoleMemberships with access to the LTI link references by this 'rliid' will be included. Also causes the member array to be included for each returned NamesAndRoleMembership. If the role parameter is also present, it will be 'and-ed' together with this parameter
role	string	If specified only NamesAndRoleMemberships having this role in the given Group will be included. Value must be a fully-qualified LTI/LIS role URN. Further, only <a href="#">purl.imsglobal.org/vocab/lis/v2/membership#Member</a> and <a href="#">purl.imsglobal.org/vocab/lis/v2/membership#Manager</a> are supported. If the 'rliid' parameter is also present, it will be 'and-ed' together with this parameter
limit	string	May be used to limit the number of NamesAndRoleMemberships returned in a page. Defaults to 50.
Returns a <a href="#">NamesAndRoleMemberships</a> object		

Get a quiz item

GET /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Scope: url:GET|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Get details about a single item in a new quiz.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.
item_id	Required integer	The id of the item associated with the quiz.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12/items/123' \
-H 'Authorization: Bearer <token>'
```

Returns a [QuizItem](#) object

List quiz items

GET /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items

Scope: url:GET|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items

Get a list of items in a new quiz.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	no description

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/1/items' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [QuizItem](#) objects

Create a quiz item

POST /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items

Scope: url:POST|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items

Create a quiz item in a new quiz. Only [QuestionItem](#) types can be created.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.
item[position]	integer	The position of the item within the quiz.
item[points_possible]	number	The number of points available to score on this item. Must be positive.
item[entry_type]	Required string	The type of the item. Allowed values: <code>Item</code>
item[entry][title]	string	The question title.
item[entry][item_body]	Required string	The question stem (rich content).
item[entry][calculator_type]	string	Type of calculator the user will have access to during the question. Allowed values: <code>none</code> , <code>basic</code> , <code>scientific</code>
item[entry][feedback][neutral]	string	General feedback to show regardless of answer (rich content).
item[entry][feedback][correct]	string	Feedback to show if the question is answered correctly (rich content).

Parameter	Type	Description
item[entry][feedback][incorrect]	string	Feedback to show if the question is answered incorrectly (rich content).
item[entry][interaction_type_slug]	Required string	The type of question. One of 'multi-answer', 'matching', 'categorization', 'file-upload', 'formula', 'ordering', 'rich-fill-blank', 'hot-spot', 'choice', 'numeric', 'true-false', or 'essay'. See <a href="#">Appendix: Question Types</a> for more info about each type.
item[entry][interaction_data]	Required Object	An object that contains the question data. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][properties]	Object	An object that contains additional properties for some question types. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][scoring_data]	Required Object	An object that describes how to score the question. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][answer_feedback]	Object	Feedback provided for each answer (rich content, only available on 'choice' question types).
item[entry][scoring_algorithm]	Required string	The algorithm used to score the question. See <a href="#">Appendix: Question Types</a> for more info about this field.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12/items' \  
-X POST \  
-H 'Authorization Bearer <token>' \  
-d 'item[position]=1' \  
-d 'item[points_possible]=25.0' \  
-d 'item[properties]={}' \  
-d 'item[entry_type]=item' \  
-d 'item[entry][title]=Question 1' \  
-d 'item[entry][feedback][correct]=good job!' \  
-d 'item[entry][calculator_type]=none' \  
-d 'item[entry][interaction_data][word_limit_enabled]=true' \  
-d 'item[entry][item_body]=pWhat is 3 * 6?/p' \  
-d 'item[entry][interaction_type_slug]=essay' \  
-d 'item[entry][scoring_data][value]=1' \  
-d 'item[entry][scoring_algorithm]=None'
```

Returns a [QuizItem](#) object

Update a quiz item

PATCH /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Scope: url:PATCH/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Update a single quiz item in a new quiz. Only [QuestionItem](#) types can be updated.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.
item_id	Required integer	The id of the item associated with the quiz.
item[position]	integer	The position of the item within the quiz.
item[points_possible]	number	The number of points available to score on this item. Must be positive.
item[entry_type]	string	The type of the item. Allowed values: <code>item</code>
item[entry][title]	string	The question title.
item[entry][item_body]	string	The question stem (rich content).
item[entry][calculator_type]	string	Type of calculator the user will have access to during the question. Allowed values: <code>none</code> , <code>basic</code> , <code>scientific</code>
item[entry][feedback][neutral]	string	General feedback to show regardless of answer (rich content).
item[entry][feedback][correct]	string	Feedback to show if the question is answered correctly (rich content).
item[entry][feedback][incorrect]	string	Feedback to show if the question is answered incorrectly (rich content).
item[entry][interaction_type_slug]	string	The type of question. One of 'multi-answer', 'matching', 'categorization', 'file-upload', 'formula', 'ordering', 'rich-fill-blank', 'hot-spot', 'choice', 'numeric', 'true-false', or 'essay'. See <a href="#">Appendix: Question Types</a> for more info about each type.
item[entry][interaction_data]	Object	An object that contains the question data. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][properties]	Object	An object that contains additional properties for some question types. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][scoring_data]	Object	An object that describes how to score the question. See <a href="#">Appendix: Question Types</a> for more info about this field.
item[entry][answer_feedback]	Object	Feedback provided for each answer (rich content, only available on 'choice' question types).
item[entry][scoring_algorithm]	string	The algorithm used to score the question. See <a href="#">Appendix: Question Types</a> for more info about this field.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12/items/145' \  
-X PATCH \  
-H 'Authorization Bearer <token>' \  
-d 'item[points_possible]=25.0' \  
-d 'item[entry][title]=Question 1' \  
-d 'item[entry][calculator_type]=scientific'
```

Returns a [QuizItem](#) object

Delete a quiz item

DELETE /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Scope: url:DELETE/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/:item\_id

Delete a single quiz item in a new quiz.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description

Parameter	Type	Description
assignment_id	Required integer	The id of the assignment associated with the quiz.
item_id	Required integer	The id of the item associated with the quiz.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12/items/123' \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns a [QuizItem](#) object

[Get items media\\_upload\\_url](#)

**GET /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/media\_upload\_url**

**Scope:** url:GET|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/items/media\_upload\_url

Get a url for uploading media for use in hot-spot question types. See the hot-spot question type in the [Appendix: Question Types](#) for more details about using this endpoint.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	no description

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/1/items/media_upload_url' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{ "url": "http://s3_upload_url" }
```

[Get a new quiz](#)

**GET /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id**

**Scope:** url:GET|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id

Get details about a single new quiz.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12' \
-H 'Authorization: Bearer <token>'
```

Returns a [NewQuiz](#) object

[List new quizzes](#)

**GET /api/quiz/v1/courses/:course\_id/quizzes**

**Scope:** url:GET|/api/quiz/v1/courses/:course\_id/quizzes

Get a list of new quizzes.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [NewQuiz](#) objects

[Create a new quiz](#)

**POST /api/quiz/v1/courses/:course\_id/quizzes**

**Scope:** url:POST|/api/quiz/v1/courses/:course\_id/quizzes

Create a new quiz for the course.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
quiz[title]	string	The title of the quiz.
quiz[assignment_group_id]	integer	The ID of the quiz's assignment group.
quiz[points_possible]	number	The total point value given to the quiz. Must be positive.
quiz[due_at]	DateTime	When the quiz is due.
quiz[lock_at]	DateTime	When to lock the quiz.
quiz[unlock_at]	DateTime	When to unlock the quiz.
quiz[grading_type]	string	The type of grading the assignment receives. Allowed values: pass_fail, percent, letter_grade, gpa_scale, points
quiz[instructions]	string	Instructions for the quiz.
quiz[quiz_settings][calculator_type]	string	Specifies which type of Calculator a student can use during Quiz taking. Should be null if no calculator is allowed. Allowed values: none, basic, scientific

Parameter	Type	Description
quiz[quiz_settings][filter_ip_address]	boolean	Whether IP filtering is needed. Must be true for filters to take effect.
quiz[quiz_settings][filters][ips][]	string	Specifies ranges of IP addresses where the quiz can be taken from. Each range is an array like [start address, end address], or null if there's no restriction.
quiz[quiz_settings][multiple_attempts][multiple_attempts_enabled]	boolean	Whether multiple attempts for this quiz is true.
quiz[quiz_settings][multiple_attempts][attempt_limit]	boolean	Whether there is an attempt limit. Only set if multiple_attempts_enabled is true.
quiz[quiz_settings][multiple_attempts][max_attempts]	Positive Integer	The allowed attempts a student can take. If null, the allowed attempts are unlimited. Only used if attempt_limit is true.
quiz[quiz_settings][multiple_attempts][score_to_keep]	string	Whichever score to keep for the attempts. Only used if multiple_attempts_enabled is true. Allowed values: average, first, highest, latest
quiz[quiz_settings][multiple_attempts][cooling_period]	boolean	Whether there is a cooling (waiting) period. Only used if multiple_attempts_enabled is true.
quiz[quiz_settings][multiple_attempts][cooling_period_seconds]	Positive Integer	Required waiting period in seconds between attempts. If null, there is no required time. Only used if cooling_period is true
quiz[quiz_settings][one_at_a_time_type]	string	Specifies the settings for questions to display when quiz taking. Allowed values: none, question
quiz[quiz_settings][allow_backtracking]	boolean	Whether to allow user to return to previous questions when 'one_at_a_time_type' is set to 'question'.
quiz[quiz_settings][result_view_settings][result_view_restricted]	boolean	Whether the results view is restricted for students. Must be true for any student restrictions to be set.
quiz[quiz_settings][result_view_settings][display_points_awarded]	boolean	Whether points are shown. Must set result_view_restricted to true to use this parameter.
quiz[quiz_settings][result_view_settings][display_points_possible]	boolean	Whether points possible is shown. Must set result_view_restricted to true to use this parameter.
quiz[quiz_settings][result_view_settings][display_items]	boolean	Whether to show items in the results view. Must be true for any items restrictions to be set.
quiz[quiz_settings][result_view_settings][display_item_response]	boolean	Whether item response is shown. Only set if display_items is true. Must be true for display_item_response_qualifier, show_item_responses_at, hide_item_responses_at, and display_item_response_correctness to be set.
quiz[quiz_settings][result_view_settings][display_item_response_qualifier]	string	Specifies after which attempts student responses should be shown to them. Only used if display_item_response is true. Allowed values: always, once_per_attempt, after_last_attempt, once_after_last_attempt
quiz[quiz_settings][result_view_settings][show_item_responses_at]	DateTime	When student responses should be shown to them. Only used if display_item_response is true.
quiz[quiz_settings][result_view_settings][hide_item_responses_at]	DateTime	When student responses should be hidden from them. Only used if display_item_response is true.
quiz[quiz_settings][result_view_settings][display_item_response_correctness]	boolean	Whether item correctness is shown. Only set if display_item_response is true. Must be true for display_item_response_correctness_qualifier, show_item_response_correctness_at, hide_item_response_correctness_at and display_item_correct_answer to be set.
quiz[quiz_settings][result_view_settings][display_item_response_correctness_qualifier]	string	Specifies after which attempts student response correctness should be shown to them. Only used if display_item_response_correctness is true. Allowed values: always, after_last_attempt
quiz[quiz_settings][result_view_settings][show_item_response_correctness_at]	DateTime	When student response correctness should be shown to them. Only used if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][hide_item_response_correctness_at]	DateTime	When student response correctness should be hidden from them. Only used if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][display_item_correct_answer]	boolean	Whether correct answer is shown. Only set if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][display_item_feedback]	boolean	Whether Item feedback is shown. Only set if display_items is true.
quiz[quiz_settings][shuffle_answers]	boolean	Whether answers should be shuffled for students.
quiz[quiz_settings][shuffle_questions]	boolean	Whether questions should be shuffled for students.
quiz[quiz_settings][require_student_access_code]	boolean	Whether an access code is needed to take the quiz.
quiz[quiz_settings][student_access_code]	string	Access code to restrict quiz access. Should be null if no restriction.
quiz[quiz_settings][has_time_limit]	boolean	Whether there is a time limit for the quiz.
quiz[quiz_settings][session_time_limit_in_seconds]	Positive Integer	Limit the time a student can work on the quiz. Should be null if no restriction.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes' \  
-X POST \  
-H 'Authorization Bearer <token>' \  
-d 'quiz[title]=New quiz' \  
-d 'quiz[assignment_group_id]=1' \  
-d 'quiz[points_possible]=100.0' \  
-d 'quiz[due_at]=2023-01-02T00:00:00Z' \  
-d 'quiz[lock_at]=2023-01-03T00:00:00Z' \  
-d 'quiz[unlock_at]=2023-01-01T00:00:00Z' \  
-d 'quiz[grading_type]=points' \  
-d 'quiz[instructions]=Instructions for quiz' \  
-d 'quiz[quiz_settings][calculator_type]=scientific' \  
-d 'quiz[quiz_settings][filter_ip_address]=true' \  
-d 'quiz[quiz_settings][filters][ips]=[["10.0.0.0", "10.10.0.0"], ["12.0.0.0", "12.10.10.0"]]' \  
-d 'quiz[quiz_settings][one_at_a_time_type]=question' \  
-d 'quiz[quiz_settings][allow_backtracking]=true' \  
-d 'quiz[quiz_settings][shuffle_answers]=true' \  
-d 'quiz[quiz_settings][shuffle_questions]=true' \  
-d 'quiz[quiz_settings][require_student_access_code]=true' \  
-d 'quiz[quiz_settings][student_access_code]=12345' \  
-d 'quiz[quiz_settings][has_time_limit]=true' \  
-d 'quiz[quiz_settings][session_time_limit_in_seconds]=7500' \  
-d 'quiz[quiz_settings][multiple_attempts][max_attempts]=4' \  
-d 'quiz[quiz_settings][multiple_attempts][attempt_limit]=true' \  
-d 'quiz[quiz_settings][multiple_attempts][score_to_keep]=average' \  
-d 'quiz[quiz_settings][multiple_attempts][cooling_period]=true' \  
-d 'quiz[quiz_settings][multiple_attempts][cooling_period_seconds]=93600' \  
-d 'quiz[quiz_settings][multiple_attempts][multiple_attempts_enabled]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_items]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_feedback]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_response]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_response_qualifier]=always' \  
-d 'quiz[quiz_settings][result_view_settings][show_item_responses_at]=2023-01-01T00:00:00Z' \  
-d 'quiz[quiz_settings][result_view_settings][hide_item_responses_at]=2023-01-02T00:00:00Z' \  
-d 'quiz[quiz_settings][result_view_settings][display_points_awarded]=true' \  
-d 'quiz[quiz_settings][result_view_settings][result_view_restricted]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_points_possible]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_correct_answer]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_response_correctness]=true' \  
-d 'quiz[quiz_settings][result_view_settings][display_item_response_correctness_qualifier]=always' \  
-d 'quiz[quiz_settings][result_view_settings][show_item_response_correctness_at]=2023-01-01T00:00:00Z' \  
-d 'quiz[quiz_settings][result_view_settings][hide_item_response_correctness_at]=2023-01-02T00:00:00Z' \  

```

Returns a [NewQuiz](#) object

[Update a single quiz](#)

PATCH /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id

**Scope:** url:PATCH|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id

Update a single quiz for the course.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.
quiz[title]	string	The title of the quiz.
quiz[assignment_group_id]	integer	The ID of the quiz's assignment group.
quiz[points_possible]	number	The total point value given to the quiz. Must be positive.
quiz[due_at]	DateTime	When the quiz is due.
quiz[lock_at]	DateTime	When to lock the quiz.
quiz[unlock_at]	DateTime	When to unlock the quiz.
quiz[grading_type]	string	The type of grading the assignment receives. Allowed values: pass_fail, percent, letter_grade, gpa_scale, points
quiz[instructions]	string	Instructions for the quiz.
quiz[quiz_settings][calculator_type]	string	Specifies which type of Calculator a student can use during Quiz taking. Should be null if no calculator is allowed. Allowed values: none, basic, scientific
quiz[quiz_settings][filter_ip_address]	boolean	Whether IP filtering is needed. Must be true for filters to take effect.
quiz[quiz_settings][filters][ips[]]	string	Specifies ranges of IP addresses where the quiz can be taken from. Each range is an array like [start address, end address], or null if there's no restriction. Specifies the range of IP addresses where the quiz can be taken from. Should be null if there's no restriction.
quiz[quiz_settings][multiple_attempts][multiple_attempts_enabled]	boolean	Whether multiple attempts for this quiz is true.
quiz[quiz_settings][multiple_attempts][attempt_limit]	boolean	Whether there is an attempt limit. Only set if multiple_attempts_enabled is true.
quiz[quiz_settings][multiple_attempts][max_attempts]	Positive Integer	The allowed attempts a student can take. If null, the allowed attempts are unlimited. Only used if attempt_limit is true.
quiz[quiz_settings][multiple_attempts][score_to_keep]	string	Whichever score to keep for the attempts. Only used if multiple_attempts_enabled is true. Allowed values: average, first, highest, latest
quiz[quiz_settings][multiple_attempts][cooling_period]	boolean	Whether there is a cooling period. Only used if multiple_attempts_enabled is true.
quiz[quiz_settings][multiple_attempts][cooling_period_seconds]	Positive Integer	Required waiting period in seconds between attempts. If null, there is no required time. Only used if cooling_period is true.
quiz[quiz_settings][one_at_a_time_type]	string	Specifies the settings for questions to display when quiz taking. Allowed values: none, question
quiz[quiz_settings][allow_backtracking]	boolean	Whether to allow user to return to previous questions when 'one_at_a_time_type' is set to 'question'.
quiz[quiz_settings][result_view_settings][result_view_restricted]	boolean	Whether the results view is restricted for students. Must be true for any student restrictions to be set.
quiz[quiz_settings][result_view_settings][display_points_awarded]	boolean	Whether points are shown. Must set result_view_restricted to true to use this parameter.
quiz[quiz_settings][result_view_settings][display_points_possible]	boolean	Whether points possible is shown. Must set result_view_restricted to true to use this parameter.
quiz[quiz_settings][result_view_settings][display_items]	boolean	Whether to show items in the results view. Must be true for any items restrictions to be set.
quiz[quiz_settings][result_view_settings][display_item_response]	boolean	Whether item response is shown. Only set if display_items is true. Must be true for display_item_response_qualifier, show_item_responses_at, hide_item_responses_at, and display_item_response_correctness to be set.
quiz[quiz_settings][result_view_settings][display_item_response_qualifier]	string	Specifies after which attempts student responses should be shown to them. Only used if display_item_response is true. Allowed values: always, once_per_attempt, after_last_attempt, once_after_last_attempt
quiz[quiz_settings][result_view_settings][show_item_responses_at]	DateTime	When student responses should be shown to them. Only used if display_item_response is true.
quiz[quiz_settings][result_view_settings][hide_item_responses_at]	DateTime	When student responses should be hidden from them. Only used if display_item_response is true.
quiz[quiz_settings][result_view_settings][display_item_response_correctness]	boolean	Whether item correctness is shown. Only set if display_item_response is true. Must be true for display_item_response_correctness_qualifier, show_item_response_correctness_at, hide_item_response_correctness_at and display_item_correct_answer to be set.
quiz[quiz_settings][result_view_settings][display_item_response_correctness_qualifier]	string	Specifies after which attempts student response correctness should be shown to them. Only used if display_item_response_correctness is true. Allowed values: always, after_last_attempt
quiz[quiz_settings][result_view_settings][show_item_response_correctness_at]	DateTime	When student response correctness should be shown to them. Only used if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][hide_item_response_correctness_at]	DateTime	When student response correctness should be hidden from them. Only used if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][display_item_correct_answer]	boolean	Whether correct answer is shown. Only set if display_item_response_correctness is true.
quiz[quiz_settings][result_view_settings][display_item_feedback]	boolean	Whether Item feedback is shown. Only set if display_items is true.
quiz[quiz_settings][shuffle_answers]	boolean	Whether answers should be shuffled for students.
quiz[quiz_settings][shuffle_questions]	boolean	Whether questions should be shuffled for students.
quiz[quiz_settings][require_student_access_code]	boolean	Whether an access code is needed to take the quiz.

Parameter	Type	Description
quiz[quiz_settings][student_access_code]	string	Access code to restrict quiz access. Should be null if no restriction.
quiz[quiz_settings][has_time_limit]	boolean	Whether there is a time limit for the quiz.
quiz[quiz_settings][session_time_limit_in_seconds]	Positive Integer	Limit the time a student can work on the quiz. Should be null if no restriction.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12' \  
-X PATCH \  
-H 'Authorization: Bearer <token>' \  
-d 'quiz[title]=New quiz' \  
-d 'quiz[assignment_group_id]=1' \  
-d 'quiz[points_possible]=100.0' \  
-d 'quiz[due_at]=2023-01-02T00:00:00Z' \  
-d 'quiz[lock_at]=2023-01-03T00:00:00Z' \  
-d 'quiz[unlock_at]=2023-01-01T00:00:00Z' \  
-d 'quiz[grading_type]=points' \  
-d 'quiz[instructions]=Instructions for quiz' \  
-d 'quiz[quiz_settings][calculator_type]=scientific' \  
-d 'quiz[quiz_settings][filter_ip_address]=true' \  
-d 'quiz[quiz_settings][filters][ips]=[["10.0.0.0", "10.10.0.0"], ["12.0.0.0", "12.10.10.0"]]' \  
-d 'quiz[quiz_settings][one_at_a_time_type]=question' \  
-d 'quiz[quiz_settings][allow_backtracking]=true' \  
-d 'quiz[quiz_settings][shuffle_answers]=true' \  
-d 'quiz[quiz_settings][shuffle_questions]=true' \  
-d 'quiz[quiz_settings][require_student_access_code]=true' \  
-d 'quiz[quiz_settings][student_access_code]=12345'
```

Returns a [NewQuiz](#) object

Delete a new quiz

DELETE /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id

Scope: url:DELETE|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id

Delete a single new quiz.

Request Parameters:

Parameter	Type	Description
course_id	Required integer	no description
assignment_id	Required integer	The id of the assignment associated with the quiz.

Example Request:

```
curl 'https://<canvas>/api/quiz/v1/courses/1/quizzes/12' \  
-X DELETE \  
-H 'Authorization: Bearer <token>'
```

Returns a [NewQuiz](#) object

Create a quiz report

POST /api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/reports

Scope: url:POST|/api/quiz/v1/courses/:course\_id/quizzes/:assignment\_id/reports

Generate a new report for this quiz. Returns a progress object that can be used to track the progress of the report generation.

Responses

- 400 Bad Request If the specified report type or format is invalid
- 409 Conflict If a quiz report of the specified type is already being generated

Request Parameters:

Parameter	Type	Description
quiz_report[report_type]	Required string	The type of report to be generated. Allowed values: student_analysis, item_analysis
quiz_report[format]	Required string	The format of report to be generated. Allowed values: csv, json

Returns a [Progress](#) object

List preferences NotificationPreferencesController#index

GET /api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preferences

Scope: url:GET|/api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preferences

GET /api/v1/users/:user\_id/communication\_channels/:type/:address/notification\_preferences

Scope: url:GET|/api/v1/users/:user\_id/communication\_channels/:type/:address/notification\_preferences

Fetch all preferences for the given communication channel

Returns a list of [NotificationPreference](#) objects

List of preference categories NotificationPreferencesController#category\_index

GET /api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preference\_categories

Scope: url:GET|/api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preference\_categories

Fetch all notification preference categories for the given communication channel

Get a preference NotificationPreferencesController#show

GET /api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preferences/:notification

Scope: url:GET|/api/v1/users/:user\_id/communication\_channels/:communication\_channel\_id/notification\_preferences/:notification

GET /api/v1/users/:user\_id/communication\_channels/:type/:address/notification\_preferences/:notification

Scope: url:GET|/api/v1/users/:user\_id/communication\_channels/:type/:address/notification\_preferences/:notification

Fetch the preference for the given notification for the given communication channel

Returns a [NotificationPreference](#) object

Update a preference NotificationPreferencesController#update

PUT /api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preferences/:notification

Scope: url:PUT|/api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preferences/:notification

PUT /api/v1/users/self/communication\_channels/:type/:address/notification\_preferences/:notification

Scope: url:PUT|/api/v1/users/self/communication\_channels/:type/:address/notification\_preferences/:notification

Change the preference for a single notification for a single communication channel

Request Parameters:

Parameter	Type	Description
notification_preferences[frequency]	Required string	The desired frequency for this notification

Update preferences by category NotificationPreferencesController#update\_preferences\_by\_category

PUT /api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preference\_categories/:category

Scope: url:PUT//api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preference\_categories/:category

Change the preferences for multiple notifications based on the category for a single communication channel

Request Parameters:

Parameter	Type	Description
category	string	The name of the category. Must be parameterized (e.g. The category "Course Content" should be "course_content")
notification_preferences[frequency]	Required string	The desired frequency for each notification in the category

Update multiple preferences NotificationPreferencesController#update\_all

PUT /api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preferences

Scope: url:PUT//api/v1/users/self/communication\_channels/:communication\_channel\_id/notification\_preferences

PUT /api/v1/users/self/communication\_channels/:type:/address/notification\_preferences

Scope: url:PUT//api/v1/users/self/communication\_channels/:type:/address/notification\_preferences

Change the preferences for multiple notifications for a single communication channel at once

Request Parameters:

Parameter	Type	Description
notification_preferences[<X>][frequency]	Required string	The desired frequency for <X> notification

Create an Originality Report Lti::OriginalityReportsApiController#create

POST /api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report

Scope: url:POST//api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report

Create a new OriginalityReport for the specified file

Request Parameters:

Parameter	Type	Description
originality_report[file_id]	integer	The id of the file being given an originality score. Required if creating a report associated with a file.
originality_report[originality_score]	Required number	A number between 0 and 100 representing the measure of the specified file's originality.
originality_report[originality_report_url]	string	The URL where the originality report for the specified file may be found.
originality_report[originality_report_file_id]	integer	The ID of the file within Canvas that contains the originality report for the submitted file provided in the request URL.
originality_report[tool_setting][resource_type_code]	string	The resource type code of the resource handler Canvas should use for the LTI launch for viewing originality reports. If set Canvas will launch to the message with type 'basic-lti-launch-request' in the specified resource handler rather than using the originality_report_url.
originality_report[tool_setting][resource_url]	string	The URL Canvas should launch to when showing an LTI originality report. Note that this value is inferred from the specified resource handler's message "path" value (See 'resource_type_code') unless it is specified. If this parameter is used a 'resource_type_code' must also be specified.
originality_report[workflow_state]	string	May be set to "pending", "error", or "scored". If an originality score is provided a workflow state of "scored" will be inferred.
originality_report[error_message]	string	A message describing the error. If set, the "workflow_state" will be set to "error."
originality_report[attempt]	integer	If no 'file_id' is given, and no file is required for the assignment (that is, the assignment allows an online text entry), this parameter may be given to clarify which attempt number the report is for (in the case of resubmissions). If this field is omitted and no 'file_id' is given, the report will be created (or updated, if it exists) for the first submission attempt with no associated file.

Returns an [OriginalityReport](#) object

Edit an Originality Report Lti::OriginalityReportsApiController#update

PUT /api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report/id

Scope: url:PUT//api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report/:id

PUT /api/lti/assignments/:assignment\_id/files/:file\_id/originality\_report

Scope: url:PUT//api/lti/assignments/:assignment\_id/files/:file\_id/originality\_report

Modify an existing originality report. An alternative to this endpoint is to POST the same parameters listed below to the CREATE endpoint.

Request Parameters:

Parameter	Type	Description
originality_report[originality_score]	number	A number between 0 and 100 representing the measure of the specified file's originality.
originality_report[originality_report_url]	string	The URL where the originality report for the specified file may be found.
originality_report[originality_report_file_id]	integer	The ID of the file within Canvas that contains the originality report for the submitted file provided in the request URL.
originality_report[tool_setting][resource_type_code]	string	The resource type code of the resource handler Canvas should use for the LTI launch for viewing originality reports. If set Canvas will launch to the message with type 'basic-lti-launch-request' in the specified resource handler rather than using the originality_report_url.
originality_report[tool_setting][resource_url]	string	The URL Canvas should launch to when showing an LTI originality report. Note that this value is inferred from the specified resource handler's message "path" value (See 'resource_type_code') unless it is specified. If this parameter is used a 'resource_type_code' must also be specified.
originality_report[workflow_state]	string	May be set to "pending", "error", or "scored". If an originality score is provided a workflow state of "scored" will be inferred.
originality_report[error_message]	string	A message describing the error. If set, the "workflow_state" will be set to "error."

Returns an [OriginalityReport](#) object



[Show an Originality Report Lti::OriginalityReportsApiController#show](#)

**GET /api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report/:id**

**Scope:** url:GET//api/lti/assignments/:assignment\_id/submissions/:submission\_id/originality\_report/:id

**GET /api/lti/assignments/:assignment\_id/files/:file\_id/originality\_report**

**Scope:** url:GET//api/lti/assignments/:assignment\_id/files/:file\_id/originality\_report

Get a single originality report

Returns an [OriginalityReport](#) object

[Redirect to root outcome group for context OutcomeGroupsApiController#redirect](#)

**GET /api/v1/global/root\_outcome\_group**

**Scope:** url:GET//api/v1/global/root\_outcome\_group

**GET /api/v1/accounts/:account\_id/root\_outcome\_group**

**Scope:** url:GET//api/v1/accounts/:account\_id/root\_outcome\_group

**GET /api/v1/courses/:course\_id/root\_outcome\_group**

**Scope:** url:GET//api/v1/courses/:course\_id/root\_outcome\_group

Convenience redirect to find the root outcome group for a particular context. Will redirect to the appropriate outcome group's URL.

[Get all outcome groups for context OutcomeGroupsApiController#index](#)

**GET /api/v1/accounts/:account\_id/outcome\_groups**

**Scope:** url:GET//api/v1/accounts/:account\_id/outcome\_groups

**GET /api/v1/courses/:course\_id/outcome\_groups**

**Scope:** url:GET//api/v1/courses/:course\_id/outcome\_groups

[Get all outcome links for context OutcomeGroupsApiController#link\\_index](#)

**GET /api/v1/accounts/:account\_id/outcome\_group\_links**

**Scope:** url:GET//api/v1/accounts/:account\_id/outcome\_group\_links

**GET /api/v1/courses/:course\_id/outcome\_group\_links**

**Scope:** url:GET//api/v1/courses/:course\_id/outcome\_group\_links

[Show an outcome group OutcomeGroupsApiController#show](#)

**GET /api/v1/global/outcome\_groups/:id**

**Scope:** url:GET//api/v1/global/outcome\_groups/:id

**GET /api/v1/accounts/:account\_id/outcome\_groups/:id**

**Scope:** url:GET//api/v1/accounts/:account\_id/outcome\_groups/:id

**GET /api/v1/courses/:course\_id/outcome\_groups/:id**

**Scope:** url:GET//api/v1/courses/:course\_id/outcome\_groups/:id

[Update an outcome group OutcomeGroupsApiController#update](#)

**PUT /api/v1/global/outcome\_groups/:id**

**Scope:** url:PUT//api/v1/global/outcome\_groups/:id

**PUT /api/v1/accounts/:account\_id/outcome\_groups/:id**

**Scope:** url:PUT//api/v1/accounts/:account\_id/outcome\_groups/:id

**PUT /api/v1/courses/:course\_id/outcome\_groups/:id**

**Scope:** url:PUT//api/v1/courses/:course\_id/outcome\_groups/:id

Modify an existing outcome group. Fields not provided are left as is; unrecognized fields are ignored.

When changing the parent outcome group, the new parent group must belong to the same context as this outcome group, and must not be a descendant of this outcome group (i.e. no cycles allowed).

**Request Parameters:**

Parameter	Type	Description
title	string	The new outcome group title.
description	string	The new outcome group description.
vendor_guid	string	A custom GUID for the learning standard.
parent_outcome_group_id	integer	The id of the new parent outcome group.

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/2.json' \  
-X PUT \  
-F 'title=Outcome Group Title' \  
-F 'description=Outcome group description' \  
-F 'vendor_guid=customid9000' \  
-F 'parent_outcome_group_id=1' \  
-H 'Authorization: Bearer <token>'

curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/2.json' \  
-X PUT \  
--data-binary '{  
  "title": "Outcome Group Title",  
  "description": "Outcome group description",  
  "vendor_guid": "customid9000",  
  "parent_outcome_group_id": 1  
}' \  
-H 'Content-Type: application/json' \  
-H 'Authorization: Bearer <token>'
```

Returns an [OutcomeGroup](#) object

[Delete an outcome group OutcomeGroupsApiController#destroy](#)

**DELETE /api/v1/global/outcome\_groups/:id**

**Scope:** url:DELETE//api/v1/global/outcome\_groups/:id

**DELETE /api/v1/accounts/:account\_id/outcome\_groups/:id**

**Scope:** url:DELETE//api/v1/accounts/:account\_id/outcome\_groups/:id

**DELETE /api/v1/courses/:course\_id/outcome\_groups/:id**

**Scope:** url:DELETE|/api/v1/courses/:course\_id/outcome\_groups/:id

Deleting an outcome group deletes descendant outcome groups and outcome links. The linked outcomes themselves are only deleted if all links to the outcome were deleted.

Aligned outcomes cannot be deleted; as such, if all remaining links to an aligned outcome are included in this group's descendants, the group deletion will fail.

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/2.json' \
-X DELETE \
-H "Authorization: Bearer <token>"
```

Returns an [OutcomeGroup](#) object

[List linked outcomes OutcomeGroupsApiController#outcomes](#)

**GET /api/v1/global/outcome\_groups/:id/outcomes**

**Scope:** url:GET|/api/v1/global/outcome\_groups/:id/outcomes

**GET /api/v1/accounts/account\_id/outcome\_groups:id/outcomes**

**Scope:** url:GET|/api/v1/accounts/:account\_id/outcome\_groups/:id/outcomes

**GET /api/v1/courses/course\_id/outcome\_groups:id/outcomes**

**Scope:** url:GET|/api/v1/courses/:course\_id/outcome\_groups/:id/outcomes

A paginated list of the immediate OutcomeLink children of the outcome group.

**Request Parameters:**

Parameter	Type	Description
outcome_style	string	The detail level of the outcomes. Defaults to "abbrev". Specify "full" for more information.

Returns a list of [OutcomeLink](#) objects

[Create/link an outcome OutcomeGroupsApiController#link](#)

**POST /api/v1/global/outcome\_groups:id/outcomes**

**Scope:** url:POST|/api/v1/global/outcome\_groups/:id/outcomes

**PUT /api/v1/global/outcome\_groups:id/outcomes:outcome\_id**

**Scope:** url:PUT|/api/v1/global/outcome\_groups/:id/outcomes/:outcome\_id

**POST /api/v1/accounts/account\_id/outcome\_groups:id/outcomes**

**Scope:** url:POST|/api/v1/accounts/:account\_id/outcome\_groups/:id/outcomes

**PUT /api/v1/accounts/account\_id/outcome\_groups:id/outcomes:outcome\_id**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/outcome\_groups/:id/outcomes/:outcome\_id

**POST /api/v1/courses/course\_id/outcome\_groups:id/outcomes**

**Scope:** url:POST|/api/v1/courses/:course\_id/outcome\_groups/:id/outcomes

**PUT /api/v1/courses/course\_id/outcome\_groups:id/outcomes:outcome\_id**

**Scope:** url:PUT|/api/v1/courses/:course\_id/outcome\_groups/:id/outcomes/:outcome\_id

Link an outcome into the outcome group. The outcome to link can either be specified by a PUT to the link URL for a specific outcome (the outcome\_id in the PUT URLs) or by supplying the information for a new outcome (title, description, ratings, mastery\_points) in a POST to the collection.

If linking an existing outcome, the outcome\_id must identify an outcome available to this context; i.e. an outcome owned by this group's context, an outcome owned by an associated account, or a global outcome. With outcome\_id present, any other parameters (except move\_from) are ignored.

If defining a new outcome, the outcome is created in the outcome group's context using the provided title, description, ratings, and mastery points; the title is required but all other fields are optional. The new outcome is then linked into the outcome group.

If ratings are provided when creating a new outcome, an embedded rubric criterion is included in the new outcome. This criterion's mastery\_points default to the maximum points in the highest rating if not specified in the mastery\_points parameter. Any ratings lacking a description are given a default of "No description". Any ratings lacking a point value are given a default of 0. If no ratings are provided, the mastery\_points parameter is ignored.

**Request Parameters:**

Parameter	Type	Description
outcome_id	integer	The ID of the existing outcome to link.
move_from	integer	The ID of the old outcome group. Only used if outcome_id is present.
title	string	The title of the new outcome. Required if outcome_id is absent.
display_name	string	A friendly name shown in reports for outcomes with cryptic titles, such as common core standards names.
description	string	The description of the new outcome.
vendor_guid	string	A custom GUID for the learning standard.
mastery_points	integer	The mastery threshold for the embedded rubric criterion.
ratings[][description]	string	The description of a rating level for the embedded rubric criterion.
ratings[][points]	integer	The points corresponding to a rating level for the embedded rubric criterion.
calculation_method	string	The new calculation method. Defaults to "decaying_average" if the Outcomes New Decaying Average Calculation Method FF is ENABLED then Defaults to "weighted_average" Allowed values: weighted_average, decaying_average, n_mastery, latest, highest, average
calculation_int	integer	The new calculation int. Only applies if the calculation_method is "weighted_average", "decaying_average" or "n_mastery". Defaults to 65

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/outcomes/1.json' \
-X PUT \
-H "Authorization: Bearer <token>"

curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/outcomes.json' \
-X POST \
-F 'title=Outcome Title' \
-F 'display_name=Title for reporting' \
-F 'description=Outcome description' \
-F 'vendor_guid=custom0000' \
-F 'mastery_points=3' \
-F 'calculation_method=decaying_average' \
-F 'calculation_int=65' \
-F 'ratings[][description]=Exceeds Expectations' \
-F 'ratings[][points]=5' \
-F 'ratings[][description]=Meets Expectations' \
-F 'ratings[][points]=3' \
-F 'ratings[][description]=Does Not Meet Expectations' \
-F 'ratings[][points]=0' \
-H "Authorization: Bearer <token>"

curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/outcomes.json' \
```

```
-X POST \
--data-binary '{
  "title": "Outcome Title",
  "display_name": "Title for reporting",
  "description": "Outcome description",
  "vendor_guid": "customid9000",
  "mastery_points": 3,
  "ratings": {
    { "description": "Exceeds Expectations", "points": 5 },
    { "description": "Meets Expectations", "points": 3 },
    { "description": "Does Not Meet Expectations", "points": 0 }
  }
}' \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <token>"
```

Returns an [OutcomeLink](#) object

[Unlink an outcome OutcomeGroupsApiController#unlink](#)

**DELETE /api/v1/global/outcome\_groups/:id/outcomes/:outcome\_id**

**Scope:** url:DELETE|/api/v1/global/outcome\_groups/:id/outcomes/:outcome\_id

**DELETE /api/v1/accounts/:account\_id/outcome\_groups/:id/outcomes/:outcome\_id**

**Scope:** url:DELETE|/api/v1/accounts/:account\_id/outcome\_groups/:id/outcomes/:outcome\_id

**DELETE /api/v1/courses/:course\_id/outcome\_groups/:id/outcomes/:outcome\_id**

**Scope:** url:DELETE|/api/v1/courses/:course\_id/outcome\_groups/:id/outcomes/:outcome\_id

Unlinking an outcome only deletes the outcome itself if this was the last link to the outcome in any group in any context. Aligned outcomes cannot be deleted; as such, if this is the last link to an aligned outcome, the unlinking will fail.

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/outcomes/1.json' \
-X DELETE \
-H "Authorization: Bearer <token>"
```

Returns an [OutcomeLink](#) object

[List subgroups OutcomeGroupsApiController#subgroups](#)

**GET /api/v1/global/outcome\_groups/:id/subgroups**

**Scope:** url:GET|/api/v1/global/outcome\_groups/:id/subgroups

**GET /api/v1/accounts/:account\_id/outcome\_groups/:id/subgroups**

**Scope:** url:GET|/api/v1/accounts/:account\_id/outcome\_groups/:id/subgroups

**GET /api/v1/courses/:course\_id/outcome\_groups/:id/subgroups**

**Scope:** url:GET|/api/v1/courses/:course\_id/outcome\_groups/:id/subgroups

A paginated list of the immediate OutcomeGroup children of the outcome group.

Returns a list of [OutcomeGroup](#) objects

[Create a subgroup OutcomeGroupsApiController#create](#)

**POST /api/v1/global/outcome\_groups/:id/subgroups**

**Scope:** url:POST|/api/v1/global/outcome\_groups/:id/subgroups

**POST /api/v1/accounts/:account\_id/outcome\_groups/:id/subgroups**

**Scope:** url:POST|/api/v1/accounts/:account\_id/outcome\_groups/:id/subgroups

**POST /api/v1/courses/:course\_id/outcome\_groups/:id/subgroups**

**Scope:** url:POST|/api/v1/courses/:course\_id/outcome\_groups/:id/subgroups

Creates a new empty subgroup under the outcome group with the given title and description.

**Request Parameters:**

Parameter	Type	Description
title	Required string	The title of the new outcome group.
description	string	The description of the new outcome group.
vendor_guid	string	A custom GUID for the learning standard

**Example Request:**

```
curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/subgroups.json' \
-X POST \
-F 'title=Outcome Group Title' \
-F 'description=Outcome group description' \
-F 'vendor_guid=customid9000' \
-H "Authorization: Bearer <token>"

curl 'https://<canvas>/api/v1/accounts/1/outcome_groups/1/subgroups.json' \
-X POST \
--data-binary '{
  "title": "Outcome Group Title",
  "description": "Outcome group description",
  "vendor_guid": "customid9000"
}' \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <token>"
```

Returns an [OutcomeGroup](#) object

[Import an outcome group OutcomeGroupsApiController#import](#)

**POST /api/v1/global/outcome\_groups/:id/import**

**Scope:** url:POST|/api/v1/global/outcome\_groups/:id/import

**POST /api/v1/accounts/:account\_id/outcome\_groups/:id/import**

**Scope:** url:POST|/api/v1/accounts/:account\_id/outcome\_groups/:id/import

**POST /api/v1/courses/:course\_id/outcome\_groups/:id/import**

**Scope:** url:POST|/api/v1/courses/:course\_id/outcome\_groups/:id/import

Creates a new subgroup of the outcome group with the same title and description as the source group, then creates links in that new subgroup to the same outcomes that are linked in the source group. Recurses on the subgroups of the source group, importing them each in turn into the new subgroup.

Allows you to copy organizational structure, but does not create copies of the outcomes themselves, only new links.

The source group must be either global, from the same context as this outcome group, or from an associated account. The source group cannot be the root outcome group of its context.

**Request Parameters:**

Parameter	Type	Description
source_outcome_group_id	Required integer	The ID of the source outcome group.

Parameter	Type	Description
async	boolean	If true, perform action asynchronously. In that case, this endpoint will return a Progress object instead of an OutcomeGroup. Use the <a href="#">progress endpoint</a> to query the status of the operation. The imported outcome group id and url will be returned in the results of the Progress object as "outcome_group_id" and "outcome_group_url"

Example Request:

```
curl 'https://<canvas>/api/v1/accounts/2/outcome_groups/3/import.json' \  
-X POST \  
-F 'source_outcome_group_id=2' \  
-H 'Authorization: Bearer <token>'
```

Returns an [OutcomeGroup](#) object

[Import Outcomes OutcomeImportsApiController#create](#)

**POST /api/v1/accounts/:account\_id/outcome\_imports(/group/:learning\_outcome\_group\_id)**

**Scope:** url:POST /api/v1/accounts/:account\_id/outcome\_imports(/group/:learning\_outcome\_group\_id)

**POST /api/v1/courses/:course\_id/outcome\_imports(/group/:learning\_outcome\_group\_id)**

**Scope:** url:POST /api/v1/courses/:course\_id/outcome\_imports(/group/:learning\_outcome\_group\_id)

Import outcomes into Canvas.

For more information on the format that's expected here, please see the "Outcomes CSV" section in the API docs.

Request Parameters:

Parameter	Type	Description
import_type	string	Choose the data format for reading outcome data. With a standard Canvas install, this option can only be 'instructure_csv', and if unprovided, will be assumed to be so. Can be part of the query string.

There are two ways to post outcome import data - either via a multipart/form-data form-field-style attachment, or via a non-multipart raw post request.

'attachment' is required for multipart/form-data style posts. Assumed to be outcome data from a file upload form field named 'attachment'.

Examples:

```
curl -F attachment=@<filename> -H "Authorization: Bearer <token>" \  
'https://<canvas>/api/v1/accounts/<account_id>/outcome_imports?import_type=instructure_csv' \  
curl -F attachment=@<filename> -H "Authorization: Bearer <token>" \  
'https://<canvas>/api/v1/courses/<course_id>/outcome_imports?import_type=instructure_csv'
```

attachment

string If you decide to do a raw post, you can skip the 'attachment' argument, but you will then be required to provide a suitable Content-Type header. You are encouraged to also provide the 'extension' argument.

Examples:

```
curl -H 'Content-Type: text/csv' --data-binary @<filename>.csv \  
-H "Authorization: Bearer <token>" \  
'https://<canvas>/api/v1/accounts/<account_id>/outcome_imports?import_type=instructure_csv' \  
curl -H 'Content-Type: text/csv' --data-binary @<filename>.csv \  
-H "Authorization: Bearer <token>" \  
'https://<canvas>/api/v1/courses/<course_id>/outcome_imports?import_type=instructure_csv'
```

extension

string Recommended for raw post request style imports. This field will be used to distinguish between csv and other file format extensions that would usually be provided with the filename in the multipart post request scenario. If not provided, this value will be inferred from the Content-Type, falling back to csv-file format if all else fails.

Returns an [OutcomeImport](#) object

[Get Outcome import status OutcomeImportsApiController#show](#)

**GET /api/v1/accounts/:account\_id/outcome\_imports/:id**

**Scope:** url:GET /api/v1/accounts/:account\_id/outcome\_imports/:id

**GET /api/v1/courses/:course\_id/outcome\_imports/:id**

**Scope:** url:GET /api/v1/courses/:course\_id/outcome\_imports/:id

Get the status of an already created Outcome import. Pass 'latest' for the outcome import id for the latest import.

Examples:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/outcome_imports/<outcome_import_id>' \  
-H "Authorization: Bearer <token>" \  
curl 'https://<canvas>/api/v1/courses/<course_id>/outcome_imports/<outcome_import_id>' \  
-H "Authorization: Bearer <token>"
```

Returns an [OutcomeImport](#) object

[Get IDs of outcome groups created after successful import OutcomeImportsApiController#created\\_group\\_ids](#)

**GET /api/v1/accounts/:account\_id/outcome\_imports/:id/created\_group\_ids**

**Scope:** url:GET /api/v1/accounts/:account\_id/outcome\_imports/:id/created\_group\_ids

**GET /api/v1/courses/:course\_id/outcome\_imports/:id/created\_group\_ids**

**Scope:** url:GET /api/v1/courses/:course\_id/outcome\_imports/:id/created\_group\_ids

Get the IDs of the outcome groups created after a successful import. Pass 'latest' for the outcome import id for the latest import.

Examples:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/outcome_imports/outcomes_group_ids/<outcome_import_id>' \  
-H "Authorization: Bearer <token>" \  
curl 'https://<canvas>/api/v1/courses/<course_id>/outcome_imports/outcome_group_ids/<outcome_import_id>' \  
-H "Authorization: Bearer <token>"
```

[Get outcome results OutcomeResultsController#index](#)

**GET /api/v1/courses/:course\_id/outcome\_results**

**Scope:** url:GET /api/v1/courses/:course\_id/outcome\_results

Gets the outcome results for users and outcomes in the specified context.

used in sLMGB

Request Parameters:

Parameter	Type	Description
user_ids[]	integer	If specified, only the users whose ids are given will be included in the results. SIS ids can be used, prefixed by "sis_user_id:". It is an error to specify an id for a user who is not a student in the context.
outcome_ids[]	integer	If specified, only the outcomes whose ids are given will be included in the results. It is an error to specify an id for an outcome which is not linked to the context.
include[]	string	String, "alignments" "outcomes" "outcomes.alignments" "outcome_groups" "outcome_links" "outcome_paths" "users" Specify additional collections to be side loaded with the result. "alignments" includes only the alignments referenced by the returned results. "outcomes.alignments" includes all alignments referenced by outcomes in the context.
include_hidden	boolean	If true, results that are hidden from the learning mastery gradebook and student rollup scores will be included

Example Response:

```
{  
  outcome_results: [OutcomeResult]  
}
```

[Set outcome ordering for LMGB OutcomeResultsController#outcome\\_order](#)

POST /api/v1/courses/:course\_id/assign\_outcome\_order

**Scope:** url:POST /api/v1/courses/:course\_id/assign\_outcome\_order

Saves the ordering of outcomes in LMGB for a user

[Get outcome result rollups OutcomeResultsController#rollups](#)

GET /api/v1/courses/:course\_id/outcome\_rollups

**Scope:** url:GET /api/v1/courses/:course\_id/outcome\_rollups

Gets the outcome rollups for the users and outcomes in the specified context.

Request Parameters:

Parameter	Type	Description
aggregate	string	If specified, instead of returning one rollup for each user, all the user rollups will be combined into one rollup for the course that will contain the average (or median, see below) rollup score for each outcome. Allowed values: course
aggregate_stat	string	If aggregate rollups requested, then this value determines what statistic is used for the aggregate. Defaults to "mean" if this value is not specified. Allowed values: mean, median
user_ids[]	integer	If specified, only the users whose ids are given will be included in the results or used in an aggregate result. it is an error to specify an id for a user who is not a student in the context
outcome_ids[]	integer	If specified, only the outcomes whose ids are given will be included in the results. it is an error to specify an id for an outcome which is not linked to the context.
include[]	string	String, "courses" "outcomes" "outcomes.alignments" "outcome_groups" "outcome_links" "outcome_paths" "users" Specify additional collections to be side loaded with the result.
exclude[]	string	Specify additional values to exclude. "missing_user_rollups" excludes rollups for users without results. Allowed values: missing_user_rollups
sort_by	string	If specified, sorts outcome result rollups. "student" sorting will sort by a user's sortable name. "outcome" sorting will sort by the given outcome's rollup score. The latter requires specifying the "sort_outcome_id" parameter. By default, the sort order is ascending. Allowed values: student, outcome
sort_outcome_id	integer	If outcome sorting requested, then this determines which outcome to use for rollup score sorting.
sort_order	string	If sorting requested, then this allows changing the default sort order of ascending to descending. Allowed values: asc, desc
add_defaults	boolean	If defaults are requested, then color and mastery level defaults will be added to outcome ratings in the rollup. This will only take effect if the Account Level Mastery Scales FF is DISABLED
contributing_scores	boolean	If contributing scores are requested, then each individual outcome score will also include all graded artifacts that contributed to the outcome score

Example Response:

```
{
  "rollups": [OutcomeRollup],
  "linked": {
    // (Optional) Included if include[] has outcomes
    "outcomes": [Outcome],
    // (Optional) Included if aggregate is not set and include[] has users
    "users": [User],
    // (Optional) Included if aggregate is 'course' and include[] has courses
    "courses": [Course]
    // (Optional) Included if include[] has outcome_groups
    "outcome_groups": [OutcomeGroup],
    // (Optional) Included if include[] has outcome_links
    "outcome_links": [OutcomeLink],
    // (Optional) Included if include[] has outcome_paths
    "outcome_paths": [OutcomePath],
    // (Optional) Included if include[] has outcomes.alignments
    "outcomes.alignments": [OutcomeAlignment]
  }
}
```

[Show an outcome OutcomesApiController#show](#)

GET /api/v1/outcomes/:id

**Scope:** url:GET /api/v1/outcomes/:id

Returns the details of the outcome with the given id.

Request Parameters:

Parameter	Type	Description
add_defaults	boolean	If defaults are requested, then color and mastery level defaults will be added to outcome ratings in the result. This will only take effect if the Account Level Mastery Scales FF is DISABLED

Returns an [Outcome](#) object

[Update an outcome OutcomesApiController#update](#)

PUT /api/v1/outcomes/:id

**Scope:** url:PUT /api/v1/outcomes/:id

Modify an existing outcome. Fields not provided are left as is; unrecognized fields are ignored.

If any new ratings are provided, the combination of all new ratings provided completely replace any existing embedded rubric criterion; it is not possible to tweak the ratings of the embedded rubric criterion.

A new embedded rubric criterion's mastery\_points default to the maximum points in the highest rating if not specified in the mastery\_points parameter. Any new ratings lacking a description are given a default of "No description". Any new ratings lacking a point value are given a default of 0.

Request Parameters:

Parameter	Type	Description
title	string	The new outcome title.
display_name	string	A friendly name shown in reports for outcomes with cryptic titles, such as common core standards names.
description	string	The new outcome description.
vendor_guid	string	A custom GUID for the learning standard.
mastery_points	integer	The new mastery threshold for the embedded rubric criterion.

Parameter	Type	Description
ratings[] [description]	string	The description of a new rating level for the embedded rubric criterion.
ratings[][points]	integer	The points corresponding to a new rating level for the embedded rubric criterion.
calculation_method	string	The new calculation method. If the Outcomes New Decaying Average Calculation Method FF is ENABLED then "weighted_average" can be used and it is same as previous "decaying_average" and new "decaying_average" will have improved version of calculation.  Allowed values: weighted_average, decaying_average, n_mastery, latest, highest, average
calculation_int	integer	The new calculation int. Only applies if the calculation_method is "decaying_average" or "n_mastery"
add_defaults	boolean	If defaults are requested, then color and mastery level defaults will be added to outcome ratings in the result. This will only take effect if the Account Level Mastery Scales FF is DISABLED

Example Request:

```
curl 'https://<canvas>/api/v1/outcomes/1.json' \  
-X PUT \  
-F 'title=Outcome Title' \  
-F 'display_name=Title for reporting' \  
-F 'description=Outcome description' \  
-F 'vendor_guid=customid9901' \  
-F 'mastery_points=3' \  
-F 'calculation_method=decaying_average' \  
-F 'calculation_int=65' \  
-F 'ratings[][description]=Exceeds Expectations' \  
-F 'ratings[][points]=5' \  
-F 'ratings[][description]=Meets Expectations' \  
-F 'ratings[][points]=3' \  
-F 'ratings[][description]=Does Not Meet Expectations' \  
-F 'ratings[][points]=0' \  
-F 'ratings[][points]=0' \  
-H 'Authorization: Bearer <token>' \  
  
curl 'https://<canvas>/api/v1/outcomes/1.json' \  
-X PUT \  
--data-binary '{  
  "title": "Outcome Title",  
  "display_name": "Title for reporting",  
  "description": "Outcome description",  
  "vendor_guid": "customid9901",  
  "mastery_points": 3,  
  "ratings": [  
    {  
      "description": "Exceeds Expectations", "points": 5 },  
    {  
      "description": "Meets Expectations", "points": 3 },  
    {  
      "description": "Does Not Meet Expectations", "points": 0 }  
    ]  
}' \  
-H 'Content-Type: application/json' \  
-H 'Authorization: Bearer <token>'
```

Returns an [Outcome](#) object

[Get aligned assignments for an outcome in a course for a particular student OutcomesApiController#outcome\\_alignments](#)

GET /api/v1/courses/:course\_id/outcome\_alignments

Scope: url:GET|/api/v1/courses/:course\_id/outcome\_alignments

[Show front page WikiPagesApiController#show\\_front\\_page](#)

GET /api/v1/courses/:course\_id/front\_page

Scope: url:GET|/api/v1/courses/:course\_id/front\_page

GET /api/v1/groups/:group\_id/front\_page

Scope: url:GET|/api/v1/groups/:group\_id/front\_page

Retrieve the content of the front page

Example Request:

```
curl -H 'Authorization: Bearer <token>' \  
https://<canvas>/api/v1/courses/123/front_page
```

Returns a [Page](#) object

[Duplicate page WikiPagesApiController#duplicate](#)

POST /api/v1/courses/:course\_id/pages:url\_or\_id/duplicate

Scope: url:POST|/api/v1/courses/:course\_id/pages/:url\_or\_id/duplicate

Duplicate a wiki page

Example Request:

```
curl -X POST -H 'Authorization: Bearer <token>' \  
https://<canvas>/api/v1/courses/123/pages/14/duplicate
```

Returns a [Page](#) object

[Update/create front page WikiPagesApiController#update\\_front\\_page](#)

PUT /api/v1/courses/:course\_id/front\_page

Scope: url:PUT|/api/v1/courses/:course\_id/front\_page

PUT /api/v1/groups/:group\_id/front\_page

Scope: url:PUT|/api/v1/groups/:group\_id/front\_page

Update the title or contents of the front page

Request Parameters:

Parameter	Type	Description
wiki_page[title]	string	The title for the new page. NOTE: changing a page's title will change its url. The updated url will be returned in the result.
wiki_page[body]	string	The content for the new page.
		Which user roles are allowed to edit this page. Any combination of these roles is allowed (separated by commas).
		"teachers"
		Allows editing by teachers in the course.
		"students"
		Allows editing by students in the course.
wiki_page[editing_roles]	string	"members"
		For group wikis, allows editing by members of the group.
		"public"
		Allows editing by any user.
		Allowed values: teachers, students, members, public
wiki_page[notify_of_update]	boolean	Whether participants should be notified when this page changes.

Parameter	Type	Description
wiki_page[published]	boolean	Whether the page is published (true) or draft state (false).

Example Request:

```
curl -X PUT -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/front_page \
-d wiki_page[body]=Updated+body+text
```

Returns a [Page](#) object

List pages [WikiPagesApiController#index](#)

GET /api/v1/courses/:course\_id/pages

Scope: url: GET /api/v1/courses/:course\_id/pages

GET /api/v1/groups/:group\_id/pages

Scope: url: GET /api/v1/groups/:group\_id/pages

A paginated list of the wiki pages associated with a course or group

Request Parameters:

Parameter	Type	Description
sort	string	Sort results by this field.
		Allowed values: title, created_at, updated_at
order	string	The sorting order. Defaults to 'asc'.
		Allowed values: asc, desc
search_term	string	The partial title of the pages to match and return.
published	boolean	If true, include only published pages. If false, exclude published pages. If not present, do not filter on published status.
		<ul style="list-style-type: none"><li>“enrollments”: Optionally include the page body with each Page.</li></ul>
include[]	string	If this is a block_editor page, returns the block_editor_attributes.
		Allowed values: body

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/pages?sort=title&order=asc
```

Returns a list of [Page](#) objects

Create page [WikiPagesApiController#create](#)

POST /api/v1/courses/:course\_id/pages

Scope: url: POST /api/v1/courses/:course\_id/pages

POST /api/v1/groups/:group\_id/pages

Scope: url: POST /api/v1/groups/:group\_id/pages

Create a new wiki page

Request Parameters:

Parameter	Type	Description
wiki_page[title]	Required string	The title for the new page.
wiki_page[body]	string	The content for the new page.
		Which user roles are allowed to edit this page. Any combination of these roles is allowed (separated by commas).
		“teachers”
		Allows editing by teachers in the course.
		“students”
		Allows editing by students in the course.
wiki_page[editing_roles]	string	“members”
		For group wikis, allows editing by members of the group.
		“public”
		Allows editing by any user.
		Allowed values: teachers, students, members, public
wiki_page[notify_of_update]	boolean	Whether participants should be notified when this page changes.
wiki_page[published]	boolean	Whether the page is published (true) or draft state (false).
wiki_page[front_page]	boolean	Set an unhidden page as the front page (if true)
wiki_page[publish_at]	Date/Time	Schedule a future date/time to publish the page. This will have no effect unless the “Scheduled Page Publication” feature is enabled in the account. If a future date is supplied, the page will be unpublished and <a href="#">wiki page</a> will be ignored.

Example Request:

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/pages \
-d wiki_page[title]=New+page \
-d wiki_page[body]=New+body+text
```

Returns a [Page](#) object

Show page [WikiPagesApiController#show](#)

GET /api/v1/courses/:course\_id/pages/:url\_or\_id

Scope: url: GET /api/v1/courses/:course\_id/pages/:url\_or\_id

GET /api/v1/groups/:group\_id/pages/:url\_or\_id

Scope: url: GET /api/v1/groups/:group\_id/pages/:url\_or\_id

Retrieve the content of a wiki page

Example Request:

```
curl -H 'Authorization: Bearer <token>' \
  https://<canvas>/api/v1/courses/123/pages/the-page-identifier
```

Returns a [Page](#) object

[Update/create page WikiPagesApiController#update](#)

**PUT /api/v1/courses/:course\_id/pages:url\_or\_id**

**Scope:** url:PUT /api/v1/courses/:course\_id/pages/:url\_or\_id

**PUT /api/v1/groups:group\_id/pages:url\_or\_id**

**Scope:** url:PUT /api/v1/groups/:group\_id/pages/:url\_or\_id

Update the title or contents of a wiki page

NOTE: You cannot specify the ID when creating a page. If you pass a numeric value as the page identifier and that does not represent a page ID that already exists, it will be interpreted as a URL.

**Request Parameters:**

Parameter	Type	Description
wiki_page[title]	string	The title for the new page. NOTE: changing a page's title will change its url. The updated url will be returned in the result.
wiki_page[body]	string	The content for the new page.
		Which user roles are allowed to edit this page. Any combination of these roles is allowed (separated by commas). "teachers" Allows editing by teachers in the course. "students" Allows editing by students in the course.
wiki_page[editing_roles]	string	"members" For group wikis, allows editing by members of the group. "public" Allows editing by any user. Allowed values: teachers, students, members, public
wiki_page[notify_of_update]	boolean	Whether participants should be notified when this page changes.
wiki_page[published]	boolean	Whether the page is published (true) or draft state (false).
wiki_page[publish_at]	DateTime	Schedule a future date/time to publish the page. This will have no effect unless the "Scheduled Page Publication" feature is enabled in the account. If a future date is set and the page is already published, it will be unpublished.
wiki_page[front_page]	boolean	Set an unhidden page as the front page (if true)

**Example Request:**

```
curl -X PUT -H 'Authorization: Bearer <token>' \
  https://<canvas>/api/v1/courses/123/pages/the-page-identifier \
  -d 'wiki_page[body]=<updated+body>text'
```

Returns a [Page](#) object

[Delete page WikiPagesApiController#destroy](#)

**DELETE /api/v1/courses/:course\_id/pages:url\_or\_id**

**Scope:** url:DELETE /api/v1/courses/:course\_id/pages/:url\_or\_id

**DELETE /api/v1/groups:group\_id/pages:url\_or\_id**

**Scope:** url:DELETE /api/v1/groups/:group\_id/pages/:url\_or\_id

Delete a wiki page

**Example Request:**

```
curl -X DELETE -H 'Authorization: Bearer <token>' \
  https://<canvas>/api/v1/courses/123/pages/the-page-identifier
```

Returns a [Page](#) object

[List revisions WikiPagesApiController#revisions](#)

**GET /api/v1/courses/:course\_id/pages:url\_or\_id/revisions**

**Scope:** url:GET /api/v1/courses/:course\_id/pages/:url\_or\_id/revisions

**GET /api/v1/groups:group\_id/pages:url\_or\_id/revisions**

**Scope:** url:GET /api/v1/groups/:group\_id/pages/:url\_or\_id/revisions

A paginated list of the revisions of a page. Callers must have update rights on the page in order to see page history.

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
  https://<canvas>/api/v1/courses/123/pages/the-page-identifier/revisions
```

Returns a list of [PageRevision](#) objects

[Show revision WikiPagesApiController#show\\_revision](#)

**GET /api/v1/courses/:course\_id/pages:url\_or\_id/revisions/latest**

**Scope:** url:GET /api/v1/courses/:course\_id/pages/:url\_or\_id/revisions/latest

**GET /api/v1/groups:group\_id/pages:url\_or\_id/revisions/latest**

**Scope:** url:GET /api/v1/groups/:group\_id/pages/:url\_or\_id/revisions/latest

**GET /api/v1/courses/:course\_id/pages:url\_or\_id/revisions/:revision\_id**

**Scope:** url:GET /api/v1/courses/:course\_id/pages/:url\_or\_id/revisions/:revision\_id

**GET /api/v1/groups:group\_id/pages:url\_or\_id/revisions/:revision\_id**

**Scope:** url:GET /api/v1/groups/:group\_id/pages/:url\_or\_id/revisions/:revision\_id

Retrieve the metadata and optionally content of a revision of the page. Note that retrieving historic versions of pages requires edit rights.

**Request Parameters:**

Parameter	Type	Description
summary	boolean	If set, exclude page content from results



Example Request:

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/pages/the-page-identifier/revisions/latest

curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/pages/the-page-identifier/revisions/4
```

Returns a [PageRevision](#) object

[Revert to revision WikiPagesApiController#revert](#)

POST /api/v1/courses/:course\_id/pages/:url\_or\_id/revisions/:revision\_id

Scope: url:POST /api/v1/courses/:course\_id/pages/:url\_or\_id/revisions/:revision\_id

POST /api/v1/groups/:group\_id/pages/:url\_or\_id/revisions/:revision\_id

Scope: url:POST /api/v1/groups/:group\_id/pages/:url\_or\_id/revisions/:revision\_id

Revert a page to a prior revision.

Request Parameters:

Parameter	Type	Description
revision_id	Required integer	The revision to revert to (use the <a href="#">List Revisions API</a> to see available revisions)

Example Request:

```
curl -X POST -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/courses/123/pages/the-page-identifier/revisions/6
```

Returns a [PageRevision](#) object

[Get all Peer Reviews PeerReviewsApiController#index](#)

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/peer\_reviews

Scope: url:GET /api/v1/courses/:course\_id/assignments/:assignment\_id/peer\_reviews

GET /api/v1/sections/:section\_id/assignments/:assignment\_id/peer\_reviews

Scope: url:GET /api/v1/sections/:section\_id/assignments/:assignment\_id/peer\_reviews

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Get a list of all Peer Reviews for this assignment

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the peer review. Allowed values: submission_comments, user

Returns a list of [PeerReview](#) objects

[Create Peer Review PeerReviewsApiController#create](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Create a peer review for the assignment

Request Parameters:

Parameter	Type	Description
user_id	Required integer	user_id to assign as reviewer on this assignment

Returns a [PeerReview](#) object

[Delete Peer Review PeerReviewsApiController#destroy](#)

DELETE /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:DELETE /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

DELETE /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Scope: url:DELETE /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:submission\_id/peer\_reviews

Delete a peer review for the assignment

Request Parameters:

Parameter	Type	Description
user_id	Required integer	user_id to delete as reviewer on this assignment

Returns a [PeerReview](#) object

[Get a single assignment \(Lti\) Lti::PlagiarismAssignmentsApiController#show](#)

GET /api/lti/assignments/:assignment\_id

Scope: url:GET /api/lti/assignments/:assignment\_id

Get a single Canvas assignment by Canvas id or LTI id. Tool providers may only access assignments that are associated with their tool.

Request Parameters:

Parameter	Type	Description
user_id	string	The id of the user. Can be a Canvas or LTI id for the user.

Returns a [LtiAssignment](#) object

Get a single user (Lti) Lti::UsersApiController#show

GET /api/lti/users/:id

Scope: url:GET|/api/lti/users/:id

Get a single Canvas user by Canvas id or LTI id. Tool providers may only access users that have been assigned an assignment associated with their tool.

Returns an [User](#) object

Get all users in a group (Lti) Lti::UsersApiController#group\_index

GET /api/lti/groups/:group\_id/users

Scope: url:GET|/api/lti/groups/:group\_id/users

Get all Canvas users in a group. Tool providers may only access groups that belong to the context the tool is installed in.

Returns a list of [User](#) objects

Get a single submission Lti::SubmissionsApiController#show

GET /api/lti/assignments/:assignment\_id/submissions/:submission\_id

Scope: url:GET|/api/lti/assignments/:assignment\_id/submissions/:submission\_id

Get a single submission, based on submission id.

Get the history of a single submission Lti::SubmissionsApiController#history

GET /api/lti/assignments/:assignment\_id/submissions/:submission\_id/history

Scope: url:GET|/api/lti/assignments/:assignment\_id/submissions/:submission\_id/history

Get a list of all attempts made for a submission, based on submission id.

List planner items PlannerController#index

GET /api/v1/planner/items

Scope: url:GET|/api/v1/planner/items

GET /api/v1/users/:user\_id/planner/items

Scope: url:GET|/api/v1/users/:user\_id/planner/items

Retrieve the paginated list of objects to be shown on the planner for the current user with the associated planner override to override an item's visibility if set.

Planner items for a student may also be retrieved by a linked observer. Use the path that accepts a user\_id and supply the student's id.

Request Parameters:

Parameter	Type	Description
start_date	Date	Only return items starting from the given date. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
end_date	Date	Only return items up to the given date. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
context_codes[]	string	List of context codes of courses and/or groups whose items you want to see. If not specified, defaults to all contexts associated to the current user. Note that concluded courses will be ignored unless specified in the includes[] parameter. The format of this field is the context type, followed by an underscore, followed by the context id. For example: course_42, group_123
observed_user_id	string	Return planner items for the given observed user. Must be accompanied by context_codes[]. The user making the request must be observing the observed user in all the courses specified by context_codes[].
filter	string	Only return items that have new or unread activity Allowed values: new_activity

Example Response:

```
{
  "context_type": "Course",
  "course_id": 1,
  "planner_override": { ... planner override object ... }, // Associated PlannerOverride object if user has toggled visibility for the object on the planner
  "submissions": false, // The statuses of the user's submissions for this object
  "plannable_id": "123",
  "plannable_type": "discussion_topic",
  "plannable": { ... discussion topic object },
  "html_url": "/courses/1/discussion_topics/8"
},
{
  "context_type": "Course",
  "course_id": 1,
  "planner_override": {
    "id": 3,
    "plannable_type": "Assignment",
    "plannable_id": 1,
    "user_id": 2,
    "workflow_state": "active",
    "marked_complete": true, // A user-defined setting for marking items complete in the planner
    "dismissed": false, // A user-defined setting for hiding items from the opportunities list
    "deleted_at": null,
    "created_at": "2017-05-18T18:35:55Z",
    "updated_at": "2017-05-18T18:35:55Z"
  },
  "submissions": { // The status as it pertains to the current user
    "excused": false,
    "graded": false,
    "late": false,
    "missing": true,
    "needs_grading": false,
    "with_feedback": false
  },
  "plannable_id": "456",
  "plannable_type": "assignment",
  "plannable": { ... assignment object ... },
  "html_url": "http://canvas.instructure.com/courses/1/assignments/1#submit"
},
{
  "planner_override": null,
  "submissions": false, // false if no associated assignment exists for the plannable item
  "plannable_id": "789",
  "plannable_type": "planner_note",
  "plannable": {
    "id": 1,
    "todo_date": "2017-05-30T06:00:00Z",
    "title": "hello",
    "details": "world",
    "user_id": 2,
    "course_id": null,
    "workflow_state": "active",
    "created_at": "2017-05-30T16:29:04Z",
    "updated_at": "2017-05-30T16:29:15Z"
  },
  "html_url": "http://canvas.instructure.com/api/v1/planner_notes/1"
}
}
```

List planner notes PlannerNotesController#index

GET /api/v1/planner\_notes

Scope: url:GET|/api/v1/planner\_notes

Retrieve the paginated list of planner notes

Retrieve planner note for a user

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
start_date	DateTime	Only return notes with todo dates since the start_date (inclusive). No default. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
end_date	DateTime	Only return notes with todo dates before the end_date (inclusive). No default. The value should be formatted as: yyyy-mm-dd or ISO 8601 YYYY-MM-DDTHH:MM:SSZ. If end_date and start_date are both specified and equivalent, then only notes with todo dates on that day are returned.
context_codes[]	string	List of context codes of courses whose notes you want to see. If not specified, defaults to all contexts that the user belongs to. The format of this field is the context type, followed by an underscore, followed by the context id. For example: course_42 Including a code matching the user's own context code (e.g. user_1) will include notes that are not associated with any particular course.

Example Response:

```
[
  {
    'id': 4,
    'title': 'Bring bio book',
    'description': 'bring bio book for friend tomorrow',
    'user_id': 1238,
    'course_id': 4567, // If the user assigns a note to a course
    'todo_date': "2017-05-09T10:12:00Z",
    'workflow_state': 'active',
  },
  {
    'id': 5,
    'title': 'Bring english book',
    'description': 'bring english book to class tomorrow',
    'user_id': 1234,
    'todo_date': "2017-05-09T10:12:00Z",
    'workflow_state': 'active',
  },
]
```

Returns a list of [PlannerNote](#) objects

Show a planner note [PlannerNotesController#show](#)

GET /api/v1/planner\_notes/:id

Scope: url:GET|/api/v1/planner\_notes/:id

Retrieve a planner note for the current user

Returns a [PlannerNote](#) object

Update a planner note [PlannerNotesController#update](#)

PUT /api/v1/planner\_notes/:id

Scope: url:PUT|/api/v1/planner\_notes/:id

Update a planner note for the current user

Request Parameters:

Parameter	Type	Description
title	string	The title of the planner note.
details	string	Text of the planner note.
todo_date	Date	The date where this planner note should appear in the planner. The value should be formatted as: yyyy-mm-dd.
course_id	integer	The ID of the course to associate with the planner note. The caller must be able to view the course in order to associate it with a planner note. Use a null or empty value to remove a planner note from a course. Note that if the planner note is linked to a learning object, its course_id cannot be changed.

Returns a [PlannerNote](#) object

Create a planner note [PlannerNotesController#create](#)

POST /api/v1/planner\_notes

Scope: url:POST|/api/v1/planner\_notes

Create a planner note for the current user

Request Parameters:

Parameter	Type	Description
title	string	The title of the planner note.
details	string	Text of the planner note.
todo_date	Date	The date where this planner note should appear in the planner. The value should be formatted as: yyyy-mm-dd.
course_id	integer	The ID of the course to associate with the planner note. The caller must be able to view the course in order to associate it with a planner note.
linked_object_type	string	The type of a learning object to link to this planner note. Must be used in conjunction with linked_object_id and course_id. Valid linked_object_type values are: 'announcement', 'assignment', 'discussion_topic', 'wiki_page', 'quiz'
linked_object_id	integer	The id of a learning object to link to this planner note. Must be used in conjunction with linked_object_type and course_id. The object must be in the same course as specified by course_id. If the title argument is not provided, the planner note will use the learning object's title as its title. Only one planner note may be linked to a specific learning object.

Returns a [PlannerNote](#) object

Delete a planner note [PlannerNotesController#destroy](#)

DELETE /api/v1/planner\_notes/:id

Scope: url:DELETE|/api/v1/planner\_notes/:id

Delete a planner note for the current user

Returns a [PlannerNote](#) object

List planner overrides [PlannerOverridesController#index](#)

GET /api/v1/planner/overrides

Scope: url:GET|/api/v1/planner/overrides

Retrieve a planner override for the current user

Returns a list of [PlannerOverride](#) objects

Show a planner override [PlannerOverridesController#show](#)

GET /api/v1/planner/overrides/:id

Scope: url:GET|/api/v1/planner/overrides/:id

Retrieve a planner override for the current user

Returns a [PlannerOverride](#) object

Update a planner override [PlannerOverridesController#update](#)

PUT /api/v1/planner/overrides/:id

Scope: url:PUT|/api/v1/planner/overrides/:id

Update a planner override's visibilty for the current user

Request Parameters:

Parameter	Type	Description
marked_complete	string	determines whether the planner item is marked as completed
dismissed	string	determines whether the planner item shows in the opportunities list

Returns a [PlannerOverride](#) object

Create a planner override [PlannerOverridesController#create](#)

POST /api/v1/planner/overrides

Scope: url:POST|/api/v1/planner/overrides

Create a planner override for the current user

Request Parameters:

Parameter	Type	Description
plannable_type	Required string	Type of the item that you are overriding in the planner Allowed values: announcement, assignment, discussion_topic, quiz, wiki_page, planner_note, calendar_event, assessment_request, sub_assignment
plannable_id	Required integer	ID of the item that you are overriding in the planner
marked_complete	boolean	If this is true, the item will show in the planner as completed
dismissed	boolean	If this is true, the item will not show in the opportunities list

Returns a [PlannerOverride](#) object

Delete a planner override [PlannerOverridesController#destroy](#)

DELETE /api/v1/planner/overrides/:id

Scope: url:DELETE|/api/v1/planner/overrides/:id

Delete a planner override for the current user

Returns a [PlannerOverride](#) object

List poll sessions for a poll [Polling::PollSessionsController#index](#)

GET /api/v1/polls/:poll\_id/poll\_sessions

Scope: url:GET|/api/v1/polls/:poll\_id/poll\_sessions

Returns the paginated list of PollSessions in this poll.

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

Get the results for a single poll session [Polling::PollSessionsController#show](#)

GET /api/v1/polls/:poll\_id/poll\_sessions/:id

Scope: url:GET|/api/v1/polls/:poll\_id/poll\_sessions/:id

Returns the poll session with the given id

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

Create a single poll session [Polling::PollSessionsController#create](#)

POST /api/v1/polls/:poll\_id/poll\_sessions

Scope: url:POST|/api/v1/polls/:poll\_id/poll\_sessions

Create a new poll session for this poll

Request Parameters:

Parameter	Type	Description
poll_sessions[][course_id]	Required integer	The id of the course this session is associated with.
poll_sessions[][course_section_id]	integer	The id of the course section this session is associated with.
poll_sessions[][has_public_results]	boolean	Whether or not results are viewable by students.

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

Update a single poll session [Polling::PollSessionsController#update](#)

PUT /api/v1/polls/:poll\_id/poll\_sessions/:id

Scope: url:PUT|/api/v1/polls/:poll\_id/poll\_sessions/:id

Update an existing poll session for this poll

Request Parameters:

Parameter	Type	Description
poll_sessions[][course_id]	integer	The id of the course this session is associated with.
poll_sessions[][course_section_id]	integer	The id of the course section this session is associated with.
poll_sessions[][has_public_results]	boolean	Whether or not results are viewable by students.

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

Delete a poll session [Polling::PollSessionsController#destroy](#)

**DELETE** /api/v1/polls/:poll\_id/poll\_sessions/:id

**Scope:** url:DELETE|/api/v1/polls/:poll\_id/poll\_sessions/:id

204 No Content response code is returned if the deletion was successful.

Open a poll session [Polling::PollSessionsController#open](#)

**GET** /api/v1/polls/:poll\_id/poll\_sessions/:id/open

**Scope:** url:GET|/api/v1/polls/:poll\_id/poll\_sessions/:id/open

Close an opened poll session [Polling::PollSessionsController#close](#)

**GET** /api/v1/polls/:poll\_id/poll\_sessions/:id/close

**Scope:** url:GET|/api/v1/polls/:poll\_id/poll\_sessions/:id/close

List opened poll sessions [Polling::PollSessionsController#opened](#)

**GET** /api/v1/poll\_sessions/opened

**Scope:** url:GET|/api/v1/poll\_sessions/opened

A paginated list of all opened poll sessions available to the current user.

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

List closed poll sessions [Polling::PollSessionsController#closed](#)

**GET** /api/v1/poll\_sessions/closed

**Scope:** url:GET|/api/v1/poll\_sessions/closed

A paginated list of all closed poll sessions available to the current user.

Example Response:

```
{
  "poll_sessions": [PollSession]
}
```

List poll choices in a poll [Polling::PollChoicesController#index](#)

**GET** /api/v1/polls/:poll\_id/poll\_choices

**Scope:** url:GET|/api/v1/polls/:poll\_id/poll\_choices

Returns the paginated list of PollChoices in this poll.

Example Response:

```
{
  "poll_choices": [PollChoice]
}
```

Get a single poll choice [Polling::PollChoicesController#show](#)

**GET** /api/v1/polls/:poll\_id/poll\_choices/:id

**Scope:** url:GET|/api/v1/polls/:poll\_id/poll\_choices/:id

Returns the poll choice with the given id

Example Response:

```
{
  "poll_choices": [PollChoice]
}
```

Create a single poll choice [Polling::PollChoicesController#create](#)

**POST** /api/v1/polls/:poll\_id/poll\_choices

**Scope:** url:POST|/api/v1/polls/:poll\_id/poll\_choices

Create a new poll choice for this poll

Request Parameters:

Parameter	Type	Description
poll_choices[][text]	Required string	The descriptive text of the poll choice.
poll_choices[][is_correct]	boolean	Whether this poll choice is considered correct or not. Defaults to false.
poll_choices[][position]	integer	The order this poll choice should be returned in the context it's sibling poll choices.

Example Response:

```
{
  "poll_choices": [PollChoice]
}
```

Update a single poll choice [Polling::PollChoicesController#update](#)

**PUT** /api/v1/polls/:poll\_id/poll\_choices/:id

**Scope:** url:PUT|/api/v1/polls/:poll\_id/poll\_choices/:id

Update an existing poll choice for this poll

Request Parameters:

Parameter	Type	Description
poll_choices[][text]	Required string	The descriptive text of the poll choice.
poll_choices[][is_correct]	boolean	Whether this poll choice is considered correct or not. Defaults to false.
poll_choices[][position]	integer	The order this poll choice should be returned in the context it's sibling poll choices.

Example Response:

```
{
  "poll_choices": [PollChoice]
}
```

[Delete a poll choice Polling::PollChoicesController#destroy](#)

**DELETE** /api/v1/polls/:poll\_id/poll\_choices/:id

**Scope:** url:DELETE//api/v1/polls/:poll\_id/poll\_choices/:id

**204 No Content** response code is returned if the deletion was successful.

[List polls Polling::PollsController#index](#)

**GET** /api/v1/polls

**Scope:** url:GET//api/v1/polls

Returns the paginated list of polls for the current user.

**Example Response:**

```
{
  "polls": [Poll]
}
```

[Get a single poll Polling::PollsController#show](#)

**GET** /api/v1/polls/:id

**Scope:** url:GET//api/v1/polls/:id

Returns the poll with the given id

**Example Response:**

```
{
  "polls": [Poll]
}
```

[Create a single poll Polling::PollsController#create](#)

**POST** /api/v1/polls

**Scope:** url:POST//api/v1/polls

Create a new poll for the current user

**Request Parameters:**

Parameter	Type	Description
polls[][question]	Required string	The title of the poll.
polls[][description]	string	A brief description or instructions for the poll.

**Example Response:**

```
{
  "polls": [Poll]
}
```

[Update a single poll Polling::PollsController#update](#)

**PUT** /api/v1/polls/:id

**Scope:** url:PUT//api/v1/polls/:id

Update an existing poll belonging to the current user

**Request Parameters:**

Parameter	Type	Description
polls[][question]	Required string	The title of the poll.
polls[][description]	string	A brief description or instructions for the poll.

**Example Response:**

```
{
  "polls": [Poll]
}
```

[Delete a poll Polling::PollsController#destroy](#)

**DELETE** /api/v1/polls/:id

**Scope:** url:DELETE//api/v1/polls/:id

**204 No Content** response code is returned if the deletion was successful.

[Get a single poll submission Polling::PollSubmissionsController#show](#)

**GET** /api/v1/polls/:poll\_id/poll\_sessions/:poll\_session\_id/poll\_submissions/:id

**Scope:** url:GET//api/v1/polls/:poll\_id/poll\_sessions/:poll\_session\_id/poll\_submissions/:id

Returns the poll submission with the given id

**Example Response:**

```
{
  "poll_submissions": [PollSubmission]
}
```

[Create a single poll submission Polling::PollSubmissionsController#create](#)

**POST** /api/v1/polls/:poll\_id/poll\_sessions/:poll\_session\_id/poll\_submissions

**Scope:** url:POST//api/v1/polls/:poll\_id/poll\_sessions/:poll\_session\_id/poll\_submissions

Create a new poll submission for this poll session

**Request Parameters:**

Parameter	Type	Description
poll_submissions[][poll_choice_id]	Required integer	The chosen poll choice for this submission.

**Example Response:**

```
{
  "poll_submissions": [PollSubmission]
}
```

[Create/update proficiency ratings OutcomeProficiencyApiController#create](#)

POST /api/v1/accounts/:account\_id/outcome\_proficiency

Scope: url:POST /api/v1/accounts/:account\_id/outcome\_proficiency

POST /api/v1/courses/:course\_id/outcome\_proficiency

Scope: url:POST /api/v1/courses/:course\_id/outcome\_proficiency

Create or update account-level proficiency ratings. These ratings will apply to all sub-accounts, unless they have their own account-level proficiency ratings defined.

Request Parameters:

Parameter	Type	Description
ratings[][description]	string	The description of the rating level.
ratings[][points]	integer	The non-negative number of points of the rating level. Points across ratings should be strictly decreasing in value.
ratings[][mastery]	integer	Indicates the rating level where mastery is first achieved. Only one rating in a proficiency should be marked for mastery.
ratings[][color]	integer	The color associated with the rating level. Should be a hex color code like '00FFFF'.

Example Request:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/outcome_proficiency' \  
-X POST \  
-F 'ratings[][description]=Exceeds Mastery' \  
-F 'ratings[][points]=4' \  
-F 'ratings[][color]=127A1B' \  
-F 'ratings[][mastery]=false' \  
-F 'ratings[][description]=Mastery' \  
-F 'ratings[][points]=3' \  
-F 'ratings[][color]=08074B' \  
-F 'ratings[][mastery]=true' \  
-F 'ratings[][description]=Near Mastery' \  
-F 'ratings[][points]=2' \  
-F 'ratings[][color]=FAB901' \  
-F 'ratings[][mastery]=false' \  
-F 'ratings[][description]=Below Mastery' \  
-F 'ratings[][points]=1' \  
-F 'ratings[][color]=F05010' \  
-F 'ratings[][mastery]=false' \  
-F 'ratings[][description]=Well Below Mastery' \  
-F 'ratings[][points]=0' \  
-F 'ratings[][color]=000000' \  
-F 'ratings[][mastery]=false' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Proficiency](#) object

[Get proficiency ratings OutcomeProficiencyApiController#show](#)

GET /api/v1/accounts/:account\_id/outcome\_proficiency

Scope: url:GET /api/v1/accounts/:account\_id/outcome\_proficiency

GET /api/v1/courses/:course\_id/outcome\_proficiency

Scope: url:GET /api/v1/courses/:course\_id/outcome\_proficiency

Get account-level proficiency ratings. If not defined for this account, it will return proficiency ratings for the nearest super-account with ratings defined. Will return 404 if none found.

```
Examples:   
curl https://<canvas>/api/v1/accounts/<account_id>/outcome_proficiency \  
-H 'Authorization: Bearer <token>'
```

Returns a [Proficiency](#) object

[Query progress ProgressController#show](#)

GET /api/v1/progress/:id

Scope: url:GET /api/v1/progress/:id

Return completion and status information about an asynchronous job

Returns a [Progress](#) object

[Cancel progress ProgressController#cancel](#)

POST /api/v1/progress/:id/cancel

Scope: url:POST /api/v1/progress/:id/cancel

Cancel an asynchronous job associated with a Progress object If you include “message” in the POSTed data, it will be set on the Progress and returned. This is handy to distinguish between cancel and fail for a workflow\_state of “failed”.

Returns a [Progress](#) object

[Query progress Lti::Lms::ProgressController#show](#)

GET /api/lti/courses/:course\_id/progress/:id

Scope: url:GET /api/lti/courses/:course\_id/progress/:id

Return completion and status information about an asynchronous job

Returns a [Progress](#) object

[Update Public JWK Lti::PublicJwkController#update](#)

PUT /api/lti/developer\_key/update\_public\_jwk

Scope: url:PUT /api/lti/developer\_key/update\_public\_jwk

Rotate the public key in jwk format when using Lti services

Request Parameters:

Parameter	Type	Description
public_jwk	Required json	The new public jwk that will be set to the tools current public jwk.

Returns a [DeveloperKey](#) object

[Retrieve assignment-overridden dates for Classic Quizzes Quizzes::QuizAssignmentOverridesController#index](#)

GET /api/v1/courses/:course\_id/quizzes/assignment\_overrides

Scope: url:GET /api/v1/courses/:course\_id/quizzes/assignment\_overrides

Retrieve the actual due-at, unlock-at, and available-at dates for quizzes based on the assignment overrides active for the current API user.

Request Parameters:

Parameter	Type	Description
quiz_assignment_overrides[][quiz_ids][]	integer	An array of quiz IDs. If omitted, overrides for all quizzes available to the operating user will be returned.

Example Response:

```
{   
  "quiz_assignment_overrides": [{   
    "quiz_id": "1",
```

```
      "due_dates": [QuizAssignmentOverride],
      "all_dates": [QuizAssignmentOverride]
    }, {
      "quiz_id": "2",
      "due_dates": [QuizAssignmentOverride],
      "all_dates": [QuizAssignmentOverride]
    }
  ]
}
```

Returns a [QuizAssignmentOverrideSetContainer](#) object

**[Retrieve assignment-overridden dates for New Quizzes Quizzes::QuizAssignmentOverridesController#new\\_quizzes](#)**

**GET /api/v1/courses/:course\_id/new\_quizzes/assignment\_overrides**

**Scope:** url:GET|/api/v1/courses/:course\_id/new\_quizzes/assignment\_overrides

Retrieve the actual due-at, unlock-at, and available-at dates for quizzes based on the assignment overrides active for the current API user.

**Request Parameters:**

Parameter	Type	Description
quiz_assignment_overrides[][quiz_ids][]	integer	An array of quiz IDs. If omitted, overrides for all quizzes available to the operating user will be returned.

**Example Response:**

```
{
  "quiz_assignment_overrides": [
    {
      "quiz_id": "1",
      "due_dates": [QuizAssignmentOverride],
      "all_dates": [QuizAssignmentOverride]
    }, {
      "quiz_id": "2",
      "due_dates": [QuizAssignmentOverride],
      "all_dates": [QuizAssignmentOverride]
    }
  ]
}
```

Returns a [QuizAssignmentOverrideSetContainer](#) object

**[Set extensions for student quiz submissions Quizzes::QuizExtensionsController#create](#)**

**POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/extensions**

**Scope:** url:POST|/api/v1/courses/:course\_id/quizzes/:quiz\_id/extensions

**Responses**

- **200 OK** if the request was successful
- **403 Forbidden** if you are not allowed to extend quizzes for this course

**Request Parameters:**

Parameter	Type	Description
quiz_extensions[][user_id]	Required integer	The ID of the user we want to add quiz extensions for.
quiz_extensions[][extra_attempts]	integer	Number of times the student is allowed to re-take the quiz over the multiple-attempt limit. This is limited to 1000 attempts or less.
quiz_extensions[][extra_time]	integer	The number of extra minutes to allow for all attempts. This will add to the existing time limit on the submission. This is limited to 10080 minutes (1 week)
quiz_extensions[][manually_unlocked]	boolean	Allow the student to take the quiz even if it's locked for everyone else.
quiz_extensions[][extend_from_now]	integer	The number of minutes to extend the quiz from the current time. This is mutually exclusive to extend_from_end_at. This is limited to 1440 minutes (24 hours)
quiz_extensions[][extend_from_end_at]	integer	The number of minutes to extend the quiz beyond the quiz's current ending time. This is mutually exclusive to extend_from_now. This is limited to 1440 minutes (24 hours)

**Example Request:**

```
{
  "quiz_extensions": [
    {
      "user_id": 3,
      "extra_attempts": 2,
      "extra_time": 20,
      "manually_unlocked": true
    }, {
      "user_id": 2,
      "extra_attempts": 2,
      "extra_time": 20,
      "manually_unlocked": false
    }
  ]
}
```

```
{
  "quiz_extensions": [
    {
      "user_id": 3,
      "extend_from_now": 20
    }
  ]
}
```

**Example Response:**

```
{
  "quiz_extensions": [QuizExtension]
}
```

**[Get available quiz IP filters. Quizzes::QuizIpFiltersController#index](#)**

**GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/ip\_filters**

**Scope:** url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/ip\_filters

Get a list of available IP filters for this Quiz.

**200 OK** response code is returned if the request was successful.

**Example Response:**

```
{
  "quiz_ip_filters": [QuizIPFilter]
}
```

**[Get a single quiz group Quizzes::QuizGroupsController#show](#)**

**GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id**

**Scope:** url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id

Returns details of the quiz group with the given id.

Returns a [QuizGroup](#) object

**[Create a question group Quizzes::QuizGroupsController#create](#)**

**POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/groups**

**Scope:** url:POST|/api/v1/courses/:course\_id/quizzes/:quiz\_id/groups

Create a new question group for this quiz

**201 Created** response code is returned if the creation was successful.

**Request Parameters:**

Parameter	Type	Description
-----------	------	-------------



Parameter	Type	Description
quiz_groups[][name]	string	The name of the question group.
quiz_groups[][pick_count]	integer	The number of questions to randomly select for this group.
quiz_groups[][question_points]	integer	The number of points to assign to each question in the group.
quiz_groups[][assessment_question_bank_id]	integer	The id of the assessment question bank to pull questions from.

Example Response:

```
{
  "quiz_groups": [QuizGroup]
}
```

Update a question group Quizzes::QuizGroupsController#update

PUT /api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id

Scope: url:PUT|/api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id

Update a question group

Request Parameters:

Parameter	Type	Description
quiz_groups[][name]	string	The name of the question group.
quiz_groups[][pick_count]	integer	The number of questions to randomly select for this group.
quiz_groups[][question_points]	integer	The number of points to assign to each question in the group.

Example Response:

```
{
  "quiz_groups": [QuizGroup]
}
```

Delete a question group Quizzes::QuizGroupsController#destroy

DELETE /api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id

Delete a question group

<b>204 No Content<b> response code is returned if the deletion was successful.

Reorder question groups Quizzes::QuizGroupsController#reorder

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id/reorder

Scope: url:POST|/api/v1/courses/:course\_id/quizzes/:quiz\_id/groups/:id/reorder

Change the order of the quiz questions within the group

<b>204 No Content<b> response code is returned if the reorder was successful.

Request Parameters:

Parameter	Type	Description
order[][id]	Required integer	The associated item's unique identifier
order[][type]	string	The type of item is always 'question' for a group Allowed values: question

List questions in a quiz or a submission Quizzes::QuizQuestionsController#index

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/questions

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/questions

Returns the paginated list of QuizQuestions in this quiz.

Request Parameters:

Parameter	Type	Description
quiz_submission_id	integer	If specified, the endpoint will return the questions that were presented for that submission. This is useful if the quiz has been modified after the submission was created and the latest quiz version's set of questions does not match the submission's. NOTE: you must specify quiz_submission_attempt as well if you specify this parameter.
quiz_submission_attempt	integer	The attempt of the submission you want the questions for.

Returns a list of [QuizQuestion](#) objects

Get a single quiz question Quizzes::QuizQuestionsController#show

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id

Returns the quiz question with the given id

Request Parameters:

Parameter	Type	Description
id	Required integer	The quiz question unique identifier.

Returns a [QuizQuestion](#) object

Create a single quiz question Quizzes::QuizQuestionsController#create

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/questions

Scope: url:POST|/api/v1/courses/:course\_id/quizzes/:quiz\_id/questions

Create a new quiz question for this quiz

Request Parameters:

Parameter	Type	Description
question[question_name]	string	The name of the question.

Parameter	Type	Description
question[question_text]	string	The text of the question.
question[quiz_group_id]	integer	The id of the quiz group to assign the question to.
question[question_type]	string	The type of question. Multiple optional fields depend upon the type of question to be used. Allowed values: calculated_question, essay_question, file_upload_question, fill_in_multiple_blanks_question, matching_question, multiple_answers_question, multiple_choice_question, multiple_dropdowns_question, numerical_question, short_answer_question, text_only_question, true_false_question
question[position]	integer	The order in which the question will be displayed in the quiz in relation to other questions.
question[points_possible]	integer	The maximum amount of points received for answering this question correctly.
question[correct_comments]	string	The comment to display if the student answers the question correctly.
question[incorrect_comments]	string	The comment to display if the student answers incorrectly.
question[neutral_comments]	string	The comment to display regardless of how the student answered.
question[text_after_answers]	string	no description
question[answers]	[Answer]	no description

Returns a [QuizQuestion](#) object

**[Update an existing quiz question Quizzes::QuizQuestionsController#update](#)**

**PUT /api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id**

**Scope:** url:PUT[/api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id]

Updates an existing quiz question for this quiz

**Request Parameters:**

Parameter	Type	Description
quiz_id	Required integer	The associated quiz's unique identifier.
id	Required integer	The quiz question's unique identifier.
question[question_name]	string	The name of the question.
question[question_text]	string	The text of the question.
question[quiz_group_id]	integer	The id of the quiz group to assign the question to.
question[question_type]	string	The type of question. Multiple optional fields depend upon the type of question to be used. Allowed values: calculated_question, essay_question, file_upload_question, fill_in_multiple_blanks_question, matching_question, multiple_answers_question, multiple_choice_question, multiple_dropdowns_question, numerical_question, short_answer_question, text_only_question, true_false_question
question[position]	integer	The order in which the question will be displayed in the quiz in relation to other questions.
question[points_possible]	integer	The maximum amount of points received for answering this question correctly.
question[correct_comments]	string	The comment to display if the student answers the question correctly.
question[incorrect_comments]	string	The comment to display if the student answers incorrectly.
question[neutral_comments]	string	The comment to display regardless of how the student answered.
question[text_after_answers]	string	no description
question[answers]	[Answer]	no description

Returns a [QuizQuestion](#) object

**[Delete a quiz question Quizzes::QuizQuestionsController#destroy](#)**

**DELETE /api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id**

**Scope:** url:DELETE[/api/v1/courses/:course\_id/quizzes/:quiz\_id/questions/:id]

**204 No Content** response code is returned if the deletion was successful.

**Request Parameters:**

Parameter	Type	Description
quiz_id	Required integer	The associated quiz's unique identifier
id	Required integer	The quiz question's unique identifier

**[Retrieve all quiz reports Quizzes::QuizReportsController#index](#)**

**GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/reports**

**Scope:** url:GET[/api/v1/courses/:course\_id/quizzes/:quiz\_id/reports]

Returns a list of all available reports.

**Request Parameters:**

Parameter	Type	Description
includes_all_versions	boolean	Whether to retrieve reports that consider all the submissions or only the most recent. Defaults to false, ignored for item_analysis reports.

Returns a list of [QuizReport](#) objects

**[Create a quiz report Quizzes::QuizReportsController#create](#)**

**POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/reports**

**Scope:** url:POST[/api/v1/courses/:course\_id/quizzes/:quiz\_id/reports]

Create and return a new report for this quiz. If a previously generated report matches the arguments and is still current (i.e. there have been no new submissions), it will be returned.

Responses

- 400 Bad Request if the specified report type is invalid
- 409 Conflict if a quiz report of the specified type is already being generated

Request Parameters:

Parameter	Type	Description
quiz_report[report_type]	Required string	The type of report to be generated. Allowed values: student_analysis, item_analysis
quiz_report[includes_all_versions]	boolean	Whether the report should consider all submissions or only the most recent. Defaults to false, ignored for item_analysis.
include	String[]	Whether the output should include documents for the file and/or progress objects associated with this report. (Note: JSON-API only) Allowed values: file, progress

Returns a [QuizReport](#) object

[Get a quiz report Quizzes::QuizReportsController#show](#)

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/reports/:id

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/reports/:id

Returns the data for a single quiz report.

Request Parameters:

Parameter	Type	Description
include	String[]	Whether the output should include documents for the file and/or progress objects associated with this report. (Note: JSON-API only) Allowed values: file, progress

Returns a [QuizReport](#) object

[Abort the generation of a report, or remove a previously generated one Quizzes::QuizReportsController#abort](#)

DELETE /api/v1/courses/:course\_id/quizzes/:quiz\_id/reports/:id

Scope: url:DELETE|/api/v1/courses/:course\_id/quizzes/:quiz\_id/reports/:id

This API allows you to cancel a previous request you issued for a report to be generated. Or in the case of an already generated report, you'd like to remove it, perhaps to generate it another time with an updated version that provides new features.

You must check the report's generation status before attempting to use this interface. See the "workflow\_state" property of the QuizReport's Progress object for more information. Only when the progress reports itself in a "queued" state can the generation be aborted.

Responses

- 204 No Content if your request was accepted
- 422 Unprocessable Entity if the report is not being generated or can not be aborted at this stage

[Fetching the latest quiz statistics Quizzes::QuizStatisticsController#index](#)

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/statistics

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/statistics

This endpoint provides statistics for all quiz versions, or for a specific quiz version, in which case the output is guaranteed to represent the *latest* and most current version of the quiz.

200 OK response code is returned if the request was successful.

Request Parameters:

Parameter	Type	Description
all_versions	boolean	Whether the statistics report should include all submissions attempts.

Example Response:

```
{
  "quiz_statistics": [ QuizStatistics ]
}
```

[Submit captured events Quizzes::QuizSubmissionEventsApiController#create](#)

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/events

Scope: url:POST|/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/events

Store a set of events which were captured during a quiz taking session.

On success, the response will be 204 No Content with an empty body.

Request Parameters:

Parameter	Type	Description
quiz_submission_events[]	Required Array	The submission events to be recorded

Example Request:

```
{
  "quiz_submission_events":
  [
    {
      "client_timestamp": "2014-10-08T19:29:58Z",
      "event_type": "question_answered",
      "event_data": { "answer": "42" }
    }, {
      "client_timestamp": "2014-10-08T19:30:17Z",
      "event_type": "question_flagged",
      "event_data": { "question_id": "1", "flagged": true }
    }
  ]
}
```

[Retrieve captured events Quizzes::QuizSubmissionEventsApiController#index](#)

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/events

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/events

Retrieve the set of events captured during a specific submission attempt.

Request Parameters:

Parameter	Type	Description
attempt	integer	The specific submission attempt to look up the events for. If unspecified, the latest attempt will be used.

Example Response:

```
{
  "quiz_submission_events": [
```

```
{
  {
    "id": "3409",
    "event_type": "page_blurred",
    "event_data": null,
    "created_at": "2014-11-16T13:37:21Z"
  },
  {
    "id": "3410",
    "event_type": "page_focused",
    "event_data": null,
    "created_at": "2014-11-16T13:37:27Z"
  }
}
```

Upload a file Quizzes::QuizSubmissionFilesController#create

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/self/files

Scope: url:POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/self/files

Associate a new quiz submission file

This API endpoint is the first step in uploading a quiz submission file. See the [File Upload Documentation](#) for details on the file upload workflow as these parameters are interpreted as per the documentation there.

Request Parameters:

Parameter	Type	Description
name	string	The name of the quiz submission file
on_duplicate	string	How to handle duplicate names

Example Response:

```
{
  "attachments": [
    {
      "upload_url": "https://some-bucket.s3.amazonaws.com/",
      "upload_params": {
        "key": "/users/1234/files/answer_pic.jpg",
        "acl": "private",
        "Filename": "answer_pic.jpg",
        "AWSAccessKeyId": "some_id",
        "Policy": "some_opaque_string",
        "Signature": "another_opaque_string",
        "Content-Type": "image/jpeg"
      }
    }
  ]
}
```

Get all quiz submission questions. Quizzes::QuizSubmissionQuestionsController#index

GET /api/v1/quiz\_submissions/:quiz\_submission\_id/questions

Scope: url:GET /api/v1/quiz\_submissions/:quiz\_submission\_id/questions

Get a list of all the question records for this quiz submission.

200 OK response code is returned if the request was successful.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the quiz submission question. Allowed values: quiz_question

Example Response:

```
{
  "quiz_submission_questions": [QuizSubmissionQuestion]
}
```

Answering questions Quizzes::QuizSubmissionQuestionsController#answer

POST /api/v1/quiz\_submissions/:quiz\_submission\_id/questions

Scope: url:POST /api/v1/quiz\_submissions/:quiz\_submission\_id/questions

Provide or update an answer to one or more QuizQuestions.

Request Parameters:

Parameter	Type	Description
attempt	Required integer	The attempt number of the quiz submission being taken. Note that this must be the latest attempt index, as questions for earlier attempts can not be modified.
validation_token	Required string	The unique validation token you received when the Quiz Submission was created.
access_code	string	Access code for the Quiz, if any.
quiz_questions[]	QuizSubmissionQuestion	Set of question IDs and the answer value. See <a href="#">Appendix: Question Answer Formats</a> for the accepted answer formats for each question type.

Example Request:

```
{
  "attempt": 1,
  "validation_token": "YOUR_VALIDATION_TOKEN",
  "access_code": null,
  "quiz_questions": [{
    "id": "1",
    "answer": "Hello World!"
  }, {
    "id": "2",
    "answer": 42.0
  }]
}
```

Returns a list of [QuizSubmissionQuestion](#) objects

Get a formatted student numerical answer. Quizzes::QuizSubmissionQuestionsController#formatted\_answer

GET /api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/formatted\_answer

Scope: url:GET /api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/formatted\_answer

Matches the intended behavior of the UI when a numerical answer is entered and returns the resulting formatted number

Request Parameters:

Parameter	Type	Description
answer	Required Numeric	no description

Example Response:

```
{
  "formatted_answer": 12.1234
}
```

Flagging a question. Quizzes::QuizSubmissionQuestionsController#flag

PUT /api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/flag

**Scope:** url:PUT[/api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/flag

Set a flag on a quiz question to indicate that you want to return to it later.

Request Parameters:

Parameter	Type	Description
attempt	Required integer	The attempt number of the quiz submission being taken. Note that this must be the latest attempt index, as questions for earlier attempts can not be modified.
validation_token	Required string	The unique validation token you received when the Quiz Submission was created.
access_code	string	Access code for the Quiz, if any.

Example Request:

```
{
  "attempt": 1,
  "validation_token": "YOUR_VALIDATION_TOKEN",
  "access_code": null
}
```

[Unflagging a question. Quizzes::QuizSubmissionQuestionsController#unflag](#)

PUT /api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/unflag

**Scope:** url:PUT[/api/v1/quiz\_submissions/:quiz\_submission\_id/questions/:id/unflag

Remove the flag that you previously set on a quiz question after you've returned to it.

Request Parameters:

Parameter	Type	Description
attempt	Required integer	The attempt number of the quiz submission being taken. Note that this must be the latest attempt index, as questions for earlier attempts can not be modified.
validation_token	Required string	The unique validation token you received when the Quiz Submission was created.
access_code	string	Access code for the Quiz, if any.

Example Request:

```
{
  "attempt": 1,
  "validation_token": "YOUR_VALIDATION_TOKEN",
  "access_code": null
}
```

[Send a message to unsubmitted or submitted users for the quiz Quizzes::QuizSubmissionUsersController#message](#)

POST /api/v1/courses/:course\_id/quizzes/:id/submission\_users/message

**Scope:** url:POST[/api/v1/courses/:course\_id/quizzes/:id/submission\_users/message

```
{
  "body": {
    "type": "string",
    "description": "message body of the conversation to be created",
    "example": "Please take the quiz."
  },
  "recipients": {
    "type": "string",
    "description": "Who to send the message to. May be either 'submitted' or 'unsubmitted'",
    "example": "submitted"
  },
  "subject": {
    "type": "string",
    "description": "Subject of the new Conversation created",
    "example": "ATTN: Quiz 101 Students"
  }
}
```

Request Parameters:

Parameter	Type	Description
conversations	QuizUserConversation	• Body and recipients to send the message to.

[Get all quiz submissions. Quizzes::QuizSubmissionsApiController#index](#)

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions

**Scope:** url:GET[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions

Get a list of all submissions for this quiz. Users who can view or manage grades for a course will have submissions from multiple users returned. A user who can only submit will have only their own submissions returned. When a user has an in-progress submission, only that submission is returned. When there isn't an in-progress quiz\_submission, all completed submissions, including previous attempts, are returned.

200 OK response code is returned if the request was successful.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the quiz submission. Allowed values: submission, quiz, user

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

[Get the quiz submission. Quizzes::QuizSubmissionsApiController#submission](#)

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/submission

**Scope:** url:GET[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submission

Get the submission for this quiz for the current user.

200 OK response code is returned if the request was successful.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the quiz submission. Allowed values: submission, quiz, user

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

Get a single quiz submission. Quizzes::QuizSubmissionsApiController#show

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id

Scope: url:GET[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id]

Get a single quiz submission.

200 OK response code is returned if the request was successful.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the quiz submission. Allowed values: submission, quiz, user

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

Create the quiz submission (start a quiz-taking session) Quizzes::QuizSubmissionsApiController#create

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions

Scope: url:POST[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions]

Start taking a Quiz by creating a QuizSubmission which you can use to answer questions and submit your answers.

Responses

- 200 OK if the request was successful
- 400 Bad Request if the quiz is locked
- 403 Forbidden if an invalid access code is specified
- 403 Forbidden if the Quiz's IP filter restriction does not pass
- 409 Conflict if a QuizSubmission already exists for this user and quiz

Request Parameters:

Parameter	Type	Description
access_code	string	Access code for the Quiz, if any.
preview	boolean	Whether this should be a preview QuizSubmission and not count towards the user's course record. Teachers only.

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

Update student question scores and comments. Quizzes::QuizSubmissionsApiController#update

PUT /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id

Scope: url:PUT[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id]

Update the amount of points a student has scored for questions they've answered, provide comments for the student about their answer(s), or simply fudge the total score by a specific amount of points.

Responses

- 200 OK if the request was successful
- 403 Forbidden if you are not a teacher in this course
- 400 Bad Request if the attempt parameter is missing or invalid
- 400 Bad Request if the specified QS attempt is not yet complete

Request Parameters:

Parameter	Type	Description
quiz_submissions[][attempt]	Required integer	The attempt number of the quiz submission that should be updated. This attempt MUST be already completed.
quiz_submissions[][fudge_points]	number	Amount of positive or negative points to fudge the total score by.
quiz_submissions[][questions]	Hash	A set of scores and comments for each question answered by the student. The keys are the question IDs, and the values are hashes of 'score' and 'comment' entries. See <a href="#">Appendix: Manual Scoring</a> for more on this parameter.

Example Request:

```
{
  "quiz_submissions": [{
    "attempt": 1,
    "fudge_points": -2.4,
    "questions": {
      "1": {
        "score": 2.5,
        "comment": "This can't be right, but I'll let it pass this one time."
      },
      "2": {
        "score": 0,
        "comment": "Good thinking. Almost!"
      }
    }
  }]
}
```

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

See Also:

- [Appendix: Manual Scoring](#)

Complete the quiz submission (turn it in). Quizzes::QuizSubmissionsApiController#complete

POST /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/complete

Scope: url:POST[/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/complete]

Complete the quiz submission by marking it as complete and grading it. When the quiz submission has been marked as complete, no further modifications will be allowed.

Responses

- 200 OK if the request was successful
- 403 Forbidden if an invalid access code is specified
- 403 Forbidden if the Quiz's IP filter restriction does not pass
- 403 Forbidden if an invalid token is specified

- **400 Bad Request** if the QS is already complete
- **400 Bad Request** if the attempt parameter is missing
- **400 Bad Request** if the attempt parameter is not the latest attempt

Request Parameters:

Parameter	Type	Description
attempt	Required integer	The attempt number of the quiz submission that should be completed. Note that this must be the latest attempt index, as earlier attempts can not be modified.
validation_token	Required string	The unique validation token you received when this Quiz Submission was created.
access_code	string	Access code for the Quiz, if any.

Example Response:

```
{
  "quiz_submissions": [QuizSubmission]
}
```

Get current quiz submission times. Quizzes::QuizSubmissionsApiController#time

GET /api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/id/time

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:quiz\_id/submissions/:id/time

Get the current timing data for the quiz attempt, both the end\_at timestamp and the time\_left parameter.

Responses

- **200 OK** if the request was successful

Example Response:

```
{
  "end_at": [DateTime],
  "time_left": [Integer]
}
```

List quizzes in a course Quizzes::QuizzesApiController#index

GET /api/v1/courses/:course\_id/quizzes

Scope: url:GET|/api/v1/courses/:course\_id/quizzes

Returns the paginated list of Quizzes in this course.

Request Parameters:

Parameter	Type	Description
search_term	string	The partial title of the quizzes to match and return.

Example Request:

```
curl https://<canvas>/api/v1/courses/<course_id>/quizzes \
-H 'Authorization: Bearer <token>'
```

Returns a list of Quiz objects

Get a single quiz Quizzes::QuizzesApiController#show

GET /api/v1/courses/:course\_id/quizzes/id

Scope: url:GET|/api/v1/courses/:course\_id/quizzes/:id

Returns the quiz with the given id.

Returns a Quiz object

Create a quiz Quizzes::QuizzesApiController#create

POST /api/v1/courses/:course\_id/quizzes

Scope: url:POST|/api/v1/courses/:course\_id/quizzes

Create a new quiz for this course.

Request Parameters:

Parameter	Type	Description
quiz[title]	Required string	The quiz title.
quiz[description]	string	A description of the quiz.
quiz[quiz_type]	string	The type of quiz. Allowed values: practice_quiz, assignment, graded_survey, survey
quiz[assignment_group_id]	integer	The assignment group id to put the assignment in. Defaults to the top assignment group in the course. Only valid if the quiz is graded, i.e. if quiz_type is "assignment" or "graded_survey".
quiz[time_limit]	integer	Time limit to take this quiz, in minutes. Set to null for no time limit. Defaults to null.
quiz[shuffle_answers]	boolean	If true, quiz answers for multiple choice questions will be randomized for each student. Defaults to false.
quiz[hide_results]	string	Dictates whether or not quiz results are hidden from students. If null, students can see their results after any attempt. If "always", students can never see their results. If "until_after_last_attempt", students can only see results after their last attempt. (Only valid if allowed_attempts > 1). Defaults to null. Allowed values: always, until_after_last_attempt
quiz[show_correct_answers]	boolean	Only valid if hide_results=null If false, hides correct answers from students when quiz results are viewed. Defaults to true.
quiz[show_correct_answers_last_attempt]	boolean	Only valid if show_correct_answers=true and allowed_attempts > 1 If true, hides correct answers from students when quiz results are viewed until they submit the last attempt for the quiz. Defaults to false.
quiz[show_correct_answers_at]	DateTime	Only valid if show_correct_answers=true If set, the correct answers will be visible by students only after this date, otherwise the correct answers are visible once the student hands in their quiz submission.
quiz[hide_correct_answers_at]	DateTime	Only valid if show_correct_answers=true If set, the correct answers will stop being visible once this date has passed. Otherwise, the correct answers will be visible indefinitely.
quiz[allowed_attempts]	integer	Number of times a student is allowed to take a quiz. Set to -1 for unlimited attempts. Defaults to 1.
quiz[scoring_policy]	string	Required and only valid if allowed_attempts > 1. Scoring policy for a quiz that students can take multiple times. Defaults to "keep_highest". Allowed values: keep_highest, keep_latest

Parameter	Type	Description
quiz[one_question_at_a_time]	boolean	If true, shows quiz to student one question at a time. Defaults to false.
quiz[cant_go_back]	boolean	Only valid if one_question_at_a_time=true If true, questions are locked after answering. Defaults to false.
quiz[access_code]	string	Restricts access to the quiz with a password. For no access code restriction, set to null. Defaults to null.  Restricts access to the quiz to computers in a specified IP range. Filters can be a comma-separated list of addresses, or an address followed by a mask  Examples: "192.168.217.1" "192.168.217.1/24" "192.168.217.1/255.255.255.0"  For no IP filter restriction, set to null. Defaults to null.
quiz[due_at]	DateTime	The day/time the quiz is due. Accepts times in ISO 8601 format, e.g. 2011-10-21T18:48Z.
quiz[lock_at]	DateTime	The day/time the quiz is locked for students. Accepts times in ISO 8601 format, e.g. 2011-10-21T18:48Z.
quiz[unlock_at]	DateTime	The day/time the quiz is unlocked for students. Accepts times in ISO 8601 format, e.g. 2011-10-21T18:48Z.
quiz[published]	boolean	Whether the quiz should have a draft state of published or unpublished. NOTE: If students have started taking the quiz, or there are any submissions for the quiz, you may not unpublish a quiz and will receive an error.
quiz[one_time_results]	boolean	Whether students should be prevented from viewing their quiz results past the first time (right after they turn the quiz in.) Only valid if "hide_results" is not set to "always". Defaults to false.
quiz[only_visible_to_overrides]	boolean	Whether this quiz is only visible to overrides (Only useful if 'differentiated assignments' account setting is on) Defaults to false.

Returns a [Quiz](#) object

[Edit a quiz Quizzes::QuizzesApiController#update](#)

**PUT /api/v1/courses/:course\_id/quizzes/:id**

**Scope:** url:PUT|/api/v1/courses/:course\_id/quizzes/:id

Modify an existing quiz. See the documentation for quiz creation.

Additional arguments:

**Request Parameters:**

Parameter	Type	Description
quiz[notify_of_update]	boolean	If true, notifies users that the quiz has changed. Defaults to true

Returns a [Quiz](#) object

[Delete a quiz Quizzes::QuizzesApiController#destroy](#)

**DELETE /api/v1/courses/:course\_id/quizzes/:id**

**Scope:** url:DELETE|/api/v1/courses/:course\_id/quizzes/:id

[Reorder quiz items Quizzes::QuizzesApiController#reorder](#)

**POST /api/v1/courses/:course\_id/quizzes/:id/reorder**

**Scope:** url:POST|/api/v1/courses/:course\_id/quizzes/:id/reorder

Change order of the quiz questions or groups within the quiz

**204 No Content** response code is returned if the reorder was successful.

**Request Parameters:**

Parameter	Type	Description
order[][id]	Required integer	The associated item's unique identifier
order[][type]	string	The type of item is either 'question' or 'group' Allowed values: question, group

[Validate quiz access code Quizzes::QuizzesApiController#validate\\_access\\_code](#)

**POST /api/v1/courses/:course\_id/quizzes/:id/validate\_access\_code**

**Scope:** url:POST|/api/v1/courses/:course\_id/quizzes/:id/validate\_access\_code

Accepts an access code and returns a boolean indicating whether that access code is correct

**Request Parameters:**

Parameter	Type	Description
access_code	Required string	The access code being validated

[Show a collection of Results Lti::Ims::ResultsController#index](#)

**GET /api/lti/courses/:course\_id/line\_items/:line\_item\_id/results**

**Scope:** url:GET|api/lti/courses/:course\_id/line\_items/:line\_item\_id/results

Show existing Results of a line item. Can be used to retrieve a specific student's result by adding the user\_id (defined as the lti\_user\_id or the Canvas user\_id) as a query parameter (i.e. user\_id=1000). If user\_id is included, it will return only one Result in the collection if the result exists, otherwise it will be empty. May also limit number of results by adding the limit query param (i.e. limit=100)

Returns a [Result](#) object

[Show a Result Lti::Ims::ResultsController#show](#)

**GET /api/lti/courses/:course\_id/line\_items/:line\_item\_id/results/:id**

**Scope:** url:GET|api/lti/courses/:course\_id/line\_items/:line\_item\_id/results/:id

Show existing Result of a line item.

Returns a [Result](#) object

[List roles RoleOverridesController#api\\_index](#)

**GET /api/v1/accounts/:account\_id/roles**

**Scope:** url:GET|api/v1/accounts/:account\_id/roles



A paginated list of the roles available to an account.

Request Parameters:

Parameter	Type	Description
account_id	Required string	The id of the account to retrieve roles for.
state[]	string	Filter by role state. If this argument is omitted, only 'active' roles are returned. Allowed values: active, inactive
show_inherited	boolean	If this argument is true, all roles inherited from parent accounts will be included.

Returns a list of [Role](#) objects

Get a single role [RoleOverridesController#show](#)

GET /api/v1/accounts/:account\_id/roles/:id

Scope: url:GET|/api/v1/accounts/:account\_id/roles/:id

Retrieve information about a single role

Request Parameters:

Parameter	Type	Description
account_id	Required string	The id of the account containing the role
role_id	Required integer	The unique identifier for the role
role	string	The name for the role

Returns a [Role](#) object

Create a new role [RoleOverridesController#add\\_role](#)

POST /api/v1/accounts/:account\_id/roles

Scope: url:POST|/api/v1/accounts/:account\_id/roles

Create a new course-level or account-level role.

Request Parameters:

Parameter	Type	Description
label	Required string	Label for the role.
role	string	Deprecatd alias for label.  Specifies the role type that will be used as a base for the permissions granted to this role.
base_role_type	string	Defaults to 'AccountMembership' if absent  Allowed values: AccountMembership, StudentEnrollment, TeacherEnrollment, TaEnrollment, ObserverEnrollment, DesignerEnrollment
permissions[<X>][explicit]	boolean	no description
permissions[<X>][enabled]	boolean	If explicit is 1 and enabled is 1, permission <X> will be explicitly granted to this role. If explicit is 1 and enabled has any other value (typically 0), permission <X> will be explicitly denied to this role. If explicit is any other value (typically 0) or absent, or if enabled is absent, the value for permission <X> will be inherited from upstream. Ignored if permission <X> is locked upstream (in an ancestor account).

May occur multiple times with unique values for <X>. Recognized permission names for <X> are:

```
[For Account-Level Roles Only]
become_user -- Users - act as
import_sis -- SIS Data - import
manage_account_memberships -- Admins - add / remove
manage_account_settings -- Account-level settings - manage
manage_alerts -- Global announcements - add / edit / delete
manage_catalog -- Catalog - manage
Manage Course Templates granular permissions
  add_course_template -- Course Templates - add
  delete_course_template -- Course Templates - delete
  edit_course_template -- Course Templates - edit
  manage_courses_add -- Courses - add
  manage_courses_admin -- Courses - manage / update
  manage_developer_keys -- Developer keys - manage
  manage_feature_flags -- Feature Options - enable / disable
  manage_master_courses -- Blueprint Courses - add / edit / associate / delete
  manage_role_overrides -- Permissions - manage
  manage_storage_quotas -- Storage Quotas - manage
  manage_sis -- SIS data - manage
Manage Temporary Enrollments granular permissions
  temporary_enrollments_add -- Temporary Enrollments - add
  temporary_enrollments_edit -- Temporary Enrollments - edit
  temporary_enrollments_delete -- Temporary Enrollments - delete
  manage_user_logins -- Users - manage login details
  manage_user_observers -- Users - manage observers
  moderate_user_content -- Users - moderate content
  read_course_content -- Course Content - view
  read_course_list -- Courses - view list
  view_course_changes -- Courses - view change logs
  view_feature_flags -- Feature Options - view
  view_grade_changes -- Grades - view change logs
  view_notifications -- Notifications - view
  view_quiz_answer_audits -- Quizzes - view submission log
  view_statistics -- Statistics - view
  undelete_courses -- Courses - undelete

[For both Account-Level and Course-level roles]
Note: Applicable enrollment types for course-level roles are given in brackets:
      S = student, T = teacher (instructor), A = TA, D = designer, O = observer.
      Lower-case letters indicate permissions that are off by default.
      A missing letter indicates the permission cannot be enabled for the role
      or any derived custom roles.
allow_course_admin_actions -- [ Tad ] Users - allow administrative actions in courses
create_collaborations -- [ STABO ] Student Collaborations - create
create_conferences -- [ STABO ] Web conferences - create
create_forum -- [ STABO ] Discussions - create
generate_observer_pairing_code -- [ tado ] Users - Generate observer pairing codes for students
import_outcomes -- [ TABO ] Learning Outcomes - import
manage_account_banks -- [ td ] Item Banks - manage account
share_banks_with_subaccounts -- [ tad ] Item Banks - share with subaccounts
manage_assignments -- [ TABO ] Assignments and Quizzes - add / edit / delete (deprecated)
Manage Assignments and Quizzes granular permissions
  manage_assignments_add -- [ TABO ] Assignments and Quizzes - add
  manage_assignments_edit -- [ TABO ] Assignments and Quizzes - edit / manage
  manage_assignments_delete -- [ TABO ] Assignments and Quizzes - delete
  manage_calendar -- [ STABO ] Course Calendar - add / edit / delete
  manage_content -- [ TABO ] Course Content - add / edit / delete
  manage_course_visibility -- [ TAD ] Course - change visibility
Manage Courses granular permissions
  manage_courses_conclude -- [ TAB ] Courses - conclude
  manage_courses_delete -- [ TAB ] Courses - delete
  manage_courses_publish -- [ TAB ] Courses - publish
  manage_courses_reset -- [ TAB ] Courses - reset
Manage Files granular permissions
  manage_files_add -- [ TABO ] Course Files - add
  manage_files_edit -- [ TABO ] Course Files - edit
  manage_files_delete -- [ TABO ] Course Files - delete
  manage_grades -- [ TA ] Grades - edit
Manage Groups granular permissions
  manage_groups_add -- [ TAD ] Groups - add
  manage_groups_delete -- [ TAD ] Groups - delete
  manage_groups_manage -- [ TAD ] Groups - manage
  manage_interaction_alerts -- [ Ta ] Alerts - add / edit / delete
  manage_outcomes -- [ STABO ] Learning Outcomes - add / edit / delete
  manage_proficiency_calculations -- [ t d ] Outcome Proficiency Calculations - add / edit / delete
  manage_proficiency_scales -- [ t d ] Outcome Proficiency/Mastery Scales - add / edit / delete
Manage Sections granular permissions
  manage_sections_add -- [ TAB ] Course Sections - add
  manage_sections_edit -- [ TAB ] Course Sections - edit
  manage_sections_delete -- [ TAB ] Course Sections - delete
  manage_students -- [ TAD ] Users - manage students in courses
  manage_rubrics -- [ TAD ] Rubrics - add / edit / delete
Manage Pages granular permissions
```

Parameter	Type	Description
	<div>manage_wiki_create -- [ TAdo ] Pages - create</div> <div>manage_wiki_delete -- [ TAdo ] Pages - delete</div> <div>manage_wiki_update -- [ TAdo ] Pages - update</div> <div>moderate_forum -- {sTAdo} Discussions - moderate</div> <div>post_to_forum -- {STAdo} Discussions - post</div> <div>read_announcements -- {STAdo} Announcements - view</div> <div>read_email_addresses -- {sTAdo} Users - view primary email address</div> <div>read_forum -- {STAdo} Discussions - view</div> <div>read_question_banks -- { TAdo } Question banks - view and link</div> <div>read_reports -- [ TAd ] Reports - manage</div> <div>read_roster -- {STAdo} Users - view list</div> <div>read_sis -- {sTA } SIS Data - read</div> <div>select_final_grade -- [ TA ] Grades - select final grade for moderation</div> <div>send_messages -- {STAdo} Conversations - send messages to individual course members</div> <div>send_messages_all -- {sTAdo} Conversations - send messages to entire class</div> <div>Users - Teacher granular permissions</div> <div>add_teacher_to_course -- [ TAd ] Add a teacher enrollment to a course</div> <div>remove_teacher_from_course -- [ TAd ] Remove a Teacher enrollment from a course</div> <div>Users - TA granular permissions</div> <div>add_ta_to_course -- [ TAd ] Add a TA enrollment to a course</div> <div>remove_ta_from_course -- [ TAd ] Remove a TA enrollment from a course</div> <div>Users - Designer granular permissions</div> <div>add_designer_to_course -- [ TAd ] Add a designer enrollment to a course</div> <div>remove_designer_from_course -- [ TAd ] Remove a designer enrollment from a course</div> <div>Users - Observer granular permissions</div> <div>add_observer_to_course -- [ TAd ] Add an observer enrollment to a course</div> <div>remove_observer_from_course -- [ TAd ] Remove an observer enrollment from a course</div> <div>Users - Student granular permissions</div> <div>add_student_to_course -- [ TAd ] Add a student enrollment to a course</div> <div>remove_student_from_course -- [ TAd ] Remove a student enrollment from a course</div> <div>view_all_grades -- [ TAd ] Grades - view all grades</div> <div>view_analytics -- {sTA } Analytics - view pages</div> <div>view_audit_trail -- [ t ] Grades - view audit trail</div> <div>view_group_pages -- {STAdo} Groups - view all student groups</div> <div>view_user_logins -- [ TA ] Users - view login IDs</div>	<div>Some of these permissions are applicable only for roles on the site admin account, on a root account, or for course-level roles with a particular base role type; if a specified permission is inapplicable, it will be ignored.</div> <div>Additional permissions may exist based on installed plugins.</div> <div>A comprehensive list of all permissions are available:</div> <div>Course Permissions PDF: <a href="https://bit.ly/cnvs-course-permissions">bit.ly/cnvs-course-permissions</a></div> <div>Account Permissions PDF: <a href="https://bit.ly/cnvs-acct-permissions">bit.ly/cnvs-acct-permissions</a></div>

permissions[<X>][locked]	boolean	If the value is 1, permission <X> will be locked downstream (new roles in subaccounts cannot override the setting). For any other value, permission <X> is left unlocked. Ignored if permission <X> is already locked upstream. May occur multiple times with unique values for <X>.
permissions[<X>][applies_to_self]	boolean	If the value is 1, permission <X> applies to the account this role is in. The default value is 1. Must be true if applies_to_descendants is false. This value is only returned if enabled is true.
permissions[<X>][applies_to_descendants]	boolean	If the value is 1, permission <X> cascades down to sub accounts of the account this role is in. The default value is 1. Must be true if applies_to_self is false.This value is only returned if enabled is true.

Example Request:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/roles.json' \  
-H 'Authorization: Bearer <token>' \  
-F 'label=New Role' \  
-F 'permissions[read_course_content][explicit]=1' \  
-F 'permissions[read_course_content][enabled]=1' \  
-F 'permissions[read_course_list][locked]=1' \  
-F 'permissions[read_question_banks][explicit]=1' \  
-F 'permissions[read_question_banks][enabled]=0' \  
-F 'permissions[read_question_banks][locked]=1'
```

Returns a [Role](#) object

Deactivate a role [RoleOverridesController#remove\\_role](#)

DELETE /api/v1/accounts/:account\_id/roles/:id

**Scope:** url:DELETE[/api/v1/accounts/:account\_id/roles/:id]

Deactivates a custom role. This hides it in the user interface and prevents it from being assigned to new users. Existing users assigned to the role will continue to function with the same permissions they had previously. Built-in roles cannot be deactivated.

Request Parameters:

Parameter	Type	Description
role_id	Required integer	The unique identifier for the role
role	string	The name for the role

Returns a [Role](#) object

Activate a role [RoleOverridesController#activate\\_role](#)

POST /api/v1/accounts/:account\_id/roles/:id/activate

**Scope:** url:POST[/api/v1/accounts/:account\_id/roles/:id/activate]

Re-activates an inactive role (allowing it to be assigned to new users)

Request Parameters:

Parameter	Type	Description
role_id	Required integer	The unique identifier for the role
role	Deprecated	The name for the role

Returns a [Role](#) object

Update a role [RoleOverridesController#update](#)

PUT /api/v1/accounts/:account\_id/roles/:id

**Scope:** url:PUT[/api/v1/accounts/:account\_id/roles/:id]

Update permissions for an existing role.

Recognized roles are:

- TeacherEnrollment
- StudentEnrollment
- TaEnrollment
- ObserverEnrollment
- DesignerEnrollment
- AccountAdmin
- Any previously created custom role

Request Parameters:

Parameter	Type	Description
label	string	The label for the role. Can only change the label of a custom role that belongs directly to the account.
permissions[<X>][explicit]	boolean	no description

Parameter	Type	Description
permissions[<X>][enabled]	boolean	These arguments are described in the documentation for the <a href="#">add_role_method</a> .
permissions[<X>][applies_to_self]	boolean	If the value is 1, permission <X> applies to the account this role is in. The default value is 1. Must be true if applies_to_descendants is false. This value is only returned if enabled is true.
permissions[<X>][applies_to_descendants]	boolean	If the value is 1, permission <X> cascades down to sub accounts of the account this role is in. The default value is 1. Must be true if applies_to_self is false.This value is only returned if enabled is true.

Example Request:

```
curl https://<canvas>/api/v1/accounts/:account_id/roles/2 \
-X PUT \
-H 'Authorization: Bearer <access_token>' \
-F 'label=New Role Name' \
-F 'permissions[manage_groups][explicit]=1' \
-F 'permissions[manage_groups][enabled]=1' \
-F 'permissions[manage_groups][locked]=1' \
-F 'permissions[send_messages][explicit]=1' \
-F 'permissions[send_messages][enabled]=0'
```

Returns a [Role](#) object

Create a single rubric [RubricsController#create](#)

POST /api/v1/courses/:course\_id/rubrics

Scope: url:POST /api/v1/courses/:course\_id/rubrics

Returns the rubric with the given id.

Unfortunately this endpoint does not return a standard Rubric object, instead it returns a hash that looks like

```
{ 'rubric': Rubric, 'rubric_association': RubricAssociation }
```

This may eventually be deprecated in favor of a more standardized return value, but that is not currently planned.

Request Parameters:

Parameter	Type	Description
id	integer	The id of the rubric
rubric_association_id	integer	The id of the object with which this rubric is associated
rubric[title]	string	The title of the rubric
rubric[free_form_criterion_comments]	boolean	Whether or not you can write custom comments in the ratings field for a rubric
rubric_association[association_id]	integer	The id of the object with which this rubric is associated
rubric_association[association_type]	string	The type of object this rubric is associated with Allowed values: Assignment, Course, Account
rubric_association[use_for_grading]	boolean	Whether or not the associated rubric is used for grade calculation
rubric_association[hide_score_total]	boolean	Whether or not the score total is displayed within the rubric. This option is only available if the rubric is not used for grading.
rubric_association[purpose]	string	Whether or not the association is for grading (and thus linked to an assignment) or if it's to indicate the rubric should appear in its context
rubric[criteria]	Hash	An indexed Hash of RubricCriteria objects where the keys are integer ids and the values are the RubricCriteria objects

Update a single rubric [RubricsController#update](#)

PUT /api/v1/courses/:course\_id/rubrics/:id

Scope: url:PUT /api/v1/courses/:course\_id/rubrics/:id

Returns the rubric with the given id.

Unfortunatly this endpoint does not return a standard Rubric object, instead it returns a hash that looks like

```
{ 'rubric': Rubric, 'rubric_association': RubricAssociation }
```

This may eventually be deprecated in favor of a more standardized return value, but that is not currently planned.

Request Parameters:

Parameter	Type	Description
id	integer	The id of the rubric
rubric_association_id	integer	The id of the object with which this rubric is associated
rubric[title]	string	The title of the rubric
rubric[free_form_criterion_comments]	boolean	Whether or not you can write custom comments in the ratings field for a rubric
rubric[skip_updating_points_possible]	boolean	Whether or not to update the points possible
rubric_association[association_id]	integer	The id of the object with which this rubric is associated
rubric_association[association_type]	string	The type of object this rubric is associated with Allowed values: Assignment, Course, Account
rubric_association[use_for_grading]	boolean	Whether or not the associated rubric is used for grade calculation
rubric_association[hide_score_total]	boolean	Whether or not the score total is displayed within the rubric. This option is only available if the rubric is not used for grading.
rubric_association[purpose]	string	Whether or not the association is for grading (and thus linked to an assignment) or if it's to indicate the rubric should appear in its context Allowed values: <i>grading</i> , <i>bookmark</i>
rubric[criteria]	Hash	An indexed Hash of RubricCriteria objects where the keys are integer ids and the values are the RubricCriteria objects

Delete a single rubric [RubricsController#destroy](#)

DELETE /api/v1/courses/:course\_id/rubrics/:id

Scope: url:DELETE /api/v1/courses/:course\_id/rubrics/:id

Deletes a Rubric and removes all RubricAssociations.

Returns a [Rubric](#) object

[List rubrics RubricsApiController#index](#)

**GET /api/v1/accounts/:account\_id/rubrics**

**Scope:** url:GET|/api/v1/accounts/:account\_id/rubrics

**GET /api/v1/courses/:course\_id/rubrics**

**Scope:** url:GET|/api/v1/courses/:course\_id/rubrics

Returns the paginated list of active rubrics for the current context.

[Get a single rubric RubricsApiController#show](#)

**GET /api/v1/accounts/:account\_id/rubrics/:id**

**Scope:** url:GET|/api/v1/accounts/:account\_id/rubrics/:id

**GET /api/v1/courses/:course\_id/rubrics/:id**

**Scope:** url:GET|/api/v1/courses/:course\_id/rubrics/:id

Returns the rubric with the given id.

**Request Parameters:**

Parameter	Type	Description
include[]	string	Related records to include in the response.
		Allowed values: assessments, graded_assessments, peer_assessments, associations, assignment_associations, course_associations, account_associations
style	string	Applicable only if assessments are being returned. If included, returns either all criteria data associated with the assessment, or just the comments. If not included, both data and comments are omitted.
		Allowed values: full, comments_only

Returns a [Rubric](#) object

[Get the courses and assignments for RubricsApiController#used\\_locations](#)

**GET /api/v1/courses/:course\_id/rubrics/:id/used\_locations**

**Scope:** url:GET|/api/v1/courses/:course\_id/rubrics/:id/used\_locations

**GET /api/v1/accounts/:account\_id/rubrics/:id/used\_locations**

**Scope:** url:GET|/api/v1/accounts/:account\_id/rubrics/:id/used\_locations

Returns the rubric with the given id.

[Creates a rubric using a CSV file RubricsApiController#upload](#)

**POST /api/v1/courses/:course\_id/rubrics/upload**

**Scope:** url:POST|/api/v1/courses/:course\_id/rubrics/upload

**POST /api/v1/accounts/:account\_id/rubrics/upload**

**Scope:** url:POST|/api/v1/accounts/:account\_id/rubrics/upload

Returns the rubric import object that was created

[Templated file for importing a rubric RubricsApiController#upload\\_template](#)

**GET /api/v1/rubrics/upload\_template**

**Scope:** url:GET|/api/v1/rubrics/upload\_template

[Get the status of a rubric import RubricsApiController#upload\\_status](#)

**GET /api/v1/courses/:course\_id/rubrics/upload/:id**

**Scope:** url:GET|/api/v1/courses/:course\_id/rubrics/upload/:id

**GET /api/v1/accounts/:account\_id/rubrics/upload/:id**

**Scope:** url:GET|/api/v1/accounts/:account\_id/rubrics/upload/:id

Can return the latest rubric import for an account or course, or a specific import by id

[Create a single rubric assessment RubricAssessmentsController#create](#)

**POST /api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments**

**Scope:** url:POST|/api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments

Returns the rubric assessment with the given id. The returned object also provides the information of

:ratings, :assessor\_name, :related\_group\_submissions\_and\_assessments, :artifact

**Request Parameters:**

Parameter	Type	Description
course_id	integer	The id of the course
rubric_association_id	integer	The id of the object with which this rubric assessment is associated
provisional	string	(optional) Indicates whether this assessment is provisional, defaults to false.
final	string	(optional) Indicates a provisional grade will be marked as final. It only takes effect if the provisional param is passed as true. Defaults to false.
graded_anonymously	boolean	(optional) Defaults to false
rubric_assessment	Hash	A Hash of data to complement the rubric assessment: The user id that refers to the person being assessed
		rubric_assessment[user_id]
	Hash	Assessment type. There are only three valid types: 'grading', 'peer_review', or 'provisional_grade'
		rubric_assessment[assessment_type]
	Hash	The points awarded for this row.
		rubric_assessment[criterion_id][points]
Hash	Comments to add for this row.	
	rubric_assessment[criterion_id][comments]	
For each criterion_id, change the id by the criterion number, ex: criterion_123 If the criterion_id is not specified it defaults to false, and nothing is updated.		

Update a single rubric assessment RubricAssessmentsController#update

PUT /api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments/:id

Scope: url:PUT /api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments/:id

Returns the rubric assessment with the given id. The returned object also provides the information of :ratings, :assessor\_name, :related\_group\_submissions\_and\_assessments, :artifact

Request Parameters:

Parameter	Type	Description
id	integer	The id of the rubric assessment
course_id	integer	The id of the course
rubric_association_id	integer	The id of the object with which this rubric assessment is associated
provisional	string	(optional) Indicates whether this assessment is provisional, defaults to false.
final	string	(optional) Indicates a provisional grade will be marked as final. It only takes effect if the provisional param is passed as true. Defaults to false.
graded_anonymously	boolean	(optional) Defaults to false
rubric_assessment	Hash	A Hash of data to complement the rubric assessment: The user id that refers to the person being assessed rubric_assessment[user_id]
		Assessment type. There are only three valid types: 'grading', 'peer_review', or 'provisional_grade' rubric_assessment[assessment_type]
		The points awarded for this row. rubric_assessment[criterion_id][points]
		Comments to add for this row. rubric_assessment[criterion_id][comments]
		For each criterion_id, change the id by the criterion number, ex: criterion_123 If the criterion_id is not specified it defaults to false, and nothing is updated.

Delete a single rubric assessment RubricAssessmentsController#destroy

DELETE /api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments/:id

Scope: url:DELETE /api/v1/courses/:course\_id/rubric\_associations/:rubric\_association\_id/rubric\_assessments/:id

Deletes a rubric assessment

Returns a RubricAssessment object

Create a RubricAssociation RubricAssociationsController#create

POST /api/v1/courses/:course\_id/rubric\_associations

Scope: url:POST /api/v1/courses/:course\_id/rubric\_associations

Returns the rubric with the given id.

Request Parameters:

Parameter	Type	Description
rubric_association[rubric_id]	integer	The id of the Rubric
rubric_association[association_id]	integer	The id of the object with which this rubric is associated
rubric_association[association_type]	string	The type of object this rubric is associated with Allowed values: Assignment, Course, Account
rubric_association[title]	string	The name of the object this rubric is associated with
rubric_association[use_for_grading]	boolean	Whether or not the associated rubric is used for grade calculation
rubric_association[hide_score_total]	boolean	Whether or not the score total is displayed within the rubric. This option is only available if the rubric is not used for grading.
rubric_association[purpose]	string	Whether or not the association is for grading (and thus linked to an assignment) or if it's to indicate the rubric should appear in its context Allowed values: grading, bookmark
rubric_association[bookmarked]	boolean	Whether or not the associated rubric appears in its context

Returns a RubricAssociation object

Update a RubricAssociation RubricAssociationsController#update

PUT /api/v1/courses/:course\_id/rubric\_associations/:id

Scope: url:PUT /api/v1/courses/:course\_id/rubric\_associations/:id

Returns the rubric with the given id.

Request Parameters:

Parameter	Type	Description
id	integer	The id of the RubricAssociation to update
rubric_association[rubric_id]	integer	The id of the Rubric
rubric_association[association_id]	integer	The id of the object with which this rubric is associated
rubric_association[association_type]	string	The type of object this rubric is associated with Allowed values: Assignment, Course, Account
rubric_association[title]	string	The name of the object this rubric is associated with
rubric_association[use_for_grading]	boolean	Whether or not the associated rubric is used for grade calculation
rubric_association[hide_score_total]	boolean	Whether or not the score total is displayed within the rubric. This option is only available if the rubric is not used for grading.

Parameter	Type	Description
rubric_association[purpose]	string	Whether or not the association is for grading (and thus linked to an assignment) or if it's to indicate the rubric should appear in its context Allowed values: grading, bookmark
rubric_association[bookmarked]	boolean	Whether or not the associated rubric appears in its context
Returns a <a href="#">RubricAssociation</a> object		

[Delete a RubricAssociation RubricAssociationsController#destroy](#)

**DELETE** /api/v1/courses/:course\_id/rubric\_associations/:id

**Scope:** url:DELETE|/api/v1/courses/:course\_id/rubric\_associations/:id

Delete the RubricAssociation with the given ID

Returns a [RubricAssociation](#) object

[Download UUID Mapping for this Sandbox](#)

**GET** /api/lti/uuid\_map

**Scope:** url:GET|/api/lti/uuid\_map

This endpoint returns a CSV file with the UUID mapping for the sandbox. The CSV has three columns:

\* 'type' - The object type  
\* 'original\_uuid' - The UUID of an object from the template  
\* 'new\_uuid' - The UUID of the corresponding object in the sandbox

[Create a Score Lti::Ims::ScoresController#create](#)

**POST** /api/lti/courses/:course\_id/line\_items/:line\_item\_id/scores

**Scope:** url:POST|/api/lti/courses/:course\_id/line\_items/:line\_item\_id/scores

Create a new Result from the score params. If this is for the first created line\_item for a resourceLinkId, or it is a line item that is not attached to a resourceLinkId, then a submission record will be created for the associated assignment when gradingProgress is set to FullyGraded or PendingManual.

The submission score will also be updated when a score object is sent with either of those two values for gradingProgress. If a score object is sent with either of FullyGraded or PendingManual as the value for gradingProgress and scoreGiven is missing, the assignment will not be graded. This also supposes the line\_item meets the condition to create a submission.

A submission comment with an unknown author will be created when the comment value is included. This also supposes the line\_item meets the condition to create a submission.

It is also possible to submit a file along with this score, which will attach the file to the submission that is created. Files should be formatted as Content Items, with the correct syntax below.

Returns a url pointing to the Result. If any files were submitted, also returns the Content Items which were sent in the request, each with a url pointing to the Progress of the file upload.

**Request Parameters:**

Parameter	Type	Description
userId	Required string	The lti_user_id or the Canvas user_id. Returns a 422 if user not found in Canvas or is not a student.
activityProgress	Required string	Indicate to Canvas the status of the user towards the activity's completion. Must be one of Initialized, Started, InProgress, Submitted, Completed.
gradingProgress	Required string	Indicate to Canvas the status of the grading process. A value of PendingManual will require intervention by a grader. Values of NotReady, Failed, and Pending will cause the scoreGiven to be ignored. FullyGraded values will require no action. Possible values are NotReady, Failed, Pending, PendingManual, FullyGraded.
timestamp	Required string	Date and time when the score was modified in the tool. Should use ISO8601-formatted date with subsecond precision. Returns a 400 if the timestamp is earlier than the updated_at time of the Result.
scoreGiven	number	The Current score received in the tool for this line item and user, scaled to the scoreMaximum
scoreMaximum	number	Maximum possible score for this result; it must be present if scoreGiven is present. Returns 422 if not present when scoreGiven is present.
comment	string	Comment visible to the student about this score.
submission	Object	Contains metadata about the submission attempt. Supported fields listed below.
submission[submittedAt]	string	Date and time that the submission was originally created. Should use ISO8601-formatted date with subsecond precision.
https://canvas.instructure.com/lti/submission	Object	(EXTENSION) Optional submission type and data. Fields listed below.
https://canvas.instructure.com/lti/submission[new_submission]	boolean	(EXTENSION field) flag to indicate that this is a new submission. Defaults to true unless submission_type is none.
https://canvas.instructure.com/lti/submission[preserve_score]	boolean	(EXTENSION field) flag to prevent a request from clearing an existing grade for a submission. Defaults to false.
https://canvas.instructure.com/lti/submission[prioritize_non_tool_grade]	boolean	(EXTENSION field) flag to prevent a request from overwriting an existing grade for a submission. Defaults to false.
https://canvas.instructure.com/lti/submission[submission_type]	string	(EXTENSION field) permissible values are: none, basic_lti_launch, online_text_entry, external_tool, online_upload, or online_url. Defaults to external_tool. Ignored if content_items are provided.
https://canvas.instructure.com/lti/submission[submission_data]	string	(EXTENSION field) submission data (URL or body text). Only used for submission_types basic_lti_launch, online_text_entry, online_url. Ignored if content_items are provided.
https://canvas.instructure.com/lti/submission[submitted_at]	string	(EXTENSION field) Date and time that the submission was originally created. Should use ISO8601-formatted date with subsecond precision. This should match the date and time that the original submission happened in Canvas. Use of submission.submittedAt is preferred.
https://canvas.instructure.com/lti/submission[content_items]	Array	(EXTENSION field) Files that should be included with the submission. Each item should contain 'type: file', and a url pointing to the file. It can also contain a title, and an explicit MIME type if needed (otherwise, MIME type will be inferred from the title or url). If any items are present, submission_type will be online_upload.

**Example Request:**

```
{
  "timestamp": "2017-04-16T18:54:36.736+00:00",
  "scoreGiven": 83,
  "scoreMaximum": 100,
  "comment": "This is exceptional work.",
  "submission": {
    "submittedAt": "2017-04-14T18:54:36.736+00:00"
  },
  "activityProgress": "Completed",
  "gradingProgress": "FullyGraded",
  "userId": "5323497",
  "https://canvas.instructure.com/lti/submission": {
    "new_submission": true,
    "preserve_score": false,
    "submission_type": "online_url",
    "submission_data": "https://instructure.com",
    "submitted_at": "2017-04-14T18:54:36.736+00:00",
    "content_items": [
      {
        "type": "file",
        "url": "https://instructure.com/test_file.txt",
        "title": "Submission File",
        "media_type": "text/plain"
      }
    ]
  }
}
```

**Example Response:**

```
{
  "resultUrl": "https://canvas.instructure.com/url/to/result",
  "https://canvas.instructure.com/lti/submission": {
    "content_items": [
      {
        "type": "file",
        "url": "https://instructure.com/test_file.txt",
        "title": "Submission File",
        "progress": "https://canvas.instructure.com/url/to/progress"
      }
    ]
  }
}
```

Find recipients SearchController#recipients

GET /api/v1/conversations/find\_recipients

Scope: url:GET//api/v1/conversations/find\_recipients

GET /api/v1/search/recipients

Scope: url:GET//api/v1/search/recipients

Find valid recipients (users, courses and groups) that the current user can send messages to. The /api/v1/search/recipients path is the preferred endpoint, /api/v1/conversations/find\_recipients is deprecated.

Pagination is supported.

Request Parameters:

Parameter	Type	Description
search	string	Search terms used for matching users/courses/groups (e.g. "bob smith"). If multiple terms are given (separated via whitespace), only results matching all terms will be returned.
context	string	Limit the search to a particular course/group (e.g. "course_3" or "group_4").
exclude[]	string	Array of ids to exclude from the search. These may be user ids or course/group ids prefixed with "course_" or "group_" respectively, e.g. <code>exclude[]=1&amp;exclude=2&amp;exclude[]=course_3</code>
type	string	Limit the search just to users or contexts (groups/courses). Allowed values: user, context
user_id	integer	Search for a specific user id. This ignores the other above parameters, and will never return more than one result.
from_conversation_id	integer	When searching by user_id, only users that could be normally messaged by this user will be returned. This parameter allows you to specify a conversation that will be referenced for a shared context – if both the current user and the searched user are in the conversation, the user will be returned. This is used to start new side conversations.
permissions[]	string	Array of permission strings to be checked for each matched context (e.g. "send_messages"). This argument determines which permissions may be returned in the response; it won't prevent contexts from being returned if they don't grant the permission(s).

API response field:

- id  
The unique identifier for the user/context. For groups/courses, the id is prefixed by "group\_" / "course\_" respectively.
- name  
The name of the context or short name of the user
- full\_name  
Only set for users. The full name of the user
- avatar\_url  
Avatar image url for the user/context
- type  
"context" | "course" | "section" | "group" | "user" | null  
Type of recipients to return, defaults to null (all). "context" encompasses "course", "section" and "group"
- types[]  
Array of recipient types to return (see type above), e.g. `types[]=user&types=course`
- user\_count  
Only set for contexts, indicates number of messageable users
- common\_courses  
Only set for users. Hash of course ids and enrollment types for each course to show what they share with this user
- common\_groups  
Only set for users. Hash of group ids and enrollment types for each group to show what they share with this user
- permissions[]  
Only set for contexts. Mapping of requested permissions that the context grants the current user, e.g. { send\_messages: true }

Example Response:

```
{
  {"id": "group_1", "name": "the group", "type": "context", "user_count": 3},
  {"id": 2, "name": "greg", "full_name": "greg jones", "common_courses": {}, "common_groups": {"1": {"Member"}}}
}
```

List all courses SearchController#all\_courses

GET /api/v1/search/all\_courses

Scope: url:GET//api/v1/search/all\_courses

A paginated list of all courses visible in the public index

Request Parameters:

Parameter	Type	Description
search	string	Search terms used for matching users/courses/groups (e.g. "bob smith"). If multiple terms are given (separated via whitespace), only results matching all terms will be returned.
public_only	boolean	Only return courses with public content. Defaults to false.
open_enrollment_only	boolean	Only return courses that allow self enrollment. Defaults to false.

List course sections SectionsController#index

GET /api/v1/courses/:course\_id/sections

Scope: url:GET//api/v1/courses/:course\_id/sections

A paginated list of the list of sections for this course.

Request Parameters:

Parameter	Type	Description
-----------	------	-------------

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>• “students”: Associations to include with the group. Note: this is only available if you have permission to view users or grades in the course</li><li>• “avatar_url”: Include the avatar URLs for students returned.</li><li>• “enrollments”: If ‘students’ is also included, return the section enrollment for each student</li><li>• “total_students”: Returns the total amount of active and invited students for the course section</li><li>• “passback_status”: Include the grade passback status.</li><li>• “permissions”: Include whether section grants :manage_calendar permission to the caller</li></ul>
		Allowed values: students, avatar_url, enrollments, total_students, passback_status, permissions

search\_term string When included, searches course sections for the term. Returns only matching results. Term must be at least 2 characters.

Returns a list of [Section](#) objects

Create course section [SectionsController#create](#)

POST /api/v1/courses/:course\_id/sections

Scope: url:POST|/api/v1/courses/:course\_id/sections

Creates a new section for this course.

Request Parameters:

Parameter	Type	Description
course_section[name]	string	The name of the section
course_section[sis_section_id]	string	The sis ID of the section. Must have manage_sis permission to set. This is ignored if caller does not have permission to set.
course_section[integration_id]	string	The integration_id of the section. Must have manage_sis permission to set. This is ignored if caller does not have permission to set.
course_section[start_at]	DateTime	Section start date in ISO8601 format, e.g. 2011-01-01T01:00Z
course_section[end_at]	DateTime	Section end date in ISO8601 format. e.g. 2011-01-01T01:00Z
course_section[restrict_enrollments_to_section_dates]	boolean	Set to true to restrict user enrollments to the start and end dates of the section.
enable_sis_reactivation	boolean	When true, will first try to re-activate a deleted section with matching sis_section_id if possible.

Returns a [Section](#) object

Cross-list a Section [SectionsController#crosslist](#)

POST /api/v1/sections/:id/crosslist/:new\_course\_id

Scope: url:POST|/api/v1/sections/:id/crosslist/:new\_course\_id

Move the Section to another course. The new course may be in a different account (department), but must belong to the same root account (institution).

Request Parameters:

Parameter	Type	Description
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Returns a [Section](#) object

De-cross-list a Section [SectionsController#uncrosslist](#)

DELETE /api/v1/sections/:id/crosslist

Scope: url:DELETE|/api/v1/sections/:id/crosslist

Undo cross-listing of a Section, returning it to its original course.

Request Parameters:

Parameter	Type	Description
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Returns a [Section](#) object

Edit a section [SectionsController#update](#)

PUT /api/v1/sections/:id

Scope: url:PUT|/api/v1/sections/:id

Modify an existing section.

Request Parameters:

Parameter	Type	Description
course_section[name]	string	The name of the section
course_section[sis_section_id]	string	The sis ID of the section. Must have manage_sis permission to set.
course_section[integration_id]	string	The integration_id of the section. Must have manage_sis permission to set.
course_section[start_at]	DateTime	Section start date in ISO8601 format, e.g. 2011-01-01T01:00Z
course_section[end_at]	DateTime	Section end date in ISO8601 format. e.g. 2011-01-01T01:00Z
course_section[restrict_enrollments_to_section_dates]	boolean	Set to true to restrict user enrollments to the start and end dates of the section.
override_sis_stickiness	boolean	Default is true. If false, any fields containing “sticky” changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

Returns a [Section](#) object

Get section information [SectionsController#show](#)

GET /api/v1/courses/:course\_id/sections/:id

Scope: url:GET|/api/v1/courses/:course\_id/sections/:id

GET /api/v1/sections/:id

Scope: url:GET|/api/v1/sections/:id



Gets details about a specific section

Request Parameters:

Parameter	Type	Description
		<ul style="list-style-type: none"><li>“students”: Associations to include with the group. Note: this is only available if you have permission to view users or grades in the course</li><li>“avatar_url”: Include the avatar URLs for students returned.</li><li>“enrollments”: If ‘students’ is also included, return the section enrollment for each student</li></ul>
include[]	string	<ul style="list-style-type: none"><li>“total_students”: Returns the total amount of active and invited students for the course section</li><li>“passback_status”: Include the grade passback status.</li><li>“permissions”: Include whether section grants :manage_calendar permission to the caller</li></ul>
Allowed values: students, avatar_url, enrollments, total_students, passback_status, permissions		

Returns a [Section](#) object

Delete a section [SectionsController#destroy](#)

DELETE /api/v1/sections/:id

Scope: url:DELETE|/api/v1/sections/:id

Delete an existing section. Returns the former Section.

Returns a [Section](#) object

Get Kaltura config [ServicesApiController#show\\_kaltura\\_config](#)

GET /api/v1/services/kaltura

Scope: url:GET|/api/v1/services/kaltura

Return the config information for the Kaltura plugin in json format.

API response field:

- enabled  
Enabled state of the Kaltura plugin
- domain  
Main domain of the Kaltura instance (This is the URL where the Kaltura API resides)
- resources\_domain  
Kaltura URL for grabbing thumbnails and other resources
- rtmp\_domain  
Hostname to be used for RTMP recording
- partner\_id  
Partner ID used for communicating with the Kaltura instance

Example Response:

```
# For an enabled Kaltura plugin:
{
  'domain': 'kaltura.example.com',
  'enabled': true,
  'partner_id': '123456',
  'resource_domain': 'cdn.kaltura.example.com',
  'rtmp_domain': 'rtmp.example.com'
}

# For a disabled or unconfigured Kaltura plugin:
{
  'enabled': false
}
```

Start Kaltura session [ServicesApiController#start\\_kaltura\\_session](#)

POST /api/v1/services/kaltura\_session

Scope: url:POST|/api/v1/services/kaltura\_session

Start a new Kaltura session, so that new media can be recorded and uploaded to this Canvas instance's Kaltura instance.

API response field:

- ks  
The kaltura session id, for use in the kaltura v3 API. This can be used in the uploadtoken service, for instance, to upload a new media file into kaltura.

Example Response:

```
{
  'ks': '1e39ad585f30c4fa1ef5752b51bd69fe'
}
```

Share a BrandConfig (Theme) [SharedBrandConfigsController#create](#)

POST /api/v1/accounts/:account\_id/shared\_brand\_configs

Scope: url:POST|/api/v1/accounts/:account\_id/shared\_brand\_configs

Create a SharedBrandConfig, which will give the given brand\_config a name and make it available to other users of this account.

Request Parameters:

Parameter	Type	Description
shared_brand_config[name]		Required string Name to share this BrandConfig (theme) as.
shared_brand_config[brand_config_md5]		Required string MD5 of brand_config to share

Example Request:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/shared_brand_configs' \
-X POST \
-F 'shared_brand_config[name]=Crimson and Gold Theme' \
-F 'shared_brand_config[brand_config_md5]=a1f113321fa024e7a14cb0948597a2a4' \
-H "Authorization: Bearer <token>"
```

Returns a [SharedBrandConfig](#) object

Update a shared theme [SharedBrandConfigsController#update](#)

PUT /api/v1/accounts/:account\_id/shared\_brand\_configs/:id

Scope: url:PUT|/api/v1/accounts/:account\_id/shared\_brand\_configs/:id

Update the specified shared\_brand\_config with a new name or to point to a new brand\_config. Uses same parameters as create.

Example Request:

```
curl -X PUT 'https://<canvas>/api/v1/accounts/<account_id>/shared_brand_configs/<shared_brand_config_id>' \
-H "Authorization: Bearer <token>" \
```

```
-F 'shared_brand_config[name]=New Name' \
-F 'shared_brand_config[brand_config_md5]=a1f113321fa024e7a14cb0948597a2a4'
```

Returns a [SharedBrandConfig](#) object

### [Un-share a BrandConfig \(Theme\) SharedBrandConfigsController#destroy](#)

#### DELETE /api/v1/shared\_brand\_configs/:id

**Scope:** url:DELETE /api/v1/shared\_brand\_configs/:id

Delete a SharedBrandConfig, which will unshare it so you nor anyone else in your account will see it as an option to pick from.

#### Example Request:

```
curl -X DELETE https://<canvas>/api/v1/shared_brand_configs/<id> \
-H 'Authorization: Bearer <token>'
```

Returns a [SharedBrandConfig](#) object

### [Get SIS import error list SisImportErrorsApiController#index](#)

#### GET /api/v1/accounts/:account\_id/sis\_imports/:id/errors

**Scope:** url:GET /api/v1/accounts/:account\_id/sis\_imports/:id/errors

#### GET /api/v1/accounts/:account\_id/sis\_import\_errors

**Scope:** url:GET /api/v1/accounts/:account\_id/sis\_import\_errors

Returns the list of SIS import errors for an account or a SIS import. Import errors are only stored for 30 days.

#### Example:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/sis_imports/<id>/sis_import_errors' \
-H 'Authorization: Bearer <token>'
```

#### Example:

```
curl 'https://<canvas>/api/v1/accounts/<account_id>/sis_import_errors' \
-H 'Authorization: Bearer <token>'
```

#### Request Parameters:

Parameter	Type	Description
failure	boolean	If set, only shows errors on a sis import that would cause a failure.

Returns a list of [SisImportError](#) objects

### [Get SIS import list SisImportsApiController#index](#)

#### GET /api/v1/accounts/:account\_id/sis\_imports

**Scope:** url:GET /api/v1/accounts/:account\_id/sis\_imports

Returns the list of SIS imports for an account

#### Example:

```
curl https://<canvas>/api/v1/accounts/<account_id>/sis_imports \
-H 'Authorization: Bearer <token>'
```

#### Request Parameters:

Parameter	Type	Description
created_since	DateTime	If set, only shows imports created after the specified date (use ISO8601 format)
created_before	DateTime	If set, only shows imports created before the specified date (use ISO8601 format)
workflow_state[]	string	If set, only returns imports that are in the given state. Allowed values: initializing, created, importing, cleanup_batch, imported, imported_with_messages, aborted, failed, failed_with_messages, restoring, partially_restored, restored

Returns a list of [SisImport](#) objects

### [Get the current importing SIS import SisImportsApiController#importing](#)

#### GET /api/v1/accounts/:account\_id/sis\_imports/importing

**Scope:** url:GET /api/v1/accounts/:account\_id/sis\_imports/importing

Returns the SIS imports that are currently processing for an account. If no imports are running, will return an empty array.

#### Example:

```
curl https://<canvas>/api/v1/accounts/<account_id>/sis_imports/importing \
-H 'Authorization: Bearer <token>'
```

Returns a [SisImport](#) object

### [Import SIS data SisImportsApiController#create](#)

#### POST /api/v1/accounts/:account\_id/sis\_imports

**Scope:** url:POST /api/v1/accounts/:account\_id/sis\_imports

Import SIS data into Canvas. Must be on a root account with SIS imports enabled.

For more information on the format that's expected here, please see the “SIS CSV” section in the API docs.

#### Request Parameters:

Parameter	Type	Description
import_type	string	Choose the data format for reading SIS data. With a standard Canvas install, this option can only be ‘instructure_csv’, and if unprovided, will be assumed to be so. Can be part of the query string.
attachment	string	There are two ways to post SIS import data - either via a multipart/form-data form-field-style attachment, or via a non-multipart raw post request.  ‘attachment’ is required for multipart/form-data style posts. Assumed to be SIS data from a file upload form field named ‘attachment’.  Examples: <pre>curl -F attachment=@filename -H "Authorization: Bearer &lt;token&gt;" \ https://&lt;canvas&gt;/api/v1/accounts/&lt;account_id&gt;/sis_imports.json?import_type=instructure_csv</pre> If you decide to do a raw post, you can skip the ‘attachment’ argument, but you will then be required to provide a suitable Content-Type header. You are encouraged to also provide the ‘extension’ argument.  Examples: <pre>curl -H 'Content-Type: application/octet-stream' --data-binary @&lt;filename&gt;.zip \ -H "Authorization: Bearer &lt;token&gt;" \ https://&lt;canvas&gt;/api/v1/accounts/&lt;account_id&gt;/sis_imports.json?import_type=instructure_csv&amp;extension=zip</pre> <pre>curl -H 'Content-Type: application/zip' --data-binary @&lt;filename&gt;.zip \ -H "Authorization: Bearer &lt;token&gt;" \ https://&lt;canvas&gt;/api/v1/accounts/&lt;account_id&gt;/sis_imports.json?import_type=instructure_csv</pre> <pre>curl -H 'Content-Type: text/csv' --data-binary @&lt;filename&gt;.csv \ -H "Authorization: Bearer &lt;token&gt;" \ https://&lt;canvas&gt;/api/v1/accounts/&lt;account_id&gt;/sis_imports.json?import_type=instructure_csv</pre> <pre>curl -H 'Content-Type: text/csv' --data-binary @&lt;filename&gt;.csv \ -H "Authorization: Bearer &lt;token&gt;" \ https://&lt;canvas&gt;/api/v1/accounts/&lt;account_id&gt;/sis_imports.json?import_type=instructure_csv&amp;batch_mode=l&amp;batch_mode_term_id=15</pre>

Parameter	Type	Description
		If the attachment is a zip file, the uncompressed file(s) cannot be 100x larger than the zip, or the import will fail. For example, if the zip file is 1KB but the total size of the uncompressed file(s) is 100KB or greater the import will fail. There is a hard cap of 50 GB.
extension	string	Recommended for raw post request style imports. This field will be used to distinguish between zip, xml, csv, and other file format extensions that would usually be provided with the filename in the multipart post request scenario. If not provided, this value will be inferred from the Content-Type, falling back to zip-file format if all else fails.
batch_mode	boolean	If set, this SIS import will be run in batch mode, deleting any data previously imported via SIS that is not present in this latest import. See the SIS CSV Format page for details. Batch mode cannot be used with diffing.
batch_mode_term_id	string	Limit deletions to only this term. Required if batch mode is enabled.
multi_term_batch_mode	boolean	Runs batch mode against all terms in terms file. Requires change_threshold.
skip_deletes	boolean	When set the import will skip any deletes. This does not account for objects that are deleted during the batch mode cleanup process.
override_sis_stickiness	boolean	Default is false. If true, any fields containing "sticky" or UI changes will be overridden. See SIS CSV Format documentation for information on which fields can have SIS stickiness
add_sis_stickiness	boolean	This option, if present, will process all changes as if they were UI changes. This means that "stickiness" will be added to changed fields. This option is only processed if 'override_sis_stickiness' is also provided.
clear_sis_stickiness	boolean	This option, if present, will clear "stickiness" from all fields touched by this import. Requires that 'override_sis_stickiness' is also provided. If 'add_sis_stickiness' is also provided, 'clear_sis_stickiness' will overrule the behavior of 'add_sis_stickiness'
update_sis_id_if_login_claimed	boolean	This option, if present, will override the old (or non-existent) non-matching SIS ID with the new SIS ID in the upload, if a pseudonym is found from the login field and the SIS ID doesn't match.
diffing_data_set_identifier	string	If set on a CSV import, Canvas will attempt to optimize the SIS import by comparing this set of CSVs to the previous set that has the same data set identifier, and only applying the difference between the two. See the SIS CSV Format documentation for more details. Diffing cannot be used with batch_mode
diffing_remaster_data_set	boolean	If true, and diffing_data_set_identifier is sent, this SIS import will be part of the data set, but diffing will not be performed. See the SIS CSV Format documentation for details.
diffing_drop_status	string	If diffing_drop_status is passed, this SIS import will use this status for enrollments that are not included in the sis_batch. Defaults to 'deleted' Allowed values: deleted, completed, inactive
diffing_user_remove_status	string	For users removed from one batch to the next one using the same diffing_data_set_identifier, set their status to the value of this argument. Defaults to 'deleted'. Allowed values: deleted, suspended
batch_mode_enrollment_drop_status	string	If batch_mode_enrollment_drop_status is passed, this SIS import will use this status for enrollments that are not included in the sis_batch. This will have an effect if multi_term_batch_mode is set. Defaults to 'deleted' This will still mark courses and sections that are not included in the sis_batch as deleted, and subsequently enrollments in the deleted courses and sections as deleted. Allowed values: deleted, completed, inactive
change_threshold	integer	If set with batch_mode, the batch cleanup process will not run if the number of items deleted is higher than the percentage set. If set to 10 and a term has 200 enrollments, and batch would delete more than 20 of the enrollments the batch will abort before the enrollments are deleted. The change_threshold will be evaluated for course, sections, and enrollments independently. If set with diffing, diffing will not be performed if the files are greater than the threshold as a percent. If set to 5 and the file is more than 5% smaller or more than 5% larger than the file that is being compared to, diffing will not be performed. If the files are less than 5%, diffing will be performed. The way the percent is calculated is by taking the size of the current import and dividing it by the size of the previous import. The formula used is:  (1 - current_file_size / previous_file_size)  * 100 See the SIS CSV Format documentation for more details. Required for multi_term_batch_mode.
diff_row_count_threshold	integer	If set with diffing, diffing will not be performed if the number of rows to be run in the fully calculated diff import exceeds the threshold.

Returns a [SisImport](#) object

**[Get SIS import status SisImportsApiController#show](#)**

**GET /api/v1/accounts/:account\_id/sis\_imports/:id**

**Scope:** url:GET|/api/v1/accounts/:account\_id/sis\_imports/:id

Get the status of an already created SIS import.

Examples:  
curl https://<canvas>/api/v1/accounts/<account\_id>/sis\_imports/<sis\_import\_id> \-H 'Authorization: Bearer <token>'

Returns a [SisImport](#) object

**[Restore workflow states of SIS imported items SisImportsApiController#restore\\_states](#)**

**PUT /api/v1/accounts/:account\_id/sis\_imports/id/restore\_states**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/sis\_imports/:id/restore\_states

This will restore the workflow\_state for all the items that changed their workflow\_state during the import being restored. This will restore states for items imported with the following importers: accounts.csv terms.csv courses.csv sections.csv group\_categories.csv groups.csv users.csv admins.csv This also restores states for other items that changed during the import. An example would be if an enrollment was deleted from a sis import and the group\_membership was also deleted as a result of the enrollment deletion, both items would be restored when the sis batch is restored.

**Request Parameters:**

Parameter	Type	Description
batch_mode	boolean	If set, will only restore items that were deleted from batch_mode.
undelete_only	boolean	If set, will only restore items that were deleted. This will ignore any items that were created or modified.
unconclude_only	boolean	If set, will only restore enrollments that were concluded. This will ignore any items that were created or deleted.

**Example Request:**

curl https://<canvas>/api/v1/accounts/<account\_id>/sis\_imports/<sis\_import\_id>/restore\_states \-H 'Authorization: Bearer <token>'

Returns a [Progress](#) object

**[Abort SIS import SisImportsApiController#abort](#)**

**PUT /api/v1/accounts/:account\_id/sis\_imports/id/abort**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/sis\_imports/:id/abort

Abort a SIS import that has not completed.

Aborting a sis batch that is running can take some time for every process to see the abort event. Subsequent sis batches begin to process 10 minutes after the abort to allow each process to clean up properly.

**Example Request:**

curl https://<canvas>/api/v1/accounts/<account\_id>/sis\_imports/<sis\_import\_id>/abort \-H 'Authorization: Bearer <token>'

Returns a [SisImport](#) object

**[Abort all pending SIS imports SisImportsApiController#abort\\_all\\_pending](#)**

**PUT /api/v1/accounts/:account\_id/sis\_imports/abort\_all\_pending**

**Scope:** url:PUT|/api/v1/accounts/:account\_id/sis\_imports/abort\_all\_pending

Abort already created but not processed or processing SIS imports.

**Example Request:**

```
curl https://<canvas>/api/v1/accounts/<account_id>/sis_imports/abort_all_pending \
-H "Authorization: Bearer <token>"
```

**[Retrieve assignments enabled for grade export to SIS SisApiController#sis\\_assignments](#)**

**GET /api/sis/accounts/:account\_id/assignments**

**Scope:** url:GET|/api/sis/accounts/:account\_id/assignments

**GET /api/sis/courses/:course\_id/assignments**

**Scope:** url:GET|/api/sis/courses/:course\_id/assignments

Retrieve a list of published assignments flagged as “post\_to\_sis”. See the Assignments API for more details on assignments. Assignment group and section information are included for convenience.

Each section includes course information for the origin course and the cross-listed course, if applicable. The ‘origin\_course’ is the course to which the section belongs or the course from which the section was cross-listed. Generally, the ‘origin\_course’ should be preferred when performing integration work. The ‘xlist\_course’ is provided for consistency and is only present when the section has been cross-listed. See Sections API and Courses Api for me details.

The ‘override’ is only provided if the Differentiated Assignments course feature is turned on and the assignment has an override for that section. When there is an override for the assignment the override object’s keys/values can be merged with the top level assignment object to create a view of the assignment object specific to that section. See Assignments api for more information on assignment overrides.

restricts to courses that start before this date (if they have a start date) restricts to courses that end after this date (if they have an end date) information to include.

\*student\_overrides\*: returns individual student override information

**Request Parameters:**

Parameter	Type	Description
account_id	integer	The ID of the account to query.
course_id	integer	The ID of the course to query.
starts_before	DateTime	When searching on an account,
ends_after	DateTime	When searching on an account,
include	string	Array of additional Allowed values: student_overrides

**[Disable assignments currently enabled for grade export to SIS DisablePostToSisApiController#disable\\_post\\_to\\_sis](#)**

**PUT /api/sis/courses/:course\_id/disable\_post\_to\_sis**

**Scope:** url:PUT|/api/sis/courses/:course\_id/disable\_post\_to\_sis

Disable all assignments flagged as “post\_to\_sis”, with the option of making it specific to a grading period, in a course.

On success, the response will be 204 No Content with an empty body.

On failure, the response will be 400 Bad Request with a body of a specific message.

For disabling assignments in a specific grading period

**Request Parameters:**

Parameter	Type	Description
course_id	integer	The ID of the course.
grading_period_id	integer	The ID of the grading period.

**Example Request:**

```
curl 'https://<canvas>/api/sis/courses/<course_id>/disable_post_to_sis' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"

curl 'https://<canvas>/api/sis/courses/<course_id>/disable_post_to_sis' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0" \
-d 'grading_period_id=1'
```

**[Search course content SmartSearchController#search](#)**

**BETA: This API endpoint is not finalized, and there could be breaking changes before its final release.**

**GET /api/v1/courses/:course\_id/smartsearch**

**Scope:** url:GET|/api/v1/courses/:course\_id/smartsearch

Find course content using a meaning-based search

**Request Parameters:**

Parameter	Type	Description
q	Required string	The search query
filter[]	string	Types of objects to search. By default, all supported types are searched. Supported types include pages, assignments, announcements, and discussion_topics.

Returns a list of [SearchResult](#) objects

**[Edit a submission comment SubmissionCommentsApiController#update](#)**

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/:id**

**Scope:** url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/:id

Edit the given submission comment.

**Request Parameters:**

Parameter	Type	Description
comment	string	If this argument is present, edit the text of a comment.

Returns a [SubmissionComment](#) object

**[Delete a submission comment SubmissionCommentsApiController#destroy](#)**

**DELETE /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/:id**

**Scope:** url:DELETE|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/:id

Delete the given submission comment.

Example Request:

```
curl https://canvas/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/comments/<cid> \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns a [SubmissionComment](#) object

Upload a file [SubmissionCommentsApiController#create\\_file](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/files

Scope: url: POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/comments/files

Upload a file to attach to a submission comment

See the [File Upload Documentation](#) for details on the file upload workflow.

The final step of the file upload workflow will return the attachment data, including the new file id. The caller can then PUT the file\_id to the submission API to attach it to a comment

Submit an assignment [SubmissionsController#create](#)

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions

Scope: url: POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions

POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions

Scope: url: POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions

Make a submission for an assignment. You must be enrolled as a student in the course/section to do this.

All online turn-in submission types are supported in this API. However, there are a few things that are not yet supported:

- Files can be submitted based on a file ID of a user or group file or through the [file upload API](#). However, there is no API yet for listing the user and group files.
- Media comments can be submitted, however, there is no API yet for creating a media comment to submit.
- Integration with Google Docs is not yet supported.

Request Parameters:

Parameter	Type	Description
comment[text_comment]	string	Include a textual comment with the submission.
submission[group_comment]	boolean	Whether or not this comment should be sent to the entire group (defaults to false). Ignored if this is not a group assignment or if no text_comment is provided.
submission[submission_type]	Required string	The type of submission being made. The assignment submission_types must include this submission type as an allowed option, or the submission will be rejected with a 400 error. The submission_type given determines which of the following parameters is used. For instance, to submit a URL, submission [submission_type] must be set to "online_url", otherwise the submission [url] parameter will be ignored. "basic_lti_launch" requires the assignment submission_type "online" or "external_tool" Allowed values: online_text_entry, online_url, online_upload, media_recording, basic_lti_launch, student_annotation
submission[body]	string	Submit the assignment as an HTML document snippet. Note this HTML snippet will be sanitized using the same ruleset as a submission made from the Canvas web UI. The sanitized HTML will be returned in the response as the submission body. Requires a submission_type of "online_text_entry".
submission[url]	string	Submit the assignment as a URL. The URL scheme must be "http" or "https", no "ftp" or other URL schemes are allowed. If no scheme is given (e.g. " <a href="#">www.example.com</a> ") then "http" will be assumed. Requires a submission_type of "online_url" or "basic_lti_launch".
submission[file_ids[]]	integer	Submit the assignment as a set of one or more previously uploaded files residing in the submitting user's files section (or the group's files section, for group assignments). To upload a new file to submit, see the submissions <a href="#">Upload a file API</a> . Requires a submission_type of "online_upload".
submission[media_comment_id]	string	The media comment id to submit. Media comment ids can be submitted via this API, however, note that there is not yet an API to generate or list existing media comments, so this functionality is currently of limited use. Requires a submission_type of "media_recording".
submission[media_comment_type]	string	The type of media comment being submitted. Allowed values: audio, video
submission[user_id]	integer	Submit on behalf of the given user. Requires grading permission.
submission[annotatable_attachment_id]	integer	The Attachment ID of the document being annotated. This should match the annotatable_attachment_id on the assignment. Requires a submission_type of "student_annotation".
submission[submitted_at]	DateTime	Choose the time the submission is listed as submitted at. Requires grading permission.

List assignment submissions [SubmissionsApiController#index](#)

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions

Scope: url: GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions

GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions

Scope: url: GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions

A paginated list of all existing submissions for an assignment.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the group. "group" will add group_id and group_name. Allowed values: submission_history, submission_comments, submission_html_comments, rubric_assessment, assignment, visibility, course, user, group, read_status, student_entered_score
grouped	boolean	If this argument is true, the response will be grouped by student groups.

API response field:

- assignment\_id  
The unique identifier for the assignment.
- user\_id  
The id of the user who submitted the assignment.
- grader\_id  
The id of the user who graded the submission. This will be null for submissions that haven't been graded yet. It will be a positive number if a real user has graded the submission and a negative number if the submission was graded by a process (e.g. Quiz autograder and autograding LTI tools). Specifically autograded quizzes set grader\_id to the negative of the quiz id. Submissions autograded by LTI tools set grader\_id to the negative of the tool id.

- **canvadoc\_document\_id**  
The id for the canvadoc document associated with this submission, if it was a file upload.
- **submitted\_at**  
The timestamp when the assignment was submitted, if an actual submission has been made.
- **score**  
The raw score for the assignment submission.
- **attempt**  
If multiple submissions have been made, this is the attempt number.
- **body**  
The content of the submission, if it was submitted directly in a text field.
- **grade**  
The grade for the submission, translated into the assignment grading scheme (so a letter grade, for example).
- **grade\_matches\_current\_submission**  
A boolean flag which is false if the student has re-submitted since the submission was last graded.
- **preview\_url**  
Link to the URL in canvas where the submission can be previewed. This will require the user to log in.
- **redo\_request**  
If the submission was reassigned
- **url**  
If the submission was made as a URL.
- **late**  
Whether the submission was made after the applicable due date.
- **assignment\_visible**  
Whether this assignment is visible to the user who submitted the assignment.
- **workflow\_state**  
The current status of the submission. Possible values: "submitted", "unsubmitted", "graded", "pending\_review"

Returns a list of [Submission](#) objects

List submissions for multiple assignments [SubmissionsApiController#for\\_students](#)

**GET /api/v1/courses/:course\_id/students/submissions**

**Scope:** url:GET /api/v1/courses/:course\_id/students/submissions

**GET /api/v1/sections/:section\_id/students/submissions**

**Scope:** url:GET /api/v1/sections/:section\_id/students/submissions

A paginated list of all existing submissions for a given set of students and assignments.

Request Parameters:

Parameter	Type	Description
student_ids[]	string	List of student ids to return submissions for. If this argument is omitted, return submissions for the calling user. Students may only list their own submissions. Observers may only list those of associated students. The special id "all" will return submissions for all students in the course/section as appropriate.
assignment_ids[]	string	List of assignments to return submissions for. If none are given, submissions for all assignments are returned.
grouped	boolean	If this argument is present, the response will be grouped by student, rather than a flat array of submissions.
post_to_sis	boolean	If this argument is set to true, the response will only include submissions for assignments that have the post_to_sis flag set to true and user enrollments that were added through sis.
submitted_since	DateTime	If this argument is set, the response will only include submissions that were submitted after the specified date_time. This will exclude submissions that do not have a submitted_at which will exclude unsubmitted submissions. The value must be formatted as ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
graded_since	DateTime	If this argument is set, the response will only include submissions that were graded after the specified date_time. This will exclude submissions that have not been graded. The value must be formatted as ISO 8601 YYYY-MM-DDTHH:MM:SSZ.
grading_period_id	integer	The id of the grading period in which submissions are being requested (Requires grading periods to exist on the account)
workflow_state	string	The current status of the submission Allowed values: submitted, unsubmitted, graded, pending_review
enrollment_state	string	The current state of the enrollments. If omitted will include all enrollments that are not deleted. Allowed values: active, concluded
state_based_on_date	boolean	If omitted it is set to true. When set to false it will ignore the effective state of the student enrollments and use the workflow_state for the enrollments. The argument is ignored unless enrollment_state argument is also passed.
order	string	The order submissions will be returned in. Defaults to "id". Doesn't affect results for "grouped" mode. Allowed values: id, graded_at
order_direction	string	Determines whether ordered results are returned in ascending or descending order. Defaults to "ascending". Doesn't affect results for "grouped" mode. Allowed values: ascending, descending
include[]	string	Associations to include with the group. 'total_scores' requires the 'grouped' argument. Allowed values: submission_history, submission_comments, submission_html_comments, rubric_assessment, assignment, total_scores, visibility, course, user, sub_assignment_submissions, student_entered_score

Example Response:

```
# Without grouped:
{
  { "assignment_id": 100, grade: 5, "user_id": 1, ... },
  { "assignment_id": 101, grade: 6, "user_id": 2, ... }
}

# With grouped:
{
  {
    "user_id": 1,
    "submissions": [
      { "assignment_id": 100, grade: 5, ... },
      { "assignment_id": 101, grade: 6, ... }
    ]
  }
}
```

Get a single submission SubmissionsApiController#show

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id

Scope: url:GET/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id

GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id

Scope: url:GET/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id

Get a single submission, based on user id.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the group. Allowed values: submission_history, submission_comments, submission_html_comments, rubric_assessment, full_rubric_assessment, visibility, course, user, read_status, student_entered_score

Get a single submission by anonymous id SubmissionsApiController#show\_anonymous

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Scope: url:GET/api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

GET /api/v1/sections/:section\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Scope: url:GET/api/v1/sections/:section\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Get a single submission, based on the submission's anonymous id.

Request Parameters:

Parameter	Type	Description
include[]	string	Associations to include with the group. Allowed values: submission_history, submission_comments, rubric_assessment, full_rubric_assessment, visibility, course, user, read_status

Upload a file SubmissionsApiController#create\_file

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/files

Scope: url:POST/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/files

POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/files

Scope: url:POST/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/files

Upload a file to a submission.

This API endpoint is the first step in uploading a file to a submission as a student. See the [File Upload Documentation](#) for details on the file upload workflow.

The final step of the file upload workflow will return the attachment data, including the new file id. The caller can then POST to submit the online\_upload assignment with these file ids.

Grade or comment on a submission SubmissionsApiController#update

PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id

Scope: url:PUT/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id

PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id

Scope: url:PUT/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id

Comment on and/or update the grading for a student's assignment submission. If any submission or rubric\_assessment arguments are provided, the user must have permission to manage grades in the appropriate context (course or section).

Request Parameters:

Parameter	Type	Description
comment[text_comment]	string	Add a textual comment to the submission.
comment[attempt]	integer	The attempt number (starts at 1) to associate the comment with.
comment[group_comment]	boolean	Whether or not this comment should be sent to the entire group (defaults to false). Ignored if this is not a group assignment or if no text_comment is provided.
comment[media_comment_id]	string	Add an audio/video comment to the submission. Media comments can be added via this API, however, note that there is not yet an API to generate or list existing media comments, so this functionality is currently of limited use.
comment[media_comment_type]	string	The type of media comment being added. Allowed values: audio, video
comment[file_ids[]]	integer	Attach files to this comment that were previously uploaded using the Submission Comment API's files action
include[visibility]	string	Whether this assignment is visible to the owner of the submission
prefer_points_over_scheme	boolean	Treat posted_grade as points if the value matches a grading scheme value
submission[posted_grade]	string	Assign a score to the submission, updating both the "score" and "grade" fields on the submission record. This parameter can be passed in a few different formats:  points  A floating point or integral value, such as "13.5". The grade will be interpreted directly as the score of the assignment. Values above assignment.points_possible are allowed, for awarding extra credit.  percentage  A floating point value appended with a percent sign, such as "40%". The grade will be interpreted as a percentage score on the assignment, where 100% == assignment.points_possible. Values above 100% are allowed, for awarding extra credit.  letter grade  A letter grade, following the assignment's defined letter grading scheme. For example, "A-". The resulting score will be the high end of the defined range for the letter grade. For instance, if "B" is defined as 86% to 84%, a letter grade of "B" will be worth 86%. The letter grade will be rejected if the assignment does not have a defined letter grading scheme. For more fine-grained control of scores, pass in points or percentage rather than the letter grade.  "pass/complete/fail/incomplete"  A string value of "pass" or "complete"  will give a score of 100%. "fail" or "incomplete" will give a score of 0.

Parameter	Type	Description
		Note that assignments with <code>grading_type</code> of “pass_fail” can only be assigned a score of 0 or <code>assignment.points_possible</code> , nothing inbetween. If a <code>posted_grade</code> in the “points” or “percentage” format is sent, the grade will only be accepted if the grade equals one of those two values.
<code>submission[excuse]</code>	boolean	Sets the “excused” status of an assignment.
<code>submission[late_policy_status]</code>	string	Sets the late policy status to either “late”, “missing”, “extended”, “none”, or null.  NB: “extended” values can only be set in the UI when the “UI features for ‘extended’ Submissions” Account Feature is on
<code>submission[sticker]</code>	string	Sets the sticker for the submission.  Allowed values: apple, basketball, bell, book, bookbag, briefcase, bus, calendar, chem, design, pencil, beaker, paintbrush, computer, column, pen, tablet, telescope, calculator, paperclip, composite_notebook, scissors, ruler, clock, globe, grad, gym, mail, microscope, mouse, music, notebook, page, panda1, panda2, panda3, panda4, panda5, panda6, panda7, panda8, panda9, presentation, science, science2, star, tag, tape, target, trophy
<code>submission[seconds_late_override]</code>	integer	Sets the seconds late if late policy status is “late”
<code>rubric_assessment</code>	RubricAssessment	<p>Assign a rubric assessment to this assignment submission. The sub-parameters here depend on the rubric for the assignment. The general format is, for each row in the rubric:</p> <p>The points awarded for this row.</p> <pre>rubric_assessment[criterion_id][points]</pre> <p>The rating id for the row.</p> <pre>rubric_assessment[criterion_id][rating_id]</pre> <p>Comments to add for this row.</p> <pre>rubric_assessment[criterion_id][comments]</pre> <p>For example, if the assignment rubric is (in JSON format):</p> <pre>{   {     'id': 'crit1',     'points': 10,     'description': 'Criterion 1',     'ratings':     [       { 'id': 'rat1', 'description': 'Good', 'points': 10 },       { 'id': 'rat2', 'description': 'Poor', 'points': 3 }     ]   },   {     'id': 'crit2',     'points': 5,     'description': 'Criterion 2',     'ratings':     [       { 'id': 'rat1', 'description': 'Exemplary', 'points': 5 },       { 'id': 'rat2', 'description': 'Complete', 'points': 5 },       { 'id': 'rat3', 'description': 'Incomplete', 'points': 0 }     ]   } }</pre> <p>Then a possible set of values for <code>rubric_assessment</code> would be:</p> <pre>rubric_assessment[crit1][points]=3&amp;rubric_assessment[crit1][rating_id]=rat1&amp;rubric_assessment[crit2][points]=5&amp;rubric_assessment[crit2][rating_id]=rat2&amp;rubric_assessment[crit2][comments]=Well%20done.</pre>

[Grade or comment on a submission by anonymous id SubmissionsApiController#update\\_anonymous](#)

PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Scope: url:PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Scope: url:PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/anonymous\_submissions/:anonymous\_id

Comment on and/or update the grading for a student's assignment submission, fetching the submission by anonymous id (instead of user id). If any submission or rubric\_assessment arguments are provided, the user must have permission to manage grades in the appropriate context (course or section).

Request Parameters:

Parameter	Type	Description
<code>comment[text_comment]</code>	string	Add a textual comment to the submission.
<code>comment[group_comment]</code>	boolean	Whether or not this comment should be sent to the entire group (defaults to false). Ignored if this is not a group assignment or if no <code>text_comment</code> is provided.
<code>comment[media_comment_id]</code>	string	Add an audio/video comment to the submission. Media comments can be added via this API, however, note that there is not yet an API to generate or list existing media comments, so this functionality is currently of limited use.
<code>comment[media_comment_type]</code>	string	The type of media comment being added.  Allowed values: audio, video
<code>comment[file_ids[]]</code>	integer	Attach files to this comment that were previously uploaded using the Submission Comment API's files action
<code>include[visibility]</code>	string	Whether this assignment is visible to the owner of the submission
<code>submission[posted_grade]</code>	string	<p>Assign a score to the submission, updating both the “score” and “grade” fields on the submission record. This parameter can be passed in a few different formats:</p> <p>points</p> <p>A floating point or integral value, such as “13.5”. The grade will be interpreted directly as the score of the assignment. Values above <code>assignment.points_possible</code> are allowed, for awarding extra credit.</p> <p>percentage</p> <p>A floating point value appended with a percent sign, such as “40%”. The grade will be interpreted as a percentage score on the assignment, where 100% == <code>assignment.points_possible</code>. Values above 100% are allowed, for awarding extra credit.</p> <p>letter grade</p> <p>A letter grade, following the assignment's defined letter grading scheme. For example, “A-”. The resulting score will be the high end of the defined range for the letter grade. For instance, if “B” is defined as 86% to 84%, a letter grade of “B” will be worth 86%. The letter grade will be rejected if the assignment does not have a defined letter grading scheme. For more fine-grained control of scores, pass in points or percentage rather than the letter grade.</p> <p>“pass/complete/fail/incomplete”</p> <p>A string value of “pass” or “complete”</p> <p>will give a score of 100%. “fail” or “incomplete” will give a score of 0.</p> <p>Note that assignments with <code>grading_type</code> of “pass_fail” can only be assigned a score of 0 or <code>assignment.points_possible</code>, nothing inbetween. If a <code>posted_grade</code> in the “points” or “percentage” format is sent, the grade will only be accepted if the grade equals one of those two values.</p>
<code>submission[excuse]</code>	boolean	Sets the “excused” status of an assignment.



Parameter	Type	Description
submission[late_policy_status]	string	Sets the late policy status to either "late", "missing", "extended", "none", or null.  NB: "extended" values can only be set in the UI when the "UI features for 'extended' Submissions" Account Feature is on
submission[seconds_late_override]	integer	Sets the seconds late if late policy status is "late"
rubric_assessment	RubricAssessment	Assign a rubric assessment to this assignment submission. The sub-parameters here depend on the rubric for the assignment. The general format is, for each row in the rubric:  The points awarded for this row.  rubric_assessment[criterion_id][points]  The rating id for the row.  rubric_assessment[criterion_id][rating_id]  Comments to add for this row.  rubric_assessment[criterion_id][comments]  For example, if the assignment rubric is (in JSON format):  <pre>{   {     'id': 'crit1',     'points': 10,     'description': 'Criterion 1',     'ratings':       [         { 'id': 'rat1', 'description': 'Good', 'points': 10 },         { 'id': 'rat2', 'description': 'Poor', 'points': 3 }       ]   },   {     'id': 'crit2',     'points': 5,     'description': 'Criterion 2',     'ratings':       [         { 'id': 'rat1', 'description': 'Exemplary', 'points': 5 },         { 'id': 'rat2', 'description': 'Complete', 'points': 5 },         { 'id': 'rat3', 'description': 'Incomplete', 'points': 0 }       ]   } }</pre> Then a possible set of values for rubric_assessment would be:  rubric_assessment[crit1][points]=3&rubric_assessment[crit1][rating_id]=rat1&rubric_assessment[crit2][points]=5&rubric_assessment[crit2][rating_id]=rat2&rubric_assessment[crit2][comments]=Well%20Done.

List gradeable students SubmissionsApiController#gradeable\_students

GET /api/v1/courses/:course\_id/assignments/:assignment\_id/gradeable\_students

Scope: url:GET/api/v1/courses/:course\_id/assignments/:assignment\_id/gradeable\_students

A paginated list of students eligible to submit the assignment. The caller must have permission to view grades.

If anonymous grading is enabled for the current assignment and the allow\_new\_anonymous\_id parameter is passed, the returned data will not include any values identifying the student, but will instead include an assignment-specific anonymous ID for each student.

Section-limited instructors will only see students in their own sections.

Returns a list of [UserDisplay](#) objects

List multiple assignments gradeable students SubmissionsApiController#multiple\_gradeable\_students

GET /api/v1/courses/:course\_id/assignments/gradeable\_students

Scope: url:GET/api/v1/courses/:course\_id/assignments/gradeable\_students

A paginated list of students eligible to submit a list of assignments. The caller must have permission to view grades for the requested course.

Section-limited instructors will only see students in their own sections.

Request Parameters:

Parameter	Type	Description
assignment_ids[]	string	Assignments being requested

Example Response:

A [UserDisplay](#) with an extra assignment\_ids field to indicate what assignments that user can submit

```
[
  {
    "id": 2,
    "display_name": "Display Name",
    "avatar_image_url": "http://avatar-image-url.jpeg",
    "html_url": "http://canvas.com",
    "assignment_ids": [1, 2, 3]
  }
]
```

Grade or comment on multiple submissions SubmissionsApiController#bulk\_update

POST /api/v1/courses/:course\_id/submissions/update\_grades

Scope: url:POST/api/v1/courses/:course\_id/submissions/update\_grades

POST /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/update\_grades

Scope: url:POST/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/update\_grades

POST /api/v1/sections/:section\_id/submissions/update\_grades

Scope: url:POST/api/v1/sections/:section\_id/submissions/update\_grades

POST /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/update\_grades

Scope: url:POST/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/update\_grades

Update the grading and comments on multiple student's assignment submissions in an asynchronous job.

The user must have permission to manage grades in the appropriate context (course or section).

Request Parameters:

Parameter	Type	Description
grade_data[<student_id>][posted_grade]	string	See documentation for the posted_grade argument in the <a href="#">Submissions Update</a> documentation
grade_data[<student_id>][excuse]	boolean	See documentation for the excuse argument in the <a href="#">Submissions Update</a> documentation
grade_data[<student_id>][rubric_assessment]	RubricAssessment	See documentation for the rubric_assessment argument in the <a href="#">Submissions Update</a> documentation
grade_data[<student_id>][text_comment]	string	no description
grade_data[<student_id>][group_comment]	boolean	no description
grade_data[<student_id>][media_comment_id]	string	no description

Parameter	Type	Description
grade_data[<student_id>][media_comment_type]	string	no description Allowed values: audio, video
grade_data[<student_id>][file_ids[]]	integer	See documentation for the comment[] arguments in the <a href="#">Submissions Update</a> documentation
grade_data[<assignment_id>][<student_id>]	integer	Specifies which assignment to grade. This argument is not necessary when using the assignment-specific endpoints.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/1/assignments/2/submissions/update_grades' \  
-X POST \  
-F 'grade_data[3][posted_grade]=88' \  
-F 'grade_data[4][posted_grade]=95' \  
-H 'Authorization: Bearer <token>'
```

Returns a [Progress](#) object

[Mark submission as read SubmissionsApiController#mark\\_submission\\_read](#)

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read**

**Scope:** url:PUT[/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read]

**PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read**

**Scope:** url:PUT[/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read]

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/read.json' \  
-X PUT \  
-H 'Authorization: Bearer <token>' \  
-H 'Content-Length: 0'
```

[Mark submission as unread SubmissionsApiController#mark\\_submission\\_unread](#)

**DELETE /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read**

**Scope:** url:DELETE[/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read]

**DELETE /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read**

**Scope:** url:DELETE[/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read]

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/read.json' \  
-X DELETE \  
-H 'Authorization: Bearer <token>'
```

[Mark bulk submissions as read SubmissionsApiController#mark\\_bulk\\_submissions\\_as\\_read](#)

**PUT /api/v1/courses/:course\_id/submissions/bulk\_mark\_read**

**Scope:** url:PUT[/api/v1/courses/:course\_id/submissions/bulk\_mark\_read]

**PUT /api/v1/sections/:section\_id/submissions/bulk\_mark\_read**

**Scope:** url:PUT[/api/v1/sections/:section\_id/submissions/bulk\_mark\_read]

Accepts a string array of submission ids. Loops through and marks each submission as read

On success, the response will be 204 No Content with an empty body.

Request Parameters:

Parameter	Type	Description
submissionIds[]	string	no description

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/submissions/bulk_mark_read.json' \  
-X PUT \  
-H 'Authorization: Bearer <token>' \  
-H 'Content-Length: 0' \  
-F 'submissionIds=[ '88 ']
```

[Mark submission item as read SubmissionsApiController#mark\\_submission\\_item\\_read](#)

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read/:item**

**Scope:** url:PUT[/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/read/:item]

**PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read/:item**

**Scope:** url:PUT[/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/read/:item]

No request fields are necessary.

A submission item can be “grade”, “comment” or “rubric”

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/read/:item.json' \  
-X PUT \  
-H 'Authorization: Bearer <token>' \  
-H 'Content-Length: 0'
```

[Clear unread status for all submissions. SubmissionsApiController#submissions\\_clear\\_unread](#)

**PUT /api/v1/courses/:course\_id/submissions/:user\_id/clear\_unread**

**Scope:** url:PUT[/api/v1/courses/:course\_id/submissions/:user\_id/clear\_unread]

**PUT /api/v1/sections/:section\_id/submissions/:user\_id/clear\_unread**

**Scope:** url:PUT[/api/v1/sections/:section\_id/submissions/:user\_id/clear\_unread]

Site-admin-only endpoint.

No request fields are necessary.

On success, the response will be 204 No Content with an empty body.

Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/submissions/<user_id>/clear_unread.json' \  
-X PUT \  
-H 'Authorization: Bearer <token>'
```

```
-H "Content-Length: 0"
```

### [Get rubric assessments read state SubmissionsApiController#rubric\\_assessments\\_read\\_state](#)

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read**

**Scope:** url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read**

**Scope:** url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read

**GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read**

**Scope:** url:GET|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read

**GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read**

**Scope:** url:GET|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read

Return whether new rubric comments/grading made on a submission have been seen by the student being assessed.

#### Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/rubric_comments/read' \
-H "Authorization: Bearer <token>"

# or

curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/rubric_assessments/read' \
-H "Authorization: Bearer <token>"
```

#### Example Response:

```
{
  "read": false
}
```

### [Mark rubric assessments as read SubmissionsApiController#mark\\_rubric\\_assessments\\_read](#)

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read**

**Scope:** url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read**

**Scope:** url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read

**PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read**

**Scope:** url:PUT|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_comments/read

**PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read**

**Scope:** url:PUT|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/rubric\_assessments/read

Indicate that rubric comments/grading made on a submission have been read by the student being assessed. Only the student who owns the submission can use this endpoint.

NOTE: Rubric assessments will be marked as read automatically when they are viewed in Canvas web.

#### Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/rubric_comments/read' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"

# or

curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/rubric_assessments/read' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"
```

#### Example Response:

```
{
  "read": true
}
```

### [Get document annotations read state SubmissionsApiController#document\\_annotations\\_read\\_state](#)

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read**

**Scope:** url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read

**GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read**

**Scope:** url:GET|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read

Return whether annotations made on a submitted document have been read by the student

#### Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/document_annotations/read' \
-H "Authorization: Bearer <token>"
```

#### Example Response:

```
{
  "read": false
}
```

### [Mark document annotations as read SubmissionsApiController#mark\\_document\\_annotations\\_read](#)

**PUT /api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read**

**Scope:** url:PUT|/api/v1/courses/:course\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read

**PUT /api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read**

**Scope:** url:PUT|/api/v1/sections/:section\_id/assignments/:assignment\_id/submissions/:user\_id/document\_annotations/read

Indicate that annotations made on a submitted document have been read by the student. Only the student who owns the submission can use this endpoint.

NOTE: Document annotations will be marked as read automatically when they are viewed in Canvas web.

#### Example Request:

```
curl 'https://<canvas>/api/v1/courses/<course_id>/assignments/<assignment_id>/submissions/<user_id>/document_annotations/read' \
-X PUT \
-H "Authorization: Bearer <token>" \
-H "Content-Length: 0"
```

#### Example Response:

```
{
  "read": true
}
```

### [Submission Summary SubmissionsApiController#submission\\_summary](#)

**GET /api/v1/courses/:course\_id/assignments/:assignment\_id/submission\_summary**

**Scope:** url:GET|/api/v1/courses/:course\_id/assignments/:assignment\_id/submission\_summary

**GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submission\_summary**

**Scope:** url:GET /api/v1/sections/:section\_id/assignments/:assignment\_id/submission\_summary

Returns the number of submissions for the given assignment based on gradeable students that fall into three categories: graded, ungraded, not submitted.

**Request Parameters:**

Parameter	Type	Description
grouped	boolean	If this argument is true, the response will take into account student groups.

**Example Response:**

```
{
  "graded": 5,
  "ungraded": 10,
  "not_submitted": 42
}
```

[List available tabs for a course or group](#) **TabsController#index**

**GET /api/v1/accounts/:account\_id/tabs**

**Scope:** url:GET /api/v1/accounts/:account\_id/tabs

**GET /api/v1/courses/:course\_id/tabs**

**Scope:** url:GET /api/v1/courses/:course\_id/tabs

**GET /api/v1/groups/:group\_id/tabs**

**Scope:** url:GET /api/v1/groups/:group\_id/tabs

**GET /api/v1/users/:user\_id/tabs**

**Scope:** url:GET /api/v1/users/:user\_id/tabs

Returns a paginated list of navigation tabs available in the current context.

**Request Parameters:**

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>“course_subject_tabs”: Optional flag to return the tabs associated with a canvas_for_elementary subject course’s home page instead of the typical sidebar navigation. Only takes effect if this request is for a course context in a canvas_for_elementary-enabled account or sub-account.</li></ul> Allowed values: course_subject_tabs

**Example Request:**

```
curl -H 'Authorization: Bearer <token>' \
https://<canvas>/api/v1/groups/<group_id>/tabs"
```

**Example Response:**

```
{
  {
    "html_url": "/courses/1",
    "id": "Home",
    "label": "Home",
    "position": 1,
    "visibility": "public",
    "type": "internal"
  },
  {
    "html_url": "/courses/1/external_tools/4",
    "id": "context_external_tool_4",
    "label": "WordPress",
    "hidden": true,
    "visibility": "public",
    "position": 2,
    "type": "external"
  },
  {
    "html_url": "/courses/1/grades",
    "id": "grades",
    "label": "Grades",
    "position": 3,
    "hidden": true
    "visibility": "admins"
    "type": "internal"
  }
}
```

[Update a tab for a course](#) **TabsController#update**

**PUT /api/v1/courses/:course\_id/tabs/:tab\_id**

**Scope:** url:PUT /api/v1/courses/:course\_id/tabs/:tab\_id

Home and Settings tabs are not manageable, and can't be hidden or moved

Returns a tab object

**Request Parameters:**

Parameter	Type	Description
position	integer	The new position of the tab, 1-based
hidden	boolean	no description

**Example Request:**

```
curl https://<canvas>/api/v1/courses/<course_id>/tabs/tab_id \
-X PUT \
-H 'Authorization: Bearer <token>' \
-d 'hidden=true' \
-d 'position=2' // 1 based
```

Returns a [Tab](#) object

[List temporary enrollment pairings](#) **TemporaryEnrollmentPairingsApiController#index**

**GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings**

**Scope:** url:GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings

Returns the list of temporary enrollment pairings for a root account.

Returns a list of [TemporaryEnrollmentPairing](#) objects

[Get a single temporary enrollment pairing](#) **TemporaryEnrollmentPairingsApiController#show**

**GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/:id**

**Scope:** url:GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/:id

Returns the temporary enrollment pairing with the given id.

Returns a [TemporaryEnrollmentPairing](#) object

[New TemporaryEnrollmentPairing](#) **TemporaryEnrollmentPairingsApiController#new**

**GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/new**

**Scope:** url:GET /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/new

Initialize an unsaved Temporary Enrollment Pairing.

Returns a [TemporaryEnrollmentPairing](#) object

Create Temporary Enrollment Pairing [TemporaryEnrollmentPairingsApiController#create](#)

POST /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings

**Scope:** url:POST[/api/v1/accounts/:account\_id/temporary\_enrollment\_pairings]

Create a Temporary Enrollment Pairing.

**Request Parameters:**

Parameter	Type	Description
workflow_state	string	The workflow state of the temporary enrollment pairing.
ending_enrollment_state	string	The ending enrollment state to be given to each associated enrollment when the enrollment period has been reached. Defaults to "deleted" if no value is given. Accepted values are "deleted", "completed", and "inactive". Allowed values: deleted, completed, inactive

Returns a [TemporaryEnrollmentPairing](#) object

Delete Temporary Enrollment Pairing [TemporaryEnrollmentPairingsApiController#destroy](#)

DELETE /api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/:id

**Scope:** url:DELETE[/api/v1/accounts/:account\_id/temporary\_enrollment\_pairings/:id]

Delete a temporary enrollment pairing

Returns a [TemporaryEnrollmentPairing](#) object

List observees [UserObserveesController#index](#)

GET /api/v1/users/:user\_id/observees

**Scope:** url:GET[/api/v1/users/:user\_id/observees]

A paginated list of the users that the given user is observing.

**Note:** all users are allowed to list their own observees. Administrators can list other users' observees.

The returned observees will include an attribute "observation\_link\_root\_account\_ids", a list of ids for the root accounts the observer and observee are linked on. The observer will only be able to observe in courses associated with these root accounts.

**Request Parameters:**

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"avatar_url": Optionally include avatar_url.</li></ul> Allowed values: avatar_url

**Example Request:**

```
curl https://<canvas>/api/v1/users/<user_id>/observees \
-X GET \
-H 'Authorization: Bearer <token>'
```

Returns a list of [User](#) objects

List observers [UserObserveesController#observers](#)

GET /api/v1/users/:user\_id/observers

**Scope:** url:GET[/api/v1/users/:user\_id/observers]

A paginated list of the observers of a given user.

**Note:** all users are allowed to list their own observers. Administrators can list other users' observers.

The returned observers will include an attribute "observation\_link\_root\_account\_ids", a list of ids for the root accounts the observer and observee are linked on. The observer will only be able to observe in courses associated with these root accounts.

**Request Parameters:**

Parameter	Type	Description
include[]	string	<ul style="list-style-type: none"><li>"avatar_url": Optionally include avatar_url.</li></ul> Allowed values: avatar_url

**Example Request:**

```
curl https://<canvas>/api/v1/users/<user_id>/observers \
-X GET \
-H 'Authorization: Bearer <token>'
```

Returns a list of [User](#) objects

Add an observee with credentials [UserObserveesController#create](#)

POST /api/v1/users/:user\_id/observees

**Scope:** url:POST[/api/v1/users/:user\_id/observees]

Register the given user to observe another user, given the observee's credentials.

**Note:** all users are allowed to add their own observees, given the observee's credentials or access token are provided. Administrators can add observees given credentials, access token or the [observee's id](#).

**Request Parameters:**

Parameter	Type	Description
observee[unique_id]	string	The login id for the user to observe. Required if access_token is omitted.
observee[password]	string	The password for the user to observe. Required if access_token is omitted.
access_token	string	The access token for the user to observe. Required if observee[unique_id] OR observee[password] are omitted.
pairing_code	string	A generated pairing code for the user to observe. Required if the Observer pairing code feature flag is enabled
root_account_id	integer	The ID for the root account to associate with the observation link. Defaults to the current domain account. If 'all' is specified, a link will be created for each root account associated to both the observer and observee.

**Example Request:**

```
curl https://<canvas>/api/v1/users/<user_id>/observees \
-X POST \
-H 'Authorization: Bearer <token>' \
-F 'observee[unique_id]=UNIQUE_ID' \
-F 'observee[password]=PASSWORD'
```

Returns an [User](#) object

Show an observee UserObserveesController#show

GET /api/v1/users/:user\_id/observees/:observee\_id

Scope: url:GET|/api/v1/users/:user\_id/observees/:observee\_id

Gets information about an observed user.

Note: all users are allowed to view their own observees.

Example Request:

```
curl https://<canvas>/api/v1/users/<user_id>/observees/<observee_id> \
-X GET \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

Show an observer UserObserveesController#show\_observer

GET /api/v1/users/:user\_id/observers/:observer\_id

Scope: url:GET|/api/v1/users/:user\_id/observers/:observer\_id

Gets information about an observer.

Note: all users are allowed to view their own observers.

Example Request:

```
curl https://<canvas>/api/v1/users/<user_id>/observers/<observer_id> \
-X GET \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

Add an observee UserObserveesController#update

PUT /api/v1/users/:user\_id/observees/:observee\_id

Scope: url:PUT|/api/v1/users/:user\_id/observees/:observee\_id

Registers a user as being observed by the given user.

Request Parameters:

Parameter	Type	Description
root_account_id	integer	The ID for the root account to associate with the observation link. If not specified, a link will be created for each root account associated to both the observer and observee.

Example Request:

```
curl https://<canvas>/api/v1/users/<user_id>/observees/<observee_id> \
-X PUT \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

Remove an observee UserObserveesController#destroy

DELETE /api/v1/users/:user\_id/observees/:observee\_id

Scope: url:DELETE|/api/v1/users/:user\_id/observees/:observee\_id

Unregisters a user as being observed by the given user.

Request Parameters:

Parameter	Type	Description
root_account_id	integer	If specified, only removes the link for the given root account

Example Request:

```
curl https://<canvas>/api/v1/users/<user_id>/observees/<observee_id> \
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

Create observer pairing code ObserverPairingCodesApiController#create

POST /api/v1/users/:user\_id/observer\_pairing\_codes

Scope: url:POST|/api/v1/users/:user\_id/observer\_pairing\_codes

If the user is a student, will generate a code to be used with self registration or observees APIs to link another user to this student.

Returns a [PairingCode](#) object

List users in account UsersController#api\_index

GET /api/v1/accounts/:account\_id/users

Scope: url:GET|/api/v1/accounts/:account\_id/users

A paginated list of users associated with this account.

```
@example_request
curl https://<canvas>/api/v1/accounts/self/users?search_term=<search value> \
-X GET \
-H 'Authorization: Bearer <token>'
```

Request Parameters:

Parameter	Type	Description
search_term	string	The partial name or full ID of the users to match and return in the results list. Must be at least 3 characters.
		Note that the API will prefer matching on canonical user ID if the ID has a numeric form. It will only search against other fields if non-numeric in form, or if the numeric value doesn't yield any matches. Queries by administrative users will search on SIS ID, Integration ID, login ID, name, or email address
enrollment_type	string	When set, only return users enrolled with the specified course-level base role. This can be a base role type of 'student', 'teacher', 'ta', 'observer', or 'designer'.
sort	string	The column to sort results by.
		Allowed values: username, email, sis_id, integration_id, last_login
order	string	The order to sort the given column by.
		Allowed values: asc, desc
include_deleted_users	boolean	When set to true and used with an account context, returns users who have deleted pseudonyms for the context

Returns a list of [User](#) objects

List the activity stream UsersController#activity\_stream

GET /api/v1/users/self/activity\_stream

**Scope:** url:GET|/api/v1/users/self/activity\_stream

**GET /api/v1/users/activity\_stream**

**Scope:** url:GET|/api/v1/users/activity\_stream

Returns the current user's global activity stream, paginated.

There are many types of objects that can be returned in the activity stream. All object types have the same basic set of shared attributes:

```
{
  'created_at': '2011-07-13T09:12:00Z',
  'updated_at': '2011-07-25T00:52:41Z',
  'id': 1234,
  'title': 'Stream Item Subject',
  'message': 'This is the body text of the activity stream item. It is plain-text, and can be multiple paragraphs.',
  'type': 'DiscussionTopic|Conversation|Message|Submission|Conference|Collaboration|AssessmentRequest...',
  'read_state': false,
  'context_type': 'course', // course|group
  'course_id': 1,
  'group_id': null,
  'html_url': 'http://...' // URL to the Canvas web UI for this stream item
}
```

In addition, each item type has its own set of attributes available.

DiscussionTopic:

```
{
  'type': 'DiscussionTopic',
  'discussion_topic_id': 1234,
  'total_root_discussion_entries': 5,
  'require_initial_post': true,
  'user_has_posted': true,
  'root_discussion_entries': {
    ...
  }
}
```

For DiscussionTopic, the message is truncated at 4kb.

Announcement:

```
{
  'type': 'Announcement',
  'announcement_id': 1234,
  'total_root_discussion_entries': 5,
  'require_initial_post': true,
  'user_has_posted': null,
  'root_discussion_entries': {
    ...
  }
}
```

For Announcement, the message is truncated at 4kb.

Conversation:

```
{
  'type': 'Conversation',
  'conversation_id': 1234,
  'private': false,
  'participant_count': 3,
}
```

Message:

```
{
  'type': 'Message',
  'message_id': 1234,
  'notification_category': 'Assignment Graded'
}
```

Submission:

Returns an [Submission](#) with its Course and Assignment data.

Conference:

```
{
  'type': 'Conference',
  'web_conference_id': 1234
}
```

Collaboration:

```
{
  'type': 'Collaboration',
  'collaboration_id': 1234
}
```

AssessmentRequest:

```
{
  'type': 'AssessmentRequest',
  'assessment_request_id': 1234
}
```

**Request Parameters:**

Parameter	Type	Description
only_active_courses	boolean	If true, will only return objects for courses the user is actively participating in

[Activity stream summary UsersController#activity\\_stream\\_summary](#)

**GET /api/v1/users/self/activity\_stream/summary**

**Scope:** url:GET|/api/v1/users/self/activity\_stream/summary

Returns a summary of the current user's global activity stream.

**Request Parameters:**

Parameter	Type	Description
only_active_courses	boolean	If true, will only return objects for courses the user is actively participating in

**Example Response:**

```
{
  {
    "type": "DiscussionTopic",
    "unread_count": 2,
    "count": 7
  },
  {
    "type": "Conversation",
    "unread_count": 0,
    "count": 3
  }
}
```

[List the TODO items UsersController#todo\\_items](#)

**GET /api/v1/users/self/todo**

**Scope:** url:GET|/api/v1/users/self/todo

A paginated list of the current user's list of todo items.

There is a limit to the number of items returned.

The 'ignore' and 'ignore\_permanently' URLs can be used to update the user's preferences on what items will be displayed. Performing a DELETE request against the 'ignore' URL will hide that item from future todo item requests, until the item changes. Performing a DELETE request against the 'ignore\_permanently' URL will hide that item forever.

**Request Parameters:**

Parameter	Type	Description
include[]	string	"ungraded_quizzes"

Optionally include ungraded quizzes (such as practice quizzes and surveys) in the list. These will be returned under a `quiz` key instead of an `assignment` key in response elements.

Parameter	Type	Description
		Allowed values: ungraded_quizzes

Example Response:

```
{
  {
    'type': 'grading', // an assignment that needs grading
    'assignment': { .. assignment object .. },
    'ignore': '.. url ..',
    'ignore_permanently': '.. url ..',
    'html_url': '.. url ..',
    'needs_grading_count': 3, // number of submissions that need grading
    'context_type': 'course', // course/group
    'course_id': 1,
    'group_id': null,
  },
  {
    'type' => 'submitting', // an assignment that needs submitting soon
    'assignment' => { .. assignment object .. },
    'ignore' => '.. url ..',
    'ignore_permanently' => '.. url ..',
    'html_url': '.. url ..',
    'context_type': 'course',
    'course_id': 1,
  },
  {
    'type' => 'submitting', // a quiz that needs submitting soon
    'quiz' => { .. quiz object .. },
    'ignore' => '.. url ..',
    'ignore_permanently' => '.. url ..',
    'html_url': '.. url ..',
    'context_type': 'course',
    'course_id': 1,
  },
}
```

List counts for todo items UsersController#todo\_item\_count

GET /api/v1/users/self/todo\_item\_count

Scope: url:GET|api/v1/users/self/todo\_item\_count

Counts of different todo items such as the number of assignments needing grading as well as the number of assignments needing submitting.

There is a limit to the number of todo items this endpoint will count. It will only look at the first 100 todo items for the user. If the user has more than 100 todo items this count may not be reliable. The largest reliable number for both counts is 100.

Request Parameters:

Parameter	Type	Description
		"ungraded_quizzes"
include[]	string	Optionally include ungraded quizzes (such as practice quizzes and surveys) in the list. These will be returned under a quiz key instead of an assignment key in response elements.
		Allowed values: ungraded_quizzes

Example Response:

```
{
  needs_grading_count: 32,
  assignments_needing_submitting: 10
}
```

List upcoming assignments, calendar events UsersController#upcoming\_events

GET /api/v1/users/self/upcoming\_events

Scope: url:GET|api/v1/users/self/upcoming\_events

A paginated list of the current user's upcoming events.

Example Response:

```
{
  {
    "id"=>597,
    "title"=>"Upcoming Course Event",
    "description"=>"Attendance is correlated with passing!",
    "start_at"=>"2013-04-27T14:33:14Z",
    "end_at"=>"2013-04-27T14:33:14Z",
    "location_name"=>"Red brick house",
    "location_address"=>"110 Top of the Hill Dr.",
    "all_day"=>false,
    "all_day_date"=>nil,
    "created_at"=>"2013-04-26T14:33:14Z",
    "updated_at"=>"2013-04-26T14:33:14Z",
    "workflow_state"=>"active",
    "context_code"=>"course_12938",
    "child_events_count"=>0,
    "child_events"=>[],
    "parent_event_id"=>nil,
    "hidden"=>false,
    "url"=>"http://www.example.com/api/v1/calendar_events/597",
    "html_url"=>"http://www.example.com/calendar/evemt_id=597&include_contexts=course_12938"
  },
  {
    "id"=>"assignment_9729",
    "title"=>"Upcoming Assignment",
    "description"=>nil,
    "start_at"=>"2013-04-28T14:47:32Z",
    "end_at"=>"2013-04-28T14:47:32Z",
    "all_day"=>false,
    "all_day_date"=>"2013-04-28",
    "created_at"=>"2013-04-28T14:47:32Z",
    "updated_at"=>"2013-04-26T14:47:32Z",
    "workflow_state"=>"published",
    "context_code"=>"course_12942",
    "assignment"=>{
      "id"=>9729,
      "name"=>"Upcoming Assignment",
      "description"=>nil,
      "points_possible"=>10,
      "due_at"=>"2013-04-28T14:47:32Z",
      "assignment_group_id"=>2439,
      "automatic_peer_reviews"=>false,
      "grade_group_students_individually"=>nil,
      "grading_standard_id"=>nil,
      "grading_type"=>"points",
      "group_category_id"=>nil,
      "lock_at"=>nil,
      "peer_reviews"=>false,
      "position"=>1,
      "unlock_at"=>nil,
      "course_id"=>12942,
      "submission_types"=>["none"],
      "needs_grading_count"=>0,
      "html_url"=>"http://www.example.com/courses/12942/assignments/9729"
    },
    "url"=>"http://www.example.com/api/v1/calendar_events/assignment_9729",
    "html_url"=>"http://www.example.com/courses/12942/assignments/9729"
  }
}
```

List Missing Submissions UsersController#missing\_submissions

GET /api/v1/users/user\_id/missing\_submissions

Scope: url:GET|api/v1/users/user\_id/missing\_submissions

A paginated list of past-due assignments for which the student does not have a submission. The user sending the request must either be the student, an admin or a parent observer using the parent app

Request Parameters:

Parameter	Type	Description
user_id	string	the student's ID
observed_user_id	string	Return missing submissions for the given observed user. Must be accompanied by course_ids[]. The user making the request must be observing the observed user in all the courses specified by course_ids[].
include[]	string	"planner_overrides"

Optionally include the assignment's associated planner override, if it exists, for the current user. These will be returned under a planner\_override key



Parameter	Type	Description
	“course”	
		Optionally include the assignments' courses
		Allowed values: planner_overrides, course
	“submittable”	
		Only return assignments that the current user can submit (i.e. filter out locked assignments)
filter[]	string “current_grading_period”	
		Only return missing assignments that are in the current grading period
		Allowed values: submittable, current_grading_period

course\_ids[] string Optionally restricts the list of past-due assignments to only those associated with the specified course IDs. Required if observed\_user\_id is passed.

Returns a list of [Assignment](#) objects

### Hide a stream item UsersController#ignore\_stream\_item

**DELETE** /api/v1/users/self/activity\_stream/:id

**Scope:** url:DELETE[/api/v1/users/self/activity\_stream/:id]

Hide the given stream item.

**Example Request:**

```
curl https://<canvas>/api/v1/users/self/activity_stream/<stream_item_id> \  
-X DELETE \  
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{  
  "hidden": true  
}
```

### Hide all stream items UsersController#ignore\_all\_stream\_items

**DELETE** /api/v1/users/self/activity\_stream

**Scope:** url:DELETE[/api/v1/users/self/activity\_stream]

Hide all stream items for the user

**Example Request:**

```
curl https://<canvas>/api/v1/users/self/activity_stream \  
-X DELETE \  
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{  
  "hidden": true  
}
```

### Upload a file UsersController#create\_file

**POST** /api/v1/users/:user\_id/files

**Scope:** url:POST[/api/v1/users/:user\_id/files]

Upload a file to the user's personal files section.

This API endpoint is the first step in uploading a file to a user's files. See the [File Upload Documentation](#) for details on the file upload workflow.

Note that typically users will only be able to upload files to their own files section. Passing a user\_id of self is an easy shortcut to specify the current user.

### Show user details UsersController#api\_show

**GET** /api/v1/users/:id

**Scope:** url:GET[/api/v1/users/:id]

Shows details for user.

Also includes an attribute “permissions”, a non-comprehensive list of permissions for the user. Example:

```
"permissions": {  
  "can_update_name": true, // Whether the user can update their name.  
  "can_update_avatar": false, // Whether the user can update their avatar.  
  "limit_parent_app_web_access": false // Whether the user can interact with Canvas web from the Canvas Parent app.  
}
```

**Request Parameters:**

Parameter	Type	Description
include[]	string	Array of additional information to include on the user record. “locale”, “avatar_url”, “permissions”, “email”, and “effective_locale” will always be returned
		Allowed values: uuid, last_login

**Example Request:**

```
curl https://<canvas>/api/v1/users/self \  
-X GET \  
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

### Create a user UsersController#create

**POST** /api/v1/accounts/:account\_id/users

**Scope:** url:POST[/api/v1/accounts/:account\_id/users]

Create and return a new user and pseudonym for an account.

DEPRECATED (for self-registration only)

If you don't have the “Modify

login details for users” permission, but self-registration is enabled on the account, you can still use this endpoint to register new users. Certain fields will be required, and others will be ignored (see below).

**Request Parameters:**

Parameter	Type	Description
user[name]	string	The full name of the user. This name will be used by teacher for grading. Required if this is a self-registration.
user[short_name]	string	User's name as it will be displayed in discussions, messages, and comments.
user[sortable_name]	string	User's name as used to sort alphabetically in lists.
user[time_zone]	string	The time zone for the user. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .

Parameter	Type	Description
user[locale]	string	The user's preferred language, from the list of languages Canvas supports. This is in RFC-5646 format.
user[terms_of_use]	boolean	Whether the user accepts the terms of use. Required if this is a self-registration and this canvas instance requires users to accept the terms (on by default). If this is true, it will mark the user as having accepted the terms of use.
user[skip_registration]	boolean	Automatically mark the user as registered. If this is true, it is recommended to set "pseudonym[send_confirmation]" to true as well. Otherwise, the user will not receive any messages about their account creation. The users communication channel confirmation can be skipped by setting "communication_channel[skip_confirmation]" to true as well.
pseudonym[unique_id]	Required string	User's login ID. If this is a self-registration, it must be a valid email address.
pseudonym[password]	string	User's password. Cannot be set during self-registration.
pseudonym[sis_user_id]	string	SIS ID for the user's account. To set this parameter, the caller must be able to manage SIS permissions.
pseudonym[integration_id]	string	Integration ID for the login. To set this parameter, the caller must be able to manage SIS permissions. The Integration ID is a secondary identifier useful for more complex SIS integrations.
pseudonym[send_confirmation]	boolean	Send user notification of account creation if true. Automatically set to true during self-registration.
pseudonym[force_self_registration]	boolean	Send user a self-registration style email if true. Setting it means the users will get a notification asking them to "complete the registration process" by clicking it, setting a password, and letting them in. Will only be executed on if the user does not need admin approval. Defaults to false unless explicitly provided.
pseudonym[authentication_provider_id]	string	The authentication provider this login is associated with. Logins associated with a specific provider can only be used with that provider. Legacy providers (LDAP, CAS, SAML) will search for logins associated with them, or unassociated logins. New providers will only search for logins explicitly associated with them. This can be the integer ID of the provider, or the type of the provider (in which case, it will find the first matching provider).
communication_channel[type]	string	The communication channel type, e.g. 'email' or 'sms'.
communication_channel[address]	string	The communication channel address, e.g. the user's email address.
communication_channel[confirmation_url]	boolean	Only valid for account admins. If true, returns the new user account confirmation URL, in the response.
communication_channel[skip_confirmation]	boolean	Only valid for site admins and account admins making requests; If true, the channel is automatically validated and no confirmation email or SMS is sent. Otherwise, the user must respond to a confirmation message to confirm the channel. If this is true, it is recommended to set "pseudonym[send_confirmation]" to true as well. Otherwise, the user will not receive any messages about their account creation.
force_validations	boolean	If true, validations are performed on the newly created user (and their associated pseudonym) even if the request is made by a privileged user like an admin. When set to false, or not included in the request parameters, any newly created users are subject to validations unless the request is made by a user with a 'manage_user_logins' right. In which case, certain validations such as 'require_acceptance_of_terms' and 'require_presence_of_name' are not enforced. Use this parameter to return helpful json errors while building users with an admin request.
enable_sis_reactivation	boolean	When true, will first try to re-activate a deleted user with matching sis_user_id if possible. This is commonly done with <a href="#">user</a> and <a href="#">communication_channel</a> so that the default communication_channel is also restored.
destination	URL	If you're setting the password for the newly created user, you can provide this param with a valid URL pointing into this Canvas installation, and the response will include a destination field that's a URL that you can redirect a browser to and have the newly created user automatically logged in. The URL is only valid for a short time, and must match the domain this request is directed to, and be for a well-formed path that Canvas can recognize.
initial_enrollment_type	string	'observer' if doing a self-registration with a pairing code. This allows setting the password during user creation.
pairing_code[code]	string	If provided and valid, will link the new user as an observer to the student's whose pairing code is given.

Returns an [User](#) object

**[\[DEPRECATED\] Self register a user UsersController#create\\_self\\_registered\\_user](#)**

**POST /api/v1/accounts/account\_id/self\_registration**

**Scope:** url:POST|/api/v1/accounts/:account\_id/self\_registration

Self register and return a new user and pseudonym for an account.

If self-registration is enabled on the account, you can use this endpoint to self register new users.

**Request Parameters:**

Parameter	Type	Description
user[name]	Required string	The full name of the user. This name will be used by teacher for grading.
user[short_name]	string	User's name as it will be displayed in discussions, messages, and comments.
user[sortable_name]	string	User's name as used to sort alphabetically in lists.
user[time_zone]	string	The time zone for the user. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
user[locale]	string	The user's preferred language, from the list of languages Canvas supports. This is in RFC-5646 format.
user[terms_of_use]	Required boolean	Whether the user accepts the terms of use.
pseudonym[unique_id]	Required string	User's login ID. Must be a valid email address.
communication_channel[type]	string	The communication channel type, e.g. 'email' or 'sms'.
communication_channel[address]	string	The communication channel address, e.g. the user's email address.

Returns an [User](#) object

**[Update user settings. UsersController#settings](#)**

**GET /api/v1/users/:id/settings**

**Scope:** url:GET|/api/v1/users/:id/settings

**PUT /api/v1/users/:id/settings**

**Scope:** url:PUT|/api/v1/users/:id/settings

Update an existing user's settings.

**Request Parameters:**

Parameter	Type	Description
manual_mark_as_read	boolean	If true, require user to manually mark discussion posts as read (don't auto-mark as read).
release_notes_badge_disabled	boolean	If true, hide the badge for new release notes.
collapse_global_nav	boolean	If true, the user's page loads with the global navigation collapsed
collapse_course_nav	boolean	If true, the user's course pages will load with the course navigation collapsed.
hide_dashcard_color_overlays	boolean	If true, images on course cards will be presented without being tinted to match the course color.
comment_library_suggestions_enabled	boolean	If true, suggestions within the comment library will be shown.
elementary_dashboard_disabled	boolean	If true, will display the user's preferred class Canvas dashboard view instead of the canvas for elementary view.

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/settings \
-X PUT \
-F 'manual_mark_as_read=true'
-H 'Authorization: Bearer <token>'
```

Get custom colors UsersController#get\_custom\_colors

GET /api/v1/users/:id/colors

Scope: url:GET|api/v1/users/:id/colors

Returns all custom colors that have been saved for a user.

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/colors/ \
-X GET \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "custom_colors": {
    "course_42": "#abc123",
    "course_88": "#123abc"
  }
}
```

Get custom color UsersController#get\_custom\_color

GET /api/v1/users/:id/colors/:asset\_string

Scope: url:GET|api/v1/users/:id/colors/:asset\_string

Returns the custom colors that have been saved for a user for a given context.

The asset\_string parameter should be in the format 'context\_id', for example 'course\_42'.

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/colors/<asset_string> \
-X GET \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "hexcode": "#abc123"
}
```

Update custom color UsersController#set\_custom\_color

PUT /api/v1/users/:id/colors/:asset\_string

Scope: url:PUT|api/v1/users/:id/colors/:asset\_string

Updates a custom color for a user for a given context. This allows colors for the calendar and elsewhere to be customized on a user basis.

The asset string parameter should be in the format 'context\_id', for example 'course\_42'

Request Parameters:

Parameter	Type	Description
hexcode	string	The hexcode of the color to set for the context, if you choose to pass the hexcode as a query parameter rather than in the request body you should NOT include the '#' unless you escape it first.

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/colors/<asset_string> \
-X PUT \
-F 'hexcode=fffees'
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "hexcode": "#abc123"
}
```

Update text editor preference UsersController#set\_text\_editor\_preference

PUT /api/v1/users/:id/text\_editor\_preference

Scope: url:PUT|api/v1/users/:id/text\_editor\_preference

Updates a user's default choice for text editor. This allows the Choose an Editor prompts to preload the user's preference.

Request Parameters:

Parameter	Type	Description
		The identifier for the editor.
text_editor_preference	string	Allowed values: block_editor, rce,

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/prefere_editor \
-X PUT \
-F 'text_editor_preference=rce'
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "text_editor_preference": "rce"
}
```

Get dashboard positions UsersController#get\_dashboard\_positions

GET /api/v1/users/:id/dashboard\_positions

**Scope:** url:GET/api/v1/users/:id/dashboard\_positions

Returns all dashboard positions that have been saved for a user.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/<user_id>/dashboard_positions/' \
-X GET \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "dashboard_positions": {
    "course_42": 2,
    "course_88": 1
  }
}
```

[Update dashboard positions UsersController#set\\_dashboard\\_positions](#)

**PUT /api/v1/users/:id/dashboard\_positions**

**Scope:** url:PUT/api/v1/users/:id/dashboard\_positions

Updates the dashboard positions for a user for a given context. This allows positions for the dashboard cards and elsewhere to be customized on a per user basis.

The asset string parameter should be in the format 'context\_id', for example 'course\_42'

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/<user_id>/dashboard_positions/' \
-X PUT \
-F 'dashboard_positions[course_42]=1' \
-F 'dashboard_positions[course_53]=2' \
-F 'dashboard_positions[course_10]=3' \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "dashboard_positions": {
    "course_10": 3,
    "course_42": 1,
    "course_53": 2
  }
}
```

[Edit a user UsersController#update](#)

**PUT /api/v1/users/:id**

**Scope:** url:PUT/api/v1/users/:id

Modify an existing user. To modify a user's login, see the documentation for logins.

**Request Parameters:**

Parameter	Type	Description
user[name]	string	The full name of the user. This name will be used by teacher for grading.
user[short_name]	string	User's name as it will be displayed in discussions, messages, and comments.
user[sortable_name]	string	User's name as used to sort alphabetically in lists.
user[time_zone]	string	The time zone for the user. Allowed time zones are <a href="#">IANA time zones</a> or friendlier <a href="#">Ruby on Rails time zones</a> .
user[email]	string	The default email address of the user.
user[locale]	string	The user's preferred language, from the list of languages Canvas supports. This is in RFC-5646 format.
user[avatar][token]	string	A unique representation of the avatar record to assign as the user's current avatar. This token can be obtained from the user avatars endpoint. This supersedes the user [avatar] [url] argument, and if both are included the url will be ignored. Note: this is an internal representation and is subject to change without notice. It should be consumed with this api endpoint and used in the user update endpoint, and should not be constructed by the client.
user[avatar][url]	string	To set the user's avatar to point to an external url, do not include a token and instead pass the url here. Warning: For maximum compatibility, please use 128 px square images.
user[avatar][state]	string	To set the state of user's avatar. Only valid for account administrator. Allowed values: none, submitted, approved, locked, reported, re_reported
user[title]	string	Sets a title on the user profile. (See <a href="#">Get user profile</a> .) Profiles must be enabled on the root account.
user[bio]	string	Sets a bio on the user profile. (See <a href="#">Get user profile</a> .) Profiles must be enabled on the root account.
user[pronunciation]	string	Sets name pronunciation on the user profile. (See <a href="#">Get user profile</a> .) Profiles and name pronunciation must be enabled on the root account.
user[pronouns]	string	Sets pronouns on the user profile. Passing an empty string will empty the user's pronouns Only Available Pronouns set on the root account are allowed Adding and changing pronouns must be enabled on the root account.
user[event]	string	Suspends or unsuspends all logins for this user that the calling user has permission to Allowed values: suspend, unsuspend

override\_sis\_stickiness boolean Default is true. If false, any fields containing "sticky" changes will not be updated. See SIS CSV Format documentation for information on which fields can have SIS stickiness

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/133' \
-X PUT \
-F 'user[name]=Sheldon Cooper' \
-F 'user[short_name]=Shelly' \
-F 'user[time_zone]=Pacific Time (US & Canada)' \
-F 'user[avatar][token]=opaque_token' \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

[Terminate all user sessions UsersController#terminate\\_sessions](#)

**DELETE /api/v1/users/:id/sessions**

**Scope:** url:DELETE/api/v1/users/:id/sessions

Terminates all sessions for a user. This includes all browser-based sessions and all access tokens, including manually generated ones. The user can immediately re-authenticate to access Canvas again if they have the current credentials. All integrations will need to be re-authorized.

[Log users out of all mobile apps UsersController#expire\\_mobile\\_sessions](#)

**DELETE /api/v1/users/mobile\_sessions**

**Scope:** url:DELETE/api/v1/users/mobile\_sessions

**DELETE /api/v1/users/:id/mobile\_sessions**

**Scope:** url:DELETE[/api/v1/users/:id/mobile\_sessions]

Permanently expires any active mobile sessions, forcing them to re-authorize.

The route that takes a user id will expire mobile sessions for that user. The route that doesn't take a user id will expire mobile sessions for **all** users in the institution.

**Merge user into another user UsersController#merge\_into**

**PUT /api/v1/users/id/merge\_into/destination\_user\_id**

**Scope:** url:PUT[/api/v1/users/:id/merge\_into/:destination\_user\_id]

**PUT /api/v1/users/id/merge\_into/accounts/:destination\_account\_id/users/:destination\_user\_id**

**Scope:** url:PUT[/api/v1/users/:id/merge\_into/accounts/:destination\_account\_id/users/:destination\_user\_id]

Merge a user into another user. To merge users, the caller must have permissions to manage both users. This should be considered irreversible. This will delete the user and move all the data into the destination user.

User merge details and caveats: The from\_user is the user that was deleted in the user\_merge process. The destination\_user is the user that remains, that is being split.

Avatars: When both users have avatars, only the destination\_users avatar will remain. When one user has an avatar, will it will end up on the destination\_user.

Terms of Use: If either user has accepted terms of use, it will be left as accepted.

Communication Channels: All unique communication channels moved to the destination\_user. All notification preferences are moved to the destination\_user.

Enrollments: All unique enrollments are moved to the destination\_user. When there is an enrollment that would end up making it so that a user would be observing themselves, the enrollment is not moved over. Everything that is tied to the from\_user at the course level relating to the enrollment is also moved to the destination\_user.

Submissions: All submissions are moved to the destination\_user. If there are enrollments for both users in the same course, we prefer submissions that have grades then submissions that have work in them, and if there are no grades or no work, they are not moved.

Other notes: Access Tokens are moved on merge. Conversations are moved on merge. Favorites are moved on merge. Courses will commonly use LTI tools. LTI tools reference the user with IDs that are stored on a user object. Merging users deletes one user and moves all records from the deleted user to the destination\_user. These IDs are kept for all enrollments, group, membership, and account\_users for the from\_user at the time of the merge. When the destination\_user launches an LTI tool from a course that used to be the from\_user's, it doesn't appear as a new user to the tool provider. Instead it will send the stored ids. The destination\_user's LTI IDs remain as they were for the courses that they originally had. Future enrollments for the destination\_user will use the IDs that are on the destination\_user object. LTI IDs that are kept and tracked per context include lti\_context\_id, lti\_id and uuid. APIs that return the LTI ids will return the one for the context that it is called for, except for the user uuid. The user UUID will display the destination\_users uuid, and when getting the uuid from an api that is in a context that was recorded from a merge event, an additional attribute is added as past\_uuid.

When finding users by SIS ids in different accounts the destination\_account\_id is required.

The account can also be identified by passing the domain in destination\_account\_id.

**Example Request:**

```
curl https://<canvas>/api/v1/users/<user_id>/merge_into/<destination_user_id> \
-X PUT \
-H 'Authorization: Bearer <token>'

curl https://<canvas>/api/v1/users/<user_id>/merge_into/accounts/<destination_account_id>/users/<destination_user_id> \
-X PUT \
-H 'Authorization: Bearer <token>'
```

Returns an [User](#) object

**Split merged users into separate users UsersController#split**

**POST /api/v1/users/id/split**

**Scope:** url:POST[/api/v1/users/:id/split]

Merged users cannot be fully restored to their previous state, but this will attempt to split as much as possible to the previous state. To split a merged user, the caller must have permissions to manage all of the users logins. If there are multiple users that have been merged into one user it will split each merge into a separate user. A split can only happen within 180 days of a user merge. A user merge deletes the previous user and may be permanently deleted. In this scenario we create a new user object and proceed to move as much as possible to the new user. The user object will not have preserved the name or settings from the previous user. Some items may have been deleted during a user\_merge that cannot be restored, and/or the data has become stale because of other changes to the objects since the time of the user\_merge.

Split users details and caveats:

The from\_user is the user that was deleted in the user\_merge process. The destination\_user is the user that remains, that is being split.

Avatars: When both users had avatars, both will be remain. When from\_user had an avatar and destination\_user did not have an avatar, the destination\_user's avatar will be deleted if it still matches what was there are the time of the merge. If the destination\_user's avatar was changed at anytime after the merge, it will remain on the destination user. If the from\_user had an avatar it will be there after split.

Terms of Use: If from\_user had not accepted terms of use, they will be prompted again to accept terms of use after the split. If the destination\_user had not accepted terms of use, hey will be prompted again to accept terms of use after the split. If neither user had accepted the terms of use, but since the time of the merge had accepted, both will be prompted to accept terms of use. If both had accepted terms of use, this will remain.

Communication Channels: All communication channels are restored to what they were prior to the merge. If a communication channel was added after the merge, it will remain on the destination\_user. Notification preferences remain with the communication channels.

Enrollments: All enrollments from the time of the merge will be moved back to where they were. Enrollments created since the time of the merge that were created by sis\_import will go to the user that owns that sis\_id used for the import. Other new enrollments will remain on the destination\_user. Everything that is tied to the destination\_user at the course level relating to an enrollment is moved to the from\_user. When both users are in the same course prior to merge this can cause some unexpected items to move.

Submissions: Unlike other items tied to a course, submissions are explicitly recorded to avoid problems with grades. All submissions were moved are restored to the spot prior to merge. All submission that were created in a course that was moved in enrollments are moved over to the from\_user.

Other notes: Access Tokens are moved back on split. Conversations are moved back on split. Favorites that existing at the time of merge are moved back on split. LTI ids are restored to how they were prior to merge.

**Example Request:**

```
curl https://<canvas>/api/v1/users/<user_id>/split \
-X POST \
-H 'Authorization: Bearer <token>'
```

Returns a list of [User](#) objects

**Get a Pandata Events jwt token and its expiration date UsersController#pandata\_events\_token**

**POST /api/v1/users/self/pandata\_events\_token**

**Scope:** url:POST[/api/v1/users/self/pandata\_events\_token]

Returns a jwt auth and props token that can be used to send events to Pandata.

NOTE: This is currently only available to the mobile developer keys.

**Request Parameters:**

Parameter	Type	Description
app_key	string	The pandata events appKey for this mobile app

**Example Request:**

```
curl https://<canvas>/api/v1/users/self/pandata_events_token \
-X POST \
-H 'Authorization: Bearer <token>' \
-F 'app_key=MOBILE_APPS_KEY'
```

**Example Response:**

```
{
  "url": "https://example.com/pandata/events",
  "auth_token": "wek23kl5dnsoieioei3o9f9deo8r8eo8fdn",
  "props_token": "paowinefopwienpfionepfiownepfownef",
  "expires_at": 1521667783000,
}
```

**Get a users most recently graded submissions UsersController#user\_graded\_submissions**

**GET /api/v1/users/id/graded\_submissions**

**Scope:** url:GET[/api/v1/users/:id/graded\_submissions]

**Get user profile ProfileController#settings**

GET /api/v1/users/:user\_id/profile

**Scope:** url:GET|/api/v1/users/:user\_id/profile

Returns user profile data, including user id, name, and profile pic.

When requesting the profile for the user accessing the API, the user's calendar feed URL and LTI user id will be returned as well.

Returns a [Profile](#) object

List avatar options ProfileController#profile\_pics

GET /api/v1/users/:user\_id/avatars

**Scope:** url:GET|/api/v1/users/:user\_id/avatars

A paginated list of the possible user avatar options that can be set with the user update endpoint. The response will be an array of avatar records. If the 'type' field is 'attachment', the record will include all the normal attachment json fields; otherwise it will include only the 'url' and 'display\_name' fields. Additionally, all records will include a 'type' field and a 'token' field. The following explains each field in more detail

type	"gravatar" "attachment" "no_pic"
	The type of avatar record, for categorization purposes.
url	
	The url of the avatar
token	
	A unique representation of the avatar record which can be used to set the avatar with the user update endpoint. Note: this is an internal representation and is subject to change without notice. It should be consumed with this api endpoint and used in the user update endpoint, and should not be constructed by the client.
display_name	
	A textual description of the avatar record
id	
	'attachment' type only
	the internal id of the attachment
content-type	
	'attachment' type only
	the content-type of the attachment
filename	
	'attachment' type only
	the filename of the attachment
size	
	'attachment' type only
	the size of the attachment

Example Request:

```
curl 'https://<canvas>/api/v1/users/1/avatars.json' \  
-H "Authorization: Bearer <token>"
```

Example Response:

```
{  
  {  
    "type": "gravatar",  
    "url": "https://secure.gravatar.com/avatar/2284...",  
    "token": "<opaque_token>",  
    "display_name": "gravatar pic"  
  },  
  {  
    "type": "attachment",  
    "url": "<url to fetch thumbnail of attachment>",  
    "token": "<opaque_token>",  
    "display_name": "profile.jpg",  
    "id": 12,  
    "content-type": "image/jpeg",  
    "filename": "profile.jpg",  
    "size": 32649  
  },  
  {  
    "type": "no_pic",  
    "url": "https://<canvas>/images/dotted_pic.png",  
    "token": "<opaque_token>",  
    "display_name": "no pic"  
  }  
}
```

Returns a list of [Avatar](#) objects

List user page views PageViewsController#index

GET /api/v1/users/:user\_id/page\_views

**Scope:** url:GET|/api/v1/users/:user\_id/page\_views

Return a paginated list of the user's page view history in json format, similar to the available CSV download. Page views are returned in descending order, newest to oldest.

Request Parameters:

Parameter	Type	Description
start_time	DateTime	The beginning of the time range from which you want page views.
end_time	DateTime	The end of the time range from which you want page views.

Returns a list of [PageView](#) objects

Store custom data CustomDataController#set\_data

PUT /api/v1/users/:user\_id/custom\_data(/\*scope)

**Scope:** url:PUT|/api/v1/users/:user\_id/custom\_data(/\*scope)

Store arbitrary user data as JSON.

Arbitrary JSON data can be stored for a User. A typical scenario would be an external site/service that registers users in Canvas and wants to capture additional info about them. The part of the URL that follows /custom\_data/ defines the scope of the request, and it reflects the structure of the JSON data to be stored or retrieved.

The value `self` may be used for `user_id` to store data associated with the calling user. In order to access another user's custom data, you must be an account administrator with permission to manage users.

A namespace parameter, `ns`, is used to prevent custom\_data collisions between different apps. This parameter is required for all custom\_data requests.

A request with Content-Type multipart/form-data or Content-Type application/x-www-form-urlencoded can only be used to store strings.

Example PUT with multipart/form-data data:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/telephone' \  
-X PUT \  
-F 'ns=com.my-organization.canvas-app' \  
-F 'data=555-1234' \  
-H 'Authorization: Bearer <token>'
```

Response:

```
{  
  "data": "555-1234"  
}
```

Subscopes (or, generated scopes) can also be specified by passing values to `data[subscope]`.

Example PUT specifying subscopes:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/body/measurements' \
-X PUT \
-F 'ns=com.my-organization.canvas-app' \
-F 'data[waist]=32in' \
-F 'data[inseam]=34in' \
-F 'data[chest]=40in' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": {
    "chest": "40in",
    "waist": "32in",
    "inseam": "34in"
  }
}
```

Following such a request, subsets of the stored data to be retrieved directly from a subscope.

Example [GET](#) from a generated scope

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/body/measurements/chest' \
-X GET \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": "40in"
}
```

If you want to store more than just strings (i.e. numbers, arrays, hashes, true, false, and/or null), you must make a request with Content-Type application/json as in the following example.

Example PUT with JSON data:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data' \
-H 'Content-Type: application/json' \
-X PUT \
-d '{
  "ns": "com.my-organization.canvas-app",
  "data": {
    "a-number": 6.02e23,
    "a-bool": true,
    "a-string": "true",
    "a-hash": {"a": {"b": "ohai"}},
    "an-array": [1, "two", null, false]
  }
}' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": {
    "a-number": 6.02e+23,
    "a-bool": true,
    "a-string": "true",
    "a-hash": {
      "a": {
        "b": "ohai"
      }
    },
    "an-array": [1, "two", null, false]
  }
}
```

If the data is an Object (as it is in the above example), then subsets of the data can be accessed by including the object's (possibly nested) keys in the scope of a GET request.

Example [GET](#) with a generated scope:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/a-hash/a/b' \
-X GET \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": "ohai"
}
```

On success, this endpoint returns an object containing the data that was stored.

Responds with status code 200 if the scope already contained data, and it was overwritten by the data specified in the request.

Responds with status code 201 if the scope was previously empty, and the data specified in the request was successfully stored there.

Responds with status code 400 if the namespace parameter, `ns`, is missing or invalid, or if the `data` parameter is missing.

Responds with status code 409 if the requested scope caused a conflict and data was not stored. This happens when storing data at the requested scope would cause data at an outer scope to be lost. e.g., if `/custom_data` was `{ "fashion_app": { "hair": "blonde" }}`, but you tried to `PUT /custom_data/fashion_app/hair/style -F data=buzz`, then for the request to succeed, the value of `/custom_data/fashion_app/hair` would have to become a hash, and its old string value would be lost. In this situation, an error object is returned with the following format:

```
{
  "message": "write conflict for custom_data hash",
  "conflict_scope": "fashion_app/hair",
  "type_at_conflict": "String",
  "value_at_conflict": "blonde"
}
```

**Request Parameters:**

Parameter	Type	Description
ns	Required string	The namespace under which to store the data. This should be something other Canvas API apps aren't likely to use, such as a reverse DNS for your organization.
data	Required JSON	The data you want to store for the user, at the specified scope. If the data is composed of (possibly nested) JSON objects, scopes will be generated for the (nested) keys (see examples).

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/food_app' \
-X PUT \
-F 'ns=com.my-organization.canvas-app' \
-F 'data[weight]=81kg' \
-F 'data[favorites][meat]=pork belly' \
-F 'data[favorites][dessert]=pistachio ice cream' \
-H 'Authorization: Bearer <token>'
```

**Example Response:**

```
{
  "data": {
    "weight": "81kg",
    "favorites": {
      "meat": "pork belly",
      "dessert": "pistachio ice cream"
    }
  }
}
```

## Load custom data CustomDataController#get\_data

**GET /api/v1/users/user\_id/custom\_data(/\*scope)**

**Scope:** url: GET /api/v1/users/:user\_id/custom\_data(/\*scope)

Load custom user data.

Arbitrary JSON data can be stored for a User. This API call retrieves that data for a (optional) given scope. See [Store Custom Data](#) for details and examples.

On success, this endpoint returns an object containing the data that was requested.

Responds with status code 400 if the namespace parameter, `ns`, is missing or invalid, or if the specified scope does not contain any data.

**Request Parameters:**

Parameter	Type	Description
ns	Required string	The namespace from which to retrieve the data. This should be something other Canvas API apps aren't likely to use, such as a reverse DNS for your organization.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/food_app/favorites/dessert' \
-X GET \
-F 'ns=com.my-organization.canvas-app' \
```

```
-H 'Authorization: Bearer <token>'
```

Example Response:

```
{
  "data": "pistachio ice cream"
}
```

### Delete custom data CustomDataController#delete\_data

DELETE /api/v1/users/:user\_id/custom\_data(/\*scope)

Scope: url:DELETE /api/v1/users/:user\_id/custom\_data(/\*scope)

Delete custom user data.

Arbitrary JSON data can be stored for a User. This API call deletes that data for a given scope. Without a scope, all custom\_data is deleted. See [Store Custom Data](#) for details and examples of storage and retrieval.

As an example, we'll store some data, then delete a subset of it.

Example [PUT](#) with valid JSON data:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data' \
-X PUT \
-F 'ns=com.my-organization.canvas-app' \
-F 'data[fruit][apple]=so tasty' \
-F 'data[fruit][kiwi]=a bit sour' \
-F 'data[veggies][root][onion]=tear-jerking' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": {
    "fruit": {
      "apple": "so tasty",
      "kiwi": "a bit sour"
    },
    "veggies": {
      "root": {
        "onion": "tear-jerking"
      }
    }
  }
}
```

Example DELETE:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/fruit/kiwi' \
-X DELETE \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": "a bit sour"
}
```

Example [GET](#) following the above DELETE:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data' \
-X GET \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": {
    "fruit": {
      "apple": "so tasty"
    },
    "veggies": {
      "root": {
        "onion": "tear-jerking"
      }
    }
  }
}
```

Note that hashes left empty after a DELETE will get removed from the custom\_data store. For example, following the previous commands, if we delete /custom\_data/veggies/root/onion, then the entire /custom\_data/veggies scope will be removed.

Example DELETE that empties a parent scope:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/veggies/root/onion' \
-X DELETE \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": "tear-jerking"
}
```

Example [GET](#) following the above DELETE:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data' \
-X GET \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": {
    "fruit": {
      "apple": "so tasty"
    }
  }
}
```

On success, this endpoint returns an object containing the data that was deleted.

Responds with status code 400 if the namespace parameter, ns, is missing or invalid, or if the specified scope does not contain any data.

Request Parameters:

Parameter	Type	Description
ns	Required string	The namespace from which to delete the data. This should be something other Canvas API apps aren't likely to use, such as a reverse DNS for your organization.

Example Request:

```
curl 'https://<canvas>/api/v1/users/<user_id>/custom_data/fruit/kiwi' \
-X DELETE \
-F 'ns=com.my-organization.canvas-app' \
-H 'Authorization: Bearer <token>'
```

Example Response:

```
!!!javascript
{
  "data": "a bit sour"
}
```

### List course nicknames CourseNicknamesController#index

GET /api/v1/users/self/course\_nicknames

Scope: url:GET /api/v1/users/self/course\_nicknames

Returns all course nicknames you have set.

Example Request:

```
curl 'https://<canvas>/api/v1/users/self/course_nicknames' \
-H 'Authorization: Bearer <token>'
```

Returns a list of [CourseNickname](#) objects

### Get course nickname CourseNicknamesController#show

GET /api/v1/users/self/course\_nicknames/:course\_id



**Scope:** url:GET|/api/v1/users/self/course\_nicknames/:course\_id

Returns the nickname for a specific course.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/course_nicknames/<course_id> \'
-H 'Authorization: Bearer <token>'
```

Returns a [CourseNickname](#) object

[Set course nickname CourseNicknamesController#update](#)

**PUT /api/v1/users/self/course\_nicknames/:course\_id**

**Scope:** url:PUT|/api/v1/users/self/course\_nicknames/:course\_id

Set a nickname for the given course. This will replace the course's name in output of API calls you make subsequently, as well as in selected places in the Canvas web user interface.

**Request Parameters:**

Parameter	Type	Description
nickname	Required string	The nickname to set. It must be non-empty and shorter than 60 characters.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/course_nicknames/<course_id> \'
-X PUT \
-F 'nickname=Physics' \
-H 'Authorization: Bearer <token>'
```

Returns a [CourseNickname](#) object

[Remove course nickname CourseNicknamesController#delete](#)

**DELETE /api/v1/users/self/course\_nicknames/:course\_id**

**Scope:** url:DELETE|/api/v1/users/self/course\_nicknames/:course\_id

Remove the nickname for the given course. Subsequent course API calls will return the actual name for the course.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/course_nicknames/<course_id> \'
-X DELETE \
-H 'Authorization: Bearer <token>'
```

Returns a [CourseNickname](#) object

[Clear course nicknames CourseNicknamesController#clear](#)

**DELETE /api/v1/users/self/course\_nicknames**

**Scope:** url:DELETE|/api/v1/users/self/course\_nicknames

Remove all stored course nicknames.

**Example Request:**

```
curl 'https://<canvas>/api/v1/users/self/course_nicknames \'
-X DELETE \
-H 'Authorization: Bearer <token>'
```

[Create a Webhook Subscription Lti::SubscriptionsApiController#create](#)

**POST /api/lti/subscriptions**

**Scope:** url:POST|/api/lti/subscriptions

Creates a webbook subscription for the specified event type and context.

**Request Parameters:**

Parameter	Type	Description
subscription[ContextId]	Required string	The id of the context for the subscription.
subscription[ContextType]	Required string	The type of context for the subscription. Must be 'assignment', 'account', or 'course'.
subscription[EventTypes]	Required Array	Array of strings representing the event types for the subscription.
subscription[Format]	Required string	Format to deliver the live events. Must be 'live-event' or 'caliper'.
subscription[TransportMetadata]	Required Object	An object with a single key: 'Url'. Example: { "Url": "sqs.example" }
subscription[TransportType]	Required string	Must be either 'sqs' or 'https'.

[Delete a Webhook Subscription Lti::SubscriptionsApiController#destroy](#)

**DELETE /api/lti/subscriptions/:id**

**Scope:** url:DELETE|/api/lti/subscriptions/:id

[Show a single Webhook Subscription Lti::SubscriptionsApiController#show](#)

**GET /api/lti/subscriptions/:id**

**Scope:** url:GET|/api/lti/subscriptions/:id

[Update a Webhook Subscription Lti::SubscriptionsApiController#update](#)

**PUT /api/lti/subscriptions/:id**

**Scope:** url:PUT|/api/lti/subscriptions/:id

This endpoint uses the same parameters as the create endpoint

[List all Webhook Subscription for a tool proxy Lti::SubscriptionsApiController#index](#)

**GET /api/lti/subscriptions**

**Scope:** url:GET|/api/lti/subscriptions

This endpoint returns a paginated list with a default limit of 100 items per result set. You can retrieve the next result set by setting a 'StartKey' header in your next request with the value of the 'EndKey' header in the response.

Example use of a 'StartKey' header object:

```
{ "Id": "71d6dfba-0547-477d-b41d-db8cb528c6d1", "DeveloperKey": "100000000000001" }
```

[Update a submission's what-if score and calculate grades WhatIfGradesApiController#update](#)

**PUT /api/v1/submissions/:id/what\_if\_grades**

**Scope:** url:PUT|/api/v1/submissions/:id/what\_if\_grades

Enter a what if score for a submission and receive the calculated grades Grade calculation is a costly operation, so this API should be used sparingly

Request Parameters:

Parameter	Type	Description
student_entered_score	number	The score the student wants to test

Example Response:

```
{
  "grades": [
    {
      "current": {
        "grade": 120.0,
        "total": 24.0,
        "possible": 20.0,
        "dropped": []
      },
      "current_groups": {
        "1": {
          "id": 1,
          "global_id": 1000000000000001,
          "score": 20.0,
          "possible": 10.0,
          "weight": 0.0,
          "grade": 200.0,
          "dropped": []
        },
        "3": {
          "id": 3,
          "global_id": 1000000000000003,
          "score": 4.0,
          "possible": 10.0,
          "weight": 0.0,
          "grade": 40.0,
          "dropped": []
        }
      },
      "final": {
        "grade": 21.82,
        "total": 24.0,
        "possible": 110.0,
        "dropped": []
      },
      "final_groups": {
        "1": {
          "id": 1,
          "global_id": 1000000000000001,
          "score": 20.0,
          "possible": 100.0,
          "weight": 0.0,
          "grade": 20.0,
          "dropped": []
        },
        "3": {
          "id": 3,
          "global_id": 1000000000000003,
          "score": 4.0,
          "possible": 10.0,
          "weight": 0.0,
          "grade": 40.0,
          "dropped": []
        }
      }
    }
  ],
  "submission": {
    "id": 166,
    "student_entered_score": 20.0
  }
}
```

Returns a list of [Grades](#) objects

[Reset the what-if scores for the current user for an entire course and recalculate grades WhatIfGradesApiController#reset\\_for\\_student\\_course](#)

PUT /api/v1/courses/:course\_id/what\_if\_grades/reset

**Scope:** url: PUT /api/v1/courses/:course\_id/what\_if\_grades/reset  
© Instructure, Inc. Generated on Fri Feb 7 16:31:46 2025  
This documentation is generated directly from the Canvas LMS source code, available [on Github](#).