

GitHub v3 REST API (Version 1.1.4)

GET - /

GitHub API Root

Responses:

- 200: Response

GET - /advisories

List global security advisories

Parameters:

- ghsa_id (query): If specified, only advisories with this GHSA (GitHub Security Advisory) identifier will be returned.
- type (query): If specified, only advisories of this type will be returned. By default, a request with no other parameters defined will only return reviewed advisories that are not malware.
- cve_id (query): If specified, only advisories with this CVE (Common Vulnerabilities and Exposures) identifier will be returned.
- ecosystem (query): If specified, only advisories for these ecosystems will be returned.
- severity (query): If specified, only advisories with these severities will be returned.
- cwes (query): If specified, only advisories with these Common Weakness Enumerations (CWEs) will be returned. Example: `cwes=79,284,22` or `cwes[]=79&cwes[]=284&cwes[]=22`
- is_withdrawn (query): Whether to only return advisories that have been withdrawn.
- affects (query): If specified, only return advisories that affect any of `package` or `package@version`. A maximum of 1000 packages can be specified. If the query parameter causes the URL to exceed the maximum URL length supported by your client, you must specify fewer packages. Example: `affects=package1,package2@1.0.0,package3@^2.0.0` or `affects[]=package1&affects[]=package2@1.0.0`
- published (query): If specified, only return advisories that were published on a date or date range. For more information on the syntax of the date range, see "[Understanding the search syntax](https://docs.github.com/search-github/getting-started-with-searching-on-github/understanding-the-search-syntax#query-for-dates)."
- updated (query): If specified, only return advisories that were updated on a date or date range. For more information on the syntax of the date range, see "[Understanding the search syntax](https://docs.github.com/search-github/getting-started-with-searching-on-github/understanding-the-search-syntax#query-for-dates)."
- modified (query): If specified, only show advisories that were updated or published on a date or date range. For more information on the syntax of the date range, see "[Understanding the search syntax](https://docs.github.com/search-github/getting-started-with-searching-on-github/understanding-the-search-syntax#query-for-dates)."
- epss_percentage (query): If specified, only return advisories that have an EPSS percentage score that matches the provided value. The EPSS percentage represents the likelihood of a CVE being exploited.
- epss_percentile (query): If specified, only return advisories that have an EPSS percentile score that matches the provided value. The EPSS percentile represents the relative rank of the CVE's likelihood of being exploited compared to other CVEs.
- {}:
- {}:
- {}:
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- sort (query): The property to sort the results by.

Responses:

- 200: Response
- 429: Too many requests
- 422:

GET - /advisories/{ghsa_id}

Get a global security advisory

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /app

Get the authenticated app

Responses:

- 200: Response

POST - /app-manifests/{code}/conversions

Create a GitHub App from a manifest

Parameters:

- code (path):

Responses:

- 201: Response
- 404:
- 422:

GET - /app/hook/config

Get a webhook configuration for an app

Responses:

- 200: Response

PATCH - /app/hook/config

Update a webhook configuration for an app

Responses:

- 200: Response

GET - /app/hook/deliveries

List deliveries for an app webhook

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:

GET - /app/hook/deliveries/{delivery_id}

Get a delivery for an app webhook

Parameters:

- {}:

Responses:

- 200: Response
- 400:
- 422:

POST - /app/hook/deliveries/{delivery_id}/attempts

Redeliver a delivery for an app webhook

Parameters:

- {}:

Responses:

- 202:
- 400:
- 422:

GET - /app/installation-requests

List installation requests for the authenticated app

Parameters:

- {}:
- {}:

Responses:

- 200: List of integration installation requests
- 304:
- 401:

GET - /app/installations

List installations for the authenticated app

Parameters:

- {}:
- {}:
- {}:
- outdated (query):

Responses:

- 200: The permissions the installation has are included under the `permissions` key.

GET - /app/installations/{installation_id}

Get an installation for the authenticated app

Parameters:

- {}:

Responses:

- 200: Response
- 404:

DELETE - /app/installations/{installation_id}

Delete an installation for the authenticated app

Parameters:

- {}:

Responses:

- 204: Response
- 404:

POST - /app/installations/{installation_id}/access_tokens

Create an installation access token for an app

Parameters:

- {}:

Responses:

- 201: Response
- 403:
- 401:
- 404:
- 422:

PUT - /app/installations/{installation_id}/suspended

Suspend an app installation

Parameters:

- {}:

Responses:

- 204: Response
- 404:

DELETE - /app/installations/{installation_id}/suspended

Unsuspend an app installation

Parameters:

- {}:

Responses:

- 204: Response
- 404:

DELETE - /applications/{client_id}/grant

Delete an app authorization

Parameters:

- {}:

Responses:

- 204: Response
- 422:

POST - /applications/{client_id}/token

Check a token

Parameters:

- {}:

Responses:

- 200: Response
- 422:
- 404:

PATCH - /applications/{client_id}/token

Reset a token

Parameters:

- {}:

Responses:

- 200: Response
- 422:

DELETE - /applications/{client_id}/token

Delete an app token

Parameters:

- {}:

Responses:

- 204: Response
- 422:

POST - /applications/{client_id}/token/scoped

Create a scoped access token

Parameters:

- {}:

Responses:

- 200: Response
- 401:
- 403:
- 404:
- 422:

GET - /apps/{app_slug}

Get an app

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:

GET - /assignments/{assignment_id}

Get an assignment

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /assignments/{assignment_id}/accepted_assignments

List accepted assignments for an assignment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /assignments/{assignment_id}/grades

Get assignment grades

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /classrooms

List classrooms

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /classrooms/{classroom_id}

Get a classroom

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /classrooms/{classroom_id}/assignments

List assignments for a classroom

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /codes_of_conduct

Get all codes of conduct

Responses:

- 200: Response
- 304:

GET - /codes_of_conduct/{key}

Get a code of conduct

Parameters:

- key (path):

Responses:

- 200: Response
- 404:
- 304:

GET - /emojis

Get emojis

Responses:

- 200: Response

- 304:

GET - /enterprises/{enterprise}/code-security/configurations

Get code security configurations for an enterprise

Parameters:

- {}:
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

POST - /enterprises/{enterprise}/code-security/configurations

Create a code security configuration for an enterprise

Parameters:

- {}:

Responses:

- 201: Successfully created code security configuration
- 400:
- 403:
- 404:

GET - /enterprises/{enterprise}/code-security/configurations/defaults

Get default code security configurations for an enterprise

Parameters:

- {}:

Responses:

- 200: Response

GET - /enterprises/{enterprise}/code-security/configurations/{configuration_id}

Retrieve a code security configuration of an enterprise

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:

PATCH - /enterprises/{enterprise}/code-security/configurations/{configuration_id}

Update a custom code security configuration for an enterprise

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:
- 409:

DELETE - /enterprises/{enterprise}/code-security/configurations/{configuration_id}

Delete a code security configuration for an enterprise

Parameters:

- {}:

- {}:

Responses:

- 204:
- 400:
- 403:
- 404:
- 409:

POST - /enterprises/{enterprise}/code-security/configurations/{configuration_id}/attach

Attach an enterprise configuration to repositories

Parameters:

- {}:
- {}:

Responses:

- 202:
- 403:
- 404:
- 409:

PUT - /enterprises/{enterprise}/code-security/configurations/{configuration_id}/defaults

Set a code security configuration as a default for an enterprise

Parameters:

- {}:
- {}:

Responses:

- 200: Default successfully changed.
- 403:
- 404:

GET - /enterprises/{enterprise}/code-security/configurations/{configuration_id}/repositories

Get repositories associated with an enterprise code security configuration

Parameters:

- {}:
- {}:
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- {}:
- {}:
- status (query): A comma-separated list of statuses. If specified, only repositories with these attachment statuses will be returned. Can be: `all`, `attached`, `attaching`, `removed`, `enforced`, `failed`, `updating`, `removed_by_enterprise`

Responses:

- 200: Response
- 403:
- 404:

GET - /enterprises/{enterprise}/dependabot/alerts

List Dependabot alerts for an enterprise

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:

- 404:
- 422:

GET - /enterprises/{enterprise}/secret-scanning/alerts

List secret scanning alerts for an enterprise

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 503:

GET - /events

List public events

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 503:

GET - /feeds

Get feeds

Responses:

- 200: Response

GET - /gists

List gists for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:

POST - /gists

Create a gist

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:

GET - /gists/public

List public gists

Parameters:

- {}:

- {}:
- {}:

Responses:

- 200: Response
- 422:
- 304:
- 403:

GET - /gists/starred

List starred gists

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 401:
- 304:
- 403:

GET - /gists/{gist_id}

Get a gist

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:
- 304:

PATCH - /gists/{gist_id}

Update a gist

Parameters:

- {}:

Responses:

- 200: Response
- 422:
- 404:

DELETE - /gists/{gist_id}

Delete a gist

Parameters:

- {}:

Responses:

- 204: Response
- 404:
- 304:
- 403:

GET - /gists/{gist_id}/comments

List gist comments

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:

POST - /gists/{gist_id}/comments

Create a gist comment

Parameters:

- {}:

Responses:

- 201: Response
- 304:
- 404:
- 403:

GET - /gists/{gist_id}/comments/{comment_id}

Get a gist comment

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:

PATCH - /gists/{gist_id}/comments/{comment_id}

Update a gist comment

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /gists/{gist_id}/comments/{comment_id}

Delete a gist comment

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:

GET - /gists/{gist_id}/commits

List gist commits

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 304:
- 403:

GET - /gists/{gist_id}/forks

List gist forks

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 304:
- 403:

POST - /gists/{gist_id}/forks

Fork a gist

Parameters:

- {}:

Responses:

- 201: Response
- 404:
- 422:
- 304:
- 403:

GET - /gists/{gist_id}/star

Check if a gist is starred

Parameters:

- {}:

Responses:

- 204: Response if gist is starred
- 404: Not Found if gist is not starred
- 304:
- 403:

PUT - /gists/{gist_id}/star

Star a gist

Parameters:

- {}:

Responses:

- 204: Response
- 404:
- 304:
- 403:

DELETE - /gists/{gist_id}/star

Unstar a gist

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:

GET - /gists/{gist_id}/{sha}

Get a gist revision

Parameters:

- {}:
- sha (path):

Responses:

- 200: Response
- 422:
- 404:
- 403:

GET - /gitignore/templates

Get all gitignore templates

Responses:

- 200: Response
- 304:

GET - /gitignore/templates/{name}

Get a gitignore template

Parameters:

- name (path):

Responses:

- 200: Response
- 304:

GET - /installation/repositories

List repositories accessible to the app installation

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 304:
- 401:

DELETE - /installation/token

Revoke an installation access token

Responses:

- 204: Response

GET - /issues

List issues assigned to the authenticated user

Parameters:

- filter (query): Indicates which sorts of issues to return. `assigned` means issues assigned to you. `created` means issues created by you. `mentioned` means issues mentioning you. `subscribed` means issues you're subscribed to updates for. `all` or `repos` means all issues you can see, regardless of participation or creation.
- state (query): Indicates the state of the issues to return.
- {}:
- sort (query): What to sort results by.
- {}:
- {}:
- collab (query):
- orgs (query):
- owned (query):
- pulls (query):
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 304:
- 404:

GET - /licenses

Get all commonly used licenses

Parameters:

- featured (query):
- {}:
- {}:

Responses:

- 200: Response
- 304:

GET - /licenses/{license}

Get a license

Parameters:

- license (path):

Responses:

- 200: Response
- 403:
- 404:
- 304:

POST - /markdown

Render a Markdown document

Responses:

- 200: Response
- 304:

POST - /markdown/raw

Render a Markdown document in raw mode

Responses:

- 200: Response
- 304:

GET - /marketplace_listing/accounts/{account_id}

Get a subscription plan for an account

Parameters:

- {}:

Responses:

- 200: Response
- 404: Not Found when the account has not purchased the listing
- 401:

GET - /marketplace_listing/plans

List plans

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 401:

GET - /marketplace_listing/plans/{plan_id}/accounts

List accounts for a plan

Parameters:

- {}:
- {}:
- direction (query): To return the oldest accounts first, set to `asc`. Ignored without the `sort` parameter.
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:
- 401:

GET - /marketplace_listing/stubbed/accounts/{account_id}

Get a subscription plan for an account (stubbed)

Parameters:

- {}:

Responses:

- 200: Response
- 404: Not Found when the account has not purchased the listing
- 401:

GET - /marketplace_listing/stubbed/plans

List plans (stubbed)

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 401:

GET - /marketplace_listing/stubbed/plans/{plan_id}/accounts

List accounts for a plan (stubbed)

Parameters:

- {}:
- {}:
- direction (query): To return the oldest accounts first, set to `asc`. Ignored without the `sort` parameter.
- {}:
- {}:

Responses:

- 200: Response
- 401:

GET - /meta

Get GitHub meta information

Responses:

- 200: Response
- 304:

GET - /networks/{owner}/{repo}/events

List public events for a network of repositories

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 403:
- 304:
- 301:

GET - /notifications

List notifications for the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- per_page (query): The number of results per page (max 50). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."

Responses:

- 200: Response
- 304:
- 403:
- 401:
- 422:

PUT - /notifications

Mark notifications as read

Responses:

- 202: Response
- 205: Reset Content
- 304:
- 403:
- 401:

GET - /notifications/threads/{thread_id}

Get a thread

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

PATCH - /notifications/threads/{thread_id}

Mark a thread as read

Parameters:

- {}:

Responses:

- 205: Reset Content
- 304:
- 403:

DELETE - /notifications/threads/{thread_id}

Mark a thread as done

Parameters:

- {}:

Responses:

- 204: No content

GET - /notifications/threads/{thread_id}/subscription

Get a thread subscription for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

PUT - /notifications/threads/{thread_id}/subscription

Set a thread subscription

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

DELETE - /notifications/threads/{thread_id}/subscription

Delete a thread subscription

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 403:
- 401:

GET - /octocat

Get Octocat

Parameters:

- s (query): The words to show in Octocat's speech bubble

Responses:

- 200: Response

GET - /organizations

List organizations

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:

GET - /organizations/{org}/settings/billing/usage

Get billing usage report for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200:
- 400:
- 403:
- 500:
- 503:

GET - /orgs/{org}

Get an organization

Parameters:

- {}:

Responses:

- 200: Response
- 404:

PATCH - /orgs/{org}

Update an organization

Parameters:

- {}:

Responses:

- 200: Response
- 422: Validation failed
- 409:

DELETE - /orgs/{org}

Delete an organization

Parameters:

- {}:

Responses:

- 202:
- 404:
- 403:

GET - /orgs/{org}/actions/cache/usage

Get GitHub Actions cache usage for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/cache/usage-by-repository

List repositories with GitHub Actions cache usage for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners

List GitHub-hosted runners for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/actions/hosted-runners

Create a GitHub-hosted runner for an organization

Parameters:

- {}:

Responses:

- 201: Response

GET - /orgs/{org}/actions/hosted-runners/images/github-owned

Get GitHub-owned images for GitHub-hosted runners in an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners/images/partner

Get partner images for GitHub-hosted runners in an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners/limits

Get limits on GitHub-hosted runners for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners/machine-sizes

Get GitHub-hosted runners machine specs for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners/platforms

Get platforms for GitHub-hosted runners in an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/hosted-runners/{hosted_runner_id}

Get a GitHub-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/actions/hosted-runners/{hosted_runner_id}

Update a GitHub-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/actions/hosted-runners/{hosted_runner_id}

Delete a GitHub-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 202: Response

GET - /orgs/{org}/actions/oidc/customization/sub

Get the customization template for an OIDC subject claim for an organization

Parameters:

- {}:

Responses:

- 200: A JSON serialized template for OIDC subject claim customization

PUT - /orgs/{org}/actions/oidc/customization/sub

Set the customization template for an OIDC subject claim for an organization

Parameters:

- {}:

Responses:

- 201: Empty response
- 404:
- 403:

GET - /orgs/{org}/actions/permissions

Get GitHub Actions permissions for an organization

Parameters:

- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/permissions

Set GitHub Actions permissions for an organization

Parameters:

- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/permissions/repositories

List selected repositories enabled for GitHub Actions in an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/permissions/repositories

Set selected repositories enabled for GitHub Actions in an organization

Parameters:

- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/actions/permissions/repositories/{repository_id}

Enable a selected repository for GitHub Actions in an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/actions/permissions/repositories/{repository_id}

Disable a selected repository for GitHub Actions in an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/permissions/selected-actions

Get allowed actions and reusable workflows for an organization

Parameters:

- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/permissions/selected-actions

Set allowed actions and reusable workflows for an organization

Parameters:

- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/permissions/workflow

Get default workflow permissions for an organization

Parameters:

- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/permissions/workflow

Set default workflow permissions for an organization

Parameters:

- {}:

Responses:

- 204: Success response

GET - /orgs/{org}/actions/runner-groups

List self-hosted runner groups for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/actions/runner-groups

Create a self-hosted runner group for an organization

Parameters:

- {}:

Responses:

- 201: Response

GET - /orgs/{org}/actions/runner-groups/{runner_group_id}

Get a self-hosted runner group for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/actions/runner-groups/{runner_group_id}

Update a self-hosted runner group for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/actions/runner-groups/{runner_group_id}

Delete a self-hosted runner group from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/runner-groups/{runner_group_id}/hosted-runners

List GitHub-hosted runners in a group for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/runner-groups/{runner_group_id}/repositories

List repository access to a self-hosted runner group in an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/runner-groups/{runner_group_id}/repositories

Set repository access for a self-hosted runner group in an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/actions/runner-groups/{runner_group_id}/repositories/{repository_id}

Add repository access to a self-hosted runner group in an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/actions/runner-groups/{runner_group_id}/repositories/{repository_id}

Remove repository access to a self-hosted runner group in an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/runner-groups/{runner_group_id}/runners

List self-hosted runners in a group for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/runner-groups/{runner_group_id}/runners

Set self-hosted runners in a group for an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/actions/runner-groups/{runner_group_id}/runners/{runner_id}

Add a self-hosted runner to a group for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/actions/runner-groups/{runner_group_id}/runners/{runner_id}

Remove a self-hosted runner from a group for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/runners

List self-hosted runners for an organization

Parameters:

- name (query): The name of a self-hosted runner.
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/runners/downloads

List runner applications for an organization

Parameters:

- {}:

Responses:

- 200: Response

POST - /orgs/{org}/actions/runners/generate-jitconfig

Create configuration for a just-in-time runner for an organization

Parameters:

- {}:

Responses:

- 201:
- 404:
- 422:
- 409:

POST - /orgs/{org}/actions/runners/registration-token

Create a registration token for an organization

Parameters:

- {}:

Responses:

- 201: Response

POST - /orgs/{org}/actions/runners/remove-token

Create a remove token for an organization

Parameters:

- {}:

Responses:

- 201: Response

GET - /orgs/{org}/actions/runners/{runner_id}

Get a self-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/actions/runners/{runner_id}

Delete a self-hosted runner from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/runners/{runner_id}/labels

List labels for a self-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200:
- 404:

POST - /orgs/{org}/actions/runners/{runner_id}/labels

Add custom labels to a self-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200:
- 404:
- 422:

PUT - /orgs/{org}/actions/runners/{runner_id}/labels

Set custom labels for a self-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200:
- 404:
- 422:

DELETE - /orgs/{org}/actions/runners/{runner_id}/labels

Remove all custom labels from a self-hosted runner for an organization

Parameters:

- {}:
- {}:

Responses:

- 200:
- 404:

DELETE - /orgs/{org}/actions/runners/{runner_id}/labels/{name}

Remove a custom label from a self-hosted runner for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200:
- 404:
- 422:

GET - /orgs/{org}/actions/secrets

List organization secrets

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/secrets/public-key

Get an organization public key

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/actions/secrets/{secret_name}

Get an organization secret

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/secrets/{secret_name}

Create or update an organization secret

Parameters:

- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /orgs/{org}/actions/secrets/{secret_name}

Delete an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/secrets/{secret_name}/repositories

List selected repositories for an organization secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/actions/secrets/{secret_name}/repositories

Set selected repositories for an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/actions/secrets/{secret_name}/repositories/{repository_id}

Add selected repository to an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: No Content when repository was added to the selected list
- 409: Conflict when visibility type is not set to selected

DELETE - /orgs/{org}/actions/secrets/{secret_name}/repositories/{repository_id}

Remove selected repository from an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: Response when repository was removed from the selected list
- 409: Conflict when visibility type not set to selected

GET - /orgs/{org}/actions/variables

List organization variables

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/actions/variables

Create an organization variable

Parameters:

- {}:

Responses:

- 201: Response when creating a variable

GET - /orgs/{org}/actions/variables/{name}

Get an organization variable

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/actions/variables/{name}

Update an organization variable

Parameters:

- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/actions/variables/{name}

Delete an organization variable

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/actions/variables/{name}/repositories

List selected repositories for an organization variable

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 409: Response when the visibility of the variable is not set to `selected`

PUT - /orgs/{org}/actions/variables/{name}/repositories

Set selected repositories for an organization variable

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 409: Response when the visibility of the variable is not set to `selected`

PUT - /orgs/{org}/actions/variables/{name}/repositories/{repository_id}

Add selected repository to an organization variable

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: Response
- 409: Response when the visibility of the variable is not set to `selected`

DELETE - /orgs/{org}/actions/variables/{name}/repositories/{repository_id}

Remove selected repository from an organization variable

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: Response
- 409: Response when the visibility of the variable is not set to `selected`

GET - /orgs/{org}/attestations/{subject_digest}

List attestations

Parameters:

- {}:
- {}:
- {}:
- {}:
- subject_digest (path): The parameter should be set to the attestation's subject's SHA256 digest, in the form `sha256:HEX_DIGEST`.
- predicate_type (query): Optional filter for fetching attestations with a given predicate type. This option accepts `provenance`, `sbom`, or freeform text for custom predicate types.

Responses:

- 200: Response

GET - /orgs/{org}/blocks

List users blocked by an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/blocks/{username}

Check if a user is blocked by an organization

Parameters:

- {}:
- {}:

Responses:

- 204: If the user is blocked
- 404: If the user is not blocked

PUT - /orgs/{org}/blocks/{username}

Block a user from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422:

DELETE - /orgs/{org}/blocks/{username}

Unblock a user from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/code-scanning/alerts

List code scanning alerts for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- state (query): If specified, only code scanning alerts with this state will be returned.
- sort (query): The property by which to sort the results.
- severity (query): If specified, only code scanning alerts with this severity will be returned.

Responses:

- 200: Response
- 404:
- 503:

GET - /orgs/{org}/code-security/configurations

Get code security configurations for an organization

Parameters:

- {}:
- target_type (query): The target type of the code security configuration
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

POST - /orgs/{org}/code-security/configurations

Create a code security configuration

Parameters:

- {}:

Responses:

- 201: Successfully created code security configuration

GET - /orgs/{org}/code-security/configurations/defaults

Get default code security configurations

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:

DELETE - /orgs/{org}/code-security/configurations/detach

Detach configurations from repositories

Parameters:

- {}:

Responses:

- 204:
- 400:
- 403:
- 404:
- 409:

GET - /orgs/{org}/code-security/configurations/{configuration_id}

Get a code security configuration

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:

PATCH - /orgs/{org}/code-security/configurations/{configuration_id}

Update a code security configuration

Parameters:

- {}:
- {}:

Responses:

- 200: Response when a configuration is updated
- 204: Response when no new updates are made

DELETE - /orgs/{org}/code-security/configurations/{configuration_id}

Delete a code security configuration

Parameters:

- {}:
- {}:

Responses:

- 204:
- 400:
- 403:
- 404:
- 409:

POST - /orgs/{org}/code-security/configurations/{configuration_id}/attach

Attach a configuration to repositories

Parameters:

- {}:
- {}:

Responses:

- 202:

PUT - /orgs/{org}/code-security/configurations/{configuration_id}/defaults

Set a code security configuration as a default for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Default successfully changed.
- 403:
- 404:

GET - /orgs/{org}/code-security/configurations/{configuration_id}/repositories

Get repositories associated with a code security configuration

Parameters:

- {}:
- {}:
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- {}:
- {}:
- status (query): A comma-separated list of statuses. If specified, only repositories with these attachment statuses will be returned. Can be: `all`, `attached`, `attaching`, `detached`, `removed`, `enforced`, `failed`, `updating`, `removed_by_enterprise`

Responses:

- 200: Response
- 403:
- 404:

GET - /orgs/{org}/codespaces

List codespaces for the organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

PUT - /orgs/{org}/codespaces/access

Manage access control for organization codespaces

Parameters:

- {}:

Responses:

- 204: Response when successfully modifying permissions.
- 304:
- 400: Users are neither members nor collaborators of this organization.
- 404:
- 422:
- 500:

POST - /orgs/{org}/codespaces/access/selected_users

Add users to Codespaces access for an organization

Parameters:

- {}:

Responses:

- 204: Response when successfully modifying permissions.
- 304:
- 400: Users are neither members nor collaborators of this organization.
- 404:
- 422:
- 500:

DELETE - /orgs/{org}/codespaces/access/selected_users

Remove users from Codespaces access for an organization

Parameters:

- {}:

Responses:

- 204: Response when successfully modifying permissions.
- 304:
- 400: Users are neither members nor collaborators of this organization.
- 404:
- 422:
- 500:

GET - /orgs/{org}/codespaces/secrets

List organization secrets

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/codespaces/secrets/public-key

Get an organization public key

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/codespaces/secrets/{secret_name}

Get an organization secret

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/codespaces/secrets/{secret_name}

Create or update an organization secret

Parameters:

- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret
- 404:
- 422:

DELETE - /orgs/{org}/codespaces/secrets/{secret_name}

Delete an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /orgs/{org}/codespaces/secrets/{secret_name}/repositories

List selected repositories for an organization secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /orgs/{org}/codespaces/secrets/{secret_name}/repositories

Set selected repositories for an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 409: Conflict when visibility type not set to selected

PUT - /orgs/{org}/codespaces/secrets/{secret_name}/repositories/{repository_id}

Add selected repository to an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: No Content when repository was added to the selected list
- 404:
- 409: Conflict when visibility type is not set to selected
- 422:

DELETE - /orgs/{org}/codespaces/secrets/{secret_name}/repositories/{repository_id}

Remove selected repository from an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: Response when repository was removed from the selected list
- 404:
- 409: Conflict when visibility type not set to selected
- 422:

GET - /orgs/{org}/copilot/billing

Get Copilot seat information and settings for an organization

Parameters:

- {}:

Responses:

- 200: OK
- 500:
- 401:
- 403:
- 404:
- 422: There is a problem with your account's associated payment method.

GET - /orgs/{org}/copilot/billing/seats

List all Copilot seat assignments for an organization

Parameters:

- {}:

- {}:
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."

Responses:

- 200: Response
- 500:
- 401:
- 403:
- 404:

POST - /orgs/{org}/copilot/billing/selected_teams

Add teams to the Copilot subscription for an organization

Parameters:

- {}:

Responses:

- 201: OK
- 500:
- 401:
- 403:
- 404:
- 422: Copilot Business or Enterprise is not enabled for this organization, billing has not been set up for this organization, a public code suggestions policy has not been set for this organization, or the organization's Copilot access setting is set to enable Copilot for all users or is unconfigured.

DELETE - /orgs/{org}/copilot/billing/selected_teams

Remove teams from the Copilot subscription for an organization

Parameters:

- {}:

Responses:

- 200: OK
- 500:
- 401:
- 403:
- 404:
- 422: Copilot Business or Enterprise is not enabled for this organization, billing has not been set up for this organization, a public code suggestions policy has not been set for this organization, or the organization's Copilot access setting is set to enable Copilot for all users or is unconfigured.

POST - /orgs/{org}/copilot/billing/selected_users

Add users to the Copilot subscription for an organization

Parameters:

- {}:

Responses:

- 201: OK
- 500:
- 401:
- 403:
- 404:
- 422: Copilot Business or Enterprise is not enabled for this organization, billing has not been set up for this organization, a public code suggestions policy has not been set for this organization, or the organization's Copilot access setting is set to enable Copilot for all users or is unconfigured.

DELETE - /orgs/{org}/copilot/billing/selected_users

Remove users from the Copilot subscription for an organization

Parameters:

- {}:

Responses:

- 200: OK
- 500:
- 401:
- 403:
- 404:
- 422: Copilot Business or Enterprise is not enabled for this organization, billing has not been set up for this organization, a public code suggestions policy has not been set for this organization, the seat management setting is set to enable Copilot for all users or is unconfigured, or a user's seat cannot be cancelled because it was assigned to them via a team.

GET - /orgs/{org}/copilot/metrics

Get Copilot metrics for an organization

Parameters:

- {}:
- since (query): Show usage metrics since this date. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format ('YYYY-MM-DDTHH:MM:SSZ'). Maximum value is 28 days ago.
- until (query): Show usage metrics until this date. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format ('YYYY-MM-DDTHH:MM:SSZ') and should not precede the 'since' date if it is passed.
- {}:
- per_page (query): The number of days of metrics to display per page (max 28). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."

Responses:

- 200: Response
- 500:
- 403:
- 404:
- 422:

GET - /orgs/{org}/dependabot/alerts

List Dependabot alerts for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 400:
- 403:
- 404:
- 422:

GET - /orgs/{org}/dependabot/secrets

List organization secrets

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/dependabot/secrets/public-key

Get an organization public key

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/dependabot/secrets/{secret_name}

Get an organization secret

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/dependabot/secrets/{secret_name}

Create or update an organization secret

Parameters:

- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /orgs/{org}/dependabot/secrets/{secret_name}

Delete an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/dependabot/secrets/{secret_name}/repositories

List selected repositories for an organization secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/dependabot/secrets/{secret_name}/repositories

Set selected repositories for an organization secret

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/dependabot/secrets/{secret_name}/repositories/{repository_id}

Add selected repository to an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: No Content when repository was added to the selected list
- 409: Conflict when visibility type is not set to selected

DELETE - /orgs/{org}/dependabot/secrets/{secret_name}/repositories/{repository_id}

Remove selected repository from an organization secret

Parameters:

- {}:
- {}:
- repository_id (path):

Responses:

- 204: Response when repository was removed from the selected list
- 409: Conflict when visibility type not set to selected

GET - /orgs/{org}/docker/conflicts

Get list of conflicting packages during Docker migration for organization

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 401:

GET - /orgs/{org}/events

List public organization events

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/failed_invitations

List failed organization invitations

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /orgs/{org}/hooks

List organization webhooks

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /orgs/{org}/hooks

Create an organization webhook

Parameters:

- {}:

Responses:

- 201: Response
- 422:
- 404:

GET - /orgs/{org}/hooks/{hook_id}

Get an organization webhook

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /orgs/{org}/hooks/{hook_id}

Update an organization webhook

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:

DELETE - /orgs/{org}/hooks/{hook_id}

Delete an organization webhook

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /orgs/{org}/hooks/{hook_id}/config

Get a webhook configuration for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/hooks/{hook_id}/config

Update a webhook configuration for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/hooks/{hook_id}/deliveries

List deliveries for an organization webhook

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:

GET - /orgs/{org}/hooks/{hook_id}/deliveries/{delivery_id}

Get a webhook delivery for an organization webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:

POST - /orgs/{org}/hooks/{hook_id}/deliveries/{delivery_id}/attempts

Redeliver a delivery for an organization webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202:
- 400:
- 422:

POST - /orgs/{org}/hooks/{hook_id}/pings

Ping an organization webhook

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /orgs/{org}/insights/api/route-stats/{actor_type}/{actor_id}

Get route stats by actor

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/insights/api/subject-stats

Get subject stats

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/insights/api/summary-stats

Get summary stats

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/insights/api/summary-stats/users/{user_id}

Get summary stats by user

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/installation

Get an organization installation for the authenticated app

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/installations

List app installations for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/interaction-limits

Get interaction restrictions for an organization

Parameters:

- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/interaction-limits

Set interaction restrictions for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 422:

DELETE - /orgs/{org}/interaction-limits

Remove interaction restrictions for an organization

Parameters:

- {}:

Responses:

- 204: Response

GET - /orgs/{org}/invitations

List pending organization invitations

Parameters:

- {}:
- {}:
- {}:
- role (query): Filter invitations by their member role.
- invitation_source (query): Filter invitations by their invitation source.

Responses:

- 200: Response
- 404:

POST - /orgs/{org}/invitations

Create an organization invitation

Parameters:

- {}:

Responses:

- 201: Response
- 422:
- 404:

DELETE - /orgs/{org}/invitations/{invitation_id}

Cancel an organization invitation

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422:
- 404:

GET - /orgs/{org}/invitations/{invitation_id}/teams

List organization invitation teams

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /orgs/{org}/issue-types

List issue types for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 404:

POST - /orgs/{org}/issue-types

Create issue type for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 404:
- 422:

PUT - /orgs/{org}/issue-types/{issue_type_id}

Update issue type for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

DELETE - /orgs/{org}/issue-types/{issue_type_id}

Delete issue type for an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422:
- 404:

GET - /orgs/{org}/issues

List organization issues assigned to the authenticated user

Parameters:

- {}:
- filter (query): Indicates which sorts of issues to return. `assigned` means issues assigned to you. `created` means issues created by you. `mentioned` means issues mentioning you. `subscribed` means issues you're subscribed to updates for. `all` or `repos` means all issues you can see, regardless of participation or creation.
- state (query): Indicates the state of the issues to return.
- {}:
- type (query): Can be the name of an issue type.
- sort (query): What to sort results by.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /orgs/{org}/members

List organization members

Parameters:

- {}:
- filter (query): Filter members returned in the list. `2fa_disabled` means that only members without [two-factor authentication](https://github.com/blog/1614-two-factor-authentication) enabled will be returned. This options is only available for organization owners.
- role (query): Filter members returned by their role.
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /orgs/{org}/members/{username}

Check organization membership for a user

Parameters:

- {}:
- {}:

Responses:

- 204: Response if requester is an organization member and user is a member
- 302: Response if requester is not an organization member
- 404: Not Found if requester is an organization member and user is not a member

DELETE - /orgs/{org}/members/{username}

Remove an organization member

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403:

GET - /orgs/{org}/members/{username}/codespaces

List codespaces for a user in organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

DELETE - /orgs/{org}/members/{username}/codespaces/{codespace_name}

Delete a codespace from the organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202:
- 304:
- 500:
- 401:
- 403:
- 404:

POST - /orgs/{org}/members/{username}/codespaces/{codespace_name}/stop

Stop a codespace for an organization user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

GET - /orgs/{org}/members/{username}/copilot

Get Copilot seat assignment details for a user

Parameters:

- {}:
- {}:

Responses:

- 200: The user's GitHub Copilot seat details, including usage.
- 500:
- 401:
- 403:
- 404:
- 422: Copilot Business or Enterprise is not enabled for this organization or the user has a pending organization invitation.

GET - /orgs/{org}/memberships/{username}

Get organization membership for a user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 403:

PUT - /orgs/{org}/memberships/{username}

Set organization membership for a user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 422:
- 403:

DELETE - /orgs/{org}/memberships/{username}

Remove organization membership for a user

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403:
- 404:

GET - /orgs/{org}/migrations

List organization migrations

Parameters:

- {}:
- {}:
- {}:
- exclude (query): Exclude attributes from the API response to improve performance

Responses:

- 200: Response

POST - /orgs/{org}/migrations

Start an organization migration

Parameters:

- {}:

Responses:

- 201: Response
- 404:
- 422:

GET - /orgs/{org}/migrations/{migration_id}

Get an organization migration status

Parameters:

- {}:
- {}:
- exclude (query): Exclude attributes from the API response to improve performance

Responses:

- 200: * `pending`, which means the migration hasn't started yet. * `exporting`, which means the migration is in progress. * `exported`, which means the migration finished successfully. * `failed`, which means the migration failed.
- 404:

GET - /orgs/{org}/migrations/{migration_id}/archive

Download an organization migration archive

Parameters:

- {}:
- {}:

Responses:

- 302: Response
- 404:

DELETE - /orgs/{org}/migrations/{migration_id}/archive

Delete an organization migration archive

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:

DELETE - /orgs/{org}/migrations/{migration_id}/repos/{repo_name}/lock

Unlock an organization repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /orgs/{org}/migrations/{migration_id}/repositories

List repositories in an organization migration

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /orgs/{org}/organization-roles

Get all organization roles for an organization

Parameters:

- {}:

Responses:

- 200: Response - list of organization roles
- 404:
- 422:

DELETE - /orgs/{org}/organization-roles/teams/{team_slug}

Remove all organization roles for a team

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/organization-roles/teams/{team_slug}/{role_id}

Assign an organization role to a team

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404: Response if the organization, team or role does not exist.
- 422: Response if the organization roles feature is not enabled for the organization, or validation failed.

DELETE - /orgs/{org}/organization-roles/teams/{team_slug}/{role_id}

Remove an organization role from a team

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/organization-roles/users/{username}

Remove all organization roles for a user

Parameters:

- {}:
- {}:

Responses:

- 204: Response

PUT - /orgs/{org}/organization-roles/users/{username}/{role_id}

Assign an organization role to a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404: Response if the organization, user or role does not exist.
- 422: Response if the organization roles feature is not enabled enabled for the organization, the validation failed, or the user is not an organization member.

DELETE - /orgs/{org}/organization-roles/users/{username}/{role_id}

Remove an organization role from a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/organization-roles/{role_id}

Get an organization role

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

GET - /orgs/{org}/organization-roles/{role_id}/teams

List teams that are assigned to an organization role

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response - List of assigned teams
- 404: Response if the organization or role does not exist.

- 422: Response if the organization roles feature is not enabled or validation failed.

GET - /orgs/{org}/organization-roles/{role_id}/users

List users that are assigned to an organization role

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response - List of assigned users
- 404: Response if the organization or role does not exist.
- 422: Response if the organization roles feature is not enabled or validation failed.

GET - /orgs/{org}/outside_collaborators

List outside collaborators for an organization

Parameters:

- {}:
- filter (query): Filter the list of outside collaborators. '2fa_disabled' means that only outside collaborators without [two-factor authentication](https://github.com/blog/1614-two-factor-authentication) enabled will be returned.
- {}:
- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/outside_collaborators/{username}

Convert an organization member to outside collaborator

Parameters:

- {}:
- {}:

Responses:

- 202: User is getting converted asynchronously
- 204: User was converted
- 403: Forbidden if user is the last owner of the organization, not a member of the organization, or if the enterprise enforces a policy for inviting outside collaborators. For more information, see "[Enforcing repository management policies in your enterprise](https://docs.github.com/admin/policies/enforcing-policies-for-your-enterprise/enforcing-repository-management-policies-in-your-enterprise#enforcing-a-policy-for-inviting-outside-collaborators-to-repositories)."
- 404:

DELETE - /orgs/{org}/outside_collaborators/{username}

Remove outside collaborator from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422: Unprocessable Entity if user is a member of the organization

GET - /orgs/{org}/packages

List packages for an organization

Parameters:

- package_type (query): The type of supported package. Packages in GitHub's Gradle registry have the type 'maven'. Docker images pushed to GitHub's Container registry ('ghcr.io') have the type 'container'. You can use the type 'docker' to find images that were pushed to GitHub's Docker registry ('docker.pkg.github.com'), even if these have now been migrated to the Container registry.
- {}:
- {}:
- page (query): The page number of the results to fetch. For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."

Responses:

- 200: Response
- 403:

- 401:
- 400:

GET - /orgs/{org}/packages/{package_type}/{package_name}

Get a package for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/packages/{package_type}/{package_name}

Delete a package for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /orgs/{org}/packages/{package_type}/{package_name}/restore

Restore a package for an organization

Parameters:

- {}:
- {}:
- {}:
- token (query): package token

Responses:

- 204: Response
- 404:
- 403:
- 401:

GET - /orgs/{org}/packages/{package_type}/{package_name}/versions

List package versions for a package owned by an organization

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- state (query): The state of the package, either active or deleted.

Responses:

- 200: Response
- 404:
- 403:
- 401:

GET - /orgs/{org}/packages/{package_type}/{package_name}/versions/{package_version_id}

Get a package version for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/packages/{package_type}/{package_name}/versions/{package_version_id}

Delete package version for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /orgs/{org}/packages/{package_type}/{package_name}/versions/{package_version_id}/restore

Restore package version for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

GET - /orgs/{org}/personal-access-token-requests

List requests to access organization resources with fine-grained personal access tokens

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 500:
- 422:
- 404:
- 403:
- 200: Response

POST - /orgs/{org}/personal-access-token-requests

Review requests to access organization resources with fine-grained personal access tokens

Parameters:

- {}:

Responses:

- 500:
- 422:
- 404:
- 403:
- 202:

POST - /orgs/{org}/personal-access-token-requests/{pat_request_id}

Review a request to access organization resources with a fine-grained personal access token

Parameters:

- {}:
- pat_request_id (path): Unique identifier of the request for access via fine-grained personal access token.

Responses:

- 500:
- 422:
- 404:
- 403:
- 204:

GET - /orgs/{org}/personal-access-token-requests/{pat_request_id}/repositories

List repositories requested to be accessed by a fine-grained personal access token

Parameters:

- {}:
- pat_request_id (path): Unique identifier of the request for access via fine-grained personal access token.
- {}:
- {}:

Responses:

- 500:
- 404:
- 403:
- 200: Response

GET - /orgs/{org}/personal-access-tokens

List fine-grained personal access tokens with access to organization resources

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 500:
- 422:
- 404:
- 403:
- 200: Response

POST - /orgs/{org}/personal-access-tokens

Update the access to organization resources via fine-grained personal access tokens

Parameters:

- {}:

Responses:

- 500:
- 404:
- 202:
- 403:
- 422:

POST - /orgs/{org}/personal-access-tokens/{pat_id}

Update the access a fine-grained personal access token has to organization resources

Parameters:

- {}:
- {}:

Responses:

- 500:
- 404:
- 204:
- 403:
- 422:

GET - /orgs/{org}/personal-access-tokens/{pat_id}/repositories

List repositories a fine-grained personal access token has access to

Parameters:

- {}:
- pat_id (path): Unique identifier of the fine-grained personal access token.
- {}:
- {}:

Responses:

- 500:
- 404:
- 403:
- 200: Response

GET - /orgs/{org}/private-registries

List private registries for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 404:

POST - /orgs/{org}/private-registries

Create a private registry for an organization

Parameters:

- {}:

Responses:

- 201: The organization private registry configuration
- 404:
- 422:

GET - /orgs/{org}/private-registries/public-key

Get private registries public key for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /orgs/{org}/private-registries/{secret_name}

Get a private registry for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: The specified private registry configuration for the organization
- 404:

PATCH - /orgs/{org}/private-registries/{secret_name}

Update a private registry for an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 422:

DELETE - /orgs/{org}/private-registries/{secret_name}

Delete a private registry for an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 400:
- 404:

GET - /orgs/{org}/projects

List organization projects

Parameters:

- {}:
- state (query): Indicates the state of the projects to return.
- {}:
- {}:

Responses:

- 200: Response
- 422:

POST - /orgs/{org}/projects

Create an organization project

Parameters:

- {}:

Responses:

- 201: Response
- 401:
- 403:
- 404:
- 410:
- 422:

GET - /orgs/{org}/properties/schema

Get all custom properties for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:

PATCH - /orgs/{org}/properties/schema

Create or update custom properties for an organization

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:

GET - /orgs/{org}/properties/schema/{custom_property_name}

Get a custom property for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

PUT - /orgs/{org}/properties/schema/{custom_property_name}

Create or update a custom property for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

DELETE - /orgs/{org}/properties/schema/{custom_property_name}

Remove a custom property for an organization

Parameters:

- {}:
- {}:

Responses:

- 204:
- 403:
- 404:

GET - /orgs/{org}/properties/values

List custom property values for organization repositories

Parameters:

- {}:
- {}:
- {}:
- repository_query (query): Finds repositories in the organization with a query containing one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query] (<https://docs.github.com/rest/search/search#constructing-a-search-query>). See "[Searching for repositories] (<https://docs.github.com/articles/searching-for-repositories/>)" for a detailed list of qualifiers.

Responses:

- 200: Response
- 403:
- 404:

PATCH - /orgs/{org}/properties/values

Create or update custom property values for organization repositories

Parameters:

- {}:

Responses:

- 204: No Content when custom property values are successfully created or updated
- 403:
- 404:
- 422:

GET - /orgs/{org}/public_members

List public organization members

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/public_members/{username}

Check public organization membership for a user

Parameters:

- {}:
- {}:

Responses:

- 204: Response if user is a public member
- 404: Not Found if user is not a public member

PUT - /orgs/{org}/public_members/{username}

Set public organization membership for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403:

DELETE - /orgs/{org}/public_members/{username}

Remove public organization membership for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/repos

List organization repositories

Parameters:

- {}:
- type (query): Specifies the types of repositories you want returned.
- sort (query): The property to sort the results by.
- direction (query): The order to sort by. Default: 'asc' when using 'full_name', otherwise 'desc'.
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/repos

Create an organization repository

Parameters:

- {}:

Responses:

- 201: Response
- 403:
- 422:

GET - /orgs/{org}/rulesets

Get all organization repository rulesets

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 500:

POST - /orgs/{org}/rulesets

Create an organization repository ruleset

Parameters:

- {}:

Responses:

- 201: Response
- 404:
- 500:

GET - /orgs/{org}/rulesets/rule-suites

List organization rule suites

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 500:

GET - /orgs/{org}/rulesets/rule-suites/{rule_suite_id}

Get an organization rule suite

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 500:

GET - /orgs/{org}/rulesets/{ruleset_id}

Get an organization repository ruleset

Parameters:

- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 200: Response
- 404:
- 500:

PUT - /orgs/{org}/rulesets/{ruleset_id}

Update an organization repository ruleset

Parameters:

- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 200: Response
- 404:
- 500:

DELETE - /orgs/{org}/rulesets/{ruleset_id}

Delete an organization repository ruleset

Parameters:

- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 204: Response
- 404:
- 500:

GET - /orgs/{org}/rulesets/{ruleset_id}/history

Get organization ruleset history

Parameters:

- {}:
- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 200: Response
- 404:
- 500:

GET - /orgs/{org}/rulesets/{ruleset_id}/history/{version_id}

Get organization ruleset version

Parameters:

- {}:
- ruleset_id (path): The ID of the ruleset.
- version_id (path): The ID of the version

Responses:

- 200: Response
- 404:
- 500:

GET - /orgs/{org}/secret-scanning/alerts

List secret scanning alerts for an organization

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 503:

GET - /orgs/{org}/security-advisories

List repository security advisories for an organization

Parameters:

- {}:
- {}:
- sort (query): The property to sort the results by.
- {}:
- {}:
- per_page (query): The number of advisories to return per page. For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- state (query): Filter by the state of the repository advisories. Only advisories of this state will be returned.

Responses:

- 200: Response
- 400:
- 404:

GET - /orgs/{org}/security-managers

List security manager teams

Parameters:

- {}:

Responses:

- 200: Response

PUT - /orgs/{org}/security-managers/teams/{team_slug}

Add a security manager team

Parameters:

- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/security-managers/teams/{team_slug}

Remove a security manager team

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/settings/billing/actions

Get GitHub Actions billing for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/settings/billing/packages

Get GitHub Packages billing for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/settings/billing/shared-storage

Get shared storage billing for an organization

Parameters:

- {}:

Responses:

- 200: Response

GET - /orgs/{org}/settings/network-configurations

List hosted compute network configurations for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/settings/network-configurations

Create a hosted compute network configuration for an organization

Parameters:

- {}:

Responses:

- 201: Response

GET - /orgs/{org}/settings/network-configurations/{network_configuration_id}

Get a hosted compute network configuration for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/settings/network-configurations/{network_configuration_id}

Update a hosted compute network configuration for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/settings/network-configurations/{network_configuration_id}

Delete a hosted compute network configuration from an organization

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/settings/network-settings/{network_settings_id}

Get a hosted compute network settings resource for an organization

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/team/{team_slug}/copilot/metrics

Get Copilot metrics for a team

Parameters:

- {}:
- {}:
- since (query): Show usage metrics since this date. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format ('YYYY-MM-DDTHH:MM:SSZ'). Maximum value is 28 days ago.
- until (query): Show usage metrics until this date. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format ('YYYY-MM-DDTHH:MM:SSZ') and should not precede the 'since' date if it is passed.
- {}:
- per_page (query): The number of days of metrics to display per page (max 28). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."

Responses:

- 200: Response
- 500:
- 403:
- 404:
- 422:

GET - /orgs/{org}/teams

List teams

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:

POST - /orgs/{org}/teams

Create a team

Parameters:

- {}:

Responses:

- 201: Response
- 422:
- 403:

GET - /orgs/{org}/teams/{team_slug}

Get a team by name

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /orgs/{org}/teams/{team_slug}

Update a team

Parameters:

- {}:
- {}:

Responses:

- 200: Response when the updated information already exists
- 201: Response
- 404:
- 422:
- 403:

DELETE - /orgs/{org}/teams/{team_slug}

Delete a team

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/discussions

List discussions

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- pinned (query): Pinned discussions only filter

Responses:

- 200: Response

POST - /orgs/{org}/teams/{team_slug}/discussions

Create a discussion

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}

Get a discussion

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}

Update a discussion

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}

Delete a discussion

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments

List discussion comments

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments

Create a discussion comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

GET - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}

Get a discussion comment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}

Update a discussion comment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}

Delete a discussion comment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}/reactions

List reactions for a team discussion comment

Parameters:

- {}:
- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a team discussion comment.
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}/reactions

Create reaction for a team discussion comment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response when the reaction type has already been added to this team discussion comment
- 201: Response

DELETE - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/comments/{comment_number}/reactions/{reaction_id}

Delete team discussion comment reaction

Parameters:

- {}:
- {}:
- {}:

- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/reactions

List reactions for a team discussion

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a team discussion.
- {}:
- {}:

Responses:

- 200: Response

POST - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/reactions

Create reaction for a team discussion

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 201: Response

DELETE - /orgs/{org}/teams/{team_slug}/discussions/{discussion_number}/reactions/{reaction_id}

Delete team discussion reaction

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/invitations

List pending team invitations

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/teams/{team_slug}/members

List team members

Parameters:

- {}:
- {}:
- role (query): Filters members returned by their role in the team.
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/teams/{team_slug}/memberships/{username}

Get team membership for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: if user has no team membership

PUT - /orgs/{org}/teams/{team_slug}/memberships/{username}

Add or update team membership for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403: Forbidden if team synchronization is set up
- 422: Unprocessable Entity if you attempt to add an organization to a team

DELETE - /orgs/{org}/teams/{team_slug}/memberships/{username}

Remove team membership for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403: Forbidden if team synchronization is set up

GET - /orgs/{org}/teams/{team_slug}/projects

List team projects

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/teams/{team_slug}/projects/{project_id}

Check team permissions for a project

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: Not Found if project is not managed by this team

PUT - /orgs/{org}/teams/{team_slug}/projects/{project_id}

Add or update team project permissions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403: Forbidden if the project is not owned by the organization

DELETE - /orgs/{org}/teams/{team_slug}/projects/{project_id}

Remove a project from a team

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/repos

List team repositories

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /orgs/{org}/teams/{team_slug}/repos/{owner}/{repo}

Check team permissions for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Alternative response with repository permissions
- 204: Response if team has permission for the repository. This is the response when the repository media type hasn't been provided in the Accept header.
- 404: Not Found if team does not have permission for the repository

PUT - /orgs/{org}/teams/{team_slug}/repos/{owner}/{repo}

Add or update team repository permissions

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /orgs/{org}/teams/{team_slug}/repos/{owner}/{repo}

Remove a repository from a team

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /orgs/{org}/teams/{team_slug}/teams

List child teams

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: if child teams exist

POST - /orgs/{org}/{security_product}/{enablement}

Enable or disable a security feature for an organization

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Action started
- 422: The action could not be taken due to an in progress enablement, or a policy is preventing enablement

GET - /projects/columns/cards/{card_id}

Get a project card

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:
- 404:

PATCH - /projects/columns/cards/{card_id}

Update an existing project card

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:
- 404:
- 422:

DELETE - /projects/columns/cards/{card_id}

Delete a project card

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 403: Forbidden
- 401:
- 404:

POST - /projects/columns/cards/{card_id}/moves

Move a project card

Parameters:

- {}:

Responses:

- 201: Response
- 304:
- 403: Forbidden
- 401:
- 503: Response
- 422:

GET - /projects/columns/{column_id}

Get a project column

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:
- 401:

PATCH - /projects/columns/{column_id}

Update an existing project column

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

DELETE - /projects/columns/{column_id}

Delete a project column

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 403:
- 401:

GET - /projects/columns/{column_id}/cards

List project cards

Parameters:

- {}:
- archived_state (query): Filters the project cards that are returned by the card's state.
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

POST - /projects/columns/{column_id}/cards

Create a project card

Parameters:

- {}:

Responses:

- 201: Response
- 304:
- 403:
- 401:
- 422: Validation failed
- 503: Response

POST - /projects/columns/{column_id}/moves

Move a project column

Parameters:

- {}:

Responses:

- 201: Response
- 304:
- 403:
- 422:
- 401:

GET - /projects/{project_id}

Get a project

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

PATCH - /projects/{project_id}

Update a project

Parameters:

- {}:

Responses:

- 200: Response
- 404: Not Found if the authenticated user does not have access to the project
- 304:
- 403: Forbidden
- 401:
- 410:
- 422:

DELETE - /projects/{project_id}

Delete a project

Parameters:

- {}:

Responses:

- 204: Delete Success
- 304:
- 403: Forbidden
- 401:
- 410:
- 404:

GET - /projects/{project_id}/collaborators

List project collaborators

Parameters:

- {}:
- affiliation (query): Filters the collaborators by their affiliation. `outside` means outside collaborators of a project that are not a member of the project's organization. `direct` means collaborators with permissions to a project, regardless of organization membership status. `all` means all collaborators the authenticated user can see.
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:
- 304:
- 403:
- 401:

PUT - /projects/{project_id}/collaborators/{username}

Add project collaborator

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 422:
- 304:
- 403:
- 401:

DELETE - /projects/{project_id}/collaborators/{username}

Remove user as a collaborator

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 422:
- 401:

GET - /projects/{project_id}/collaborators/{username}/permission

Get project permission for a user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:
- 304:
- 403:
- 401:

GET - /projects/{project_id}/columns

List project columns

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

POST - /projects/{project_id}/columns

Create a project column

Parameters:

- {}:

Responses:

- 201: Response
- 304:
- 403:
- 422:
- 401:

GET - /rate_limit

Get rate limit status for the authenticated user

Responses:

- 200: Response
- 304:
- 404:

GET - /repos/{owner}/{repo}

Get a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 301:

PATCH - /repos/{owner}/{repo}

Update a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 307:
- 403:
- 422:
- 404:

DELETE - /repos/{owner}/{repo}

Delete a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403: If an organization owner has configured the organization to prevent members from deleting organization-owned repositories, a member will get this response:
- 307:
- 404:

GET - /repos/{owner}/{repo}/actions/artifacts

List artifacts for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/artifacts/{artifact_id}

Get an artifact

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/actions/artifacts/{artifact_id}

Delete an artifact

Parameters:

- {}:
- {}:

- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/artifacts/{artifact_id}/{archive_format}

Download an artifact

Parameters:

- {}:
- {}:
- {}:
- archive_format (path):

Responses:

- 302: Response
- 410:

GET - /repos/{owner}/{repo}/actions/cache/usage

Get GitHub Actions cache usage for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/caches

List GitHub Actions caches for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/actions/caches

Delete GitHub Actions caches for a repository (using a cache key)

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/actions/caches/{cache_id}

Delete a GitHub Actions cache for a repository (using a cache ID)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/jobs/{job_id}

Get a job for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/jobs/{job_id}/logs

Download job logs for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 302: Response

POST - /repos/{owner}/{repo}/actions/jobs/{job_id}/rerun

Re-run a job from a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403:

GET - /repos/{owner}/{repo}/actions/oidc/customization/sub

Get the customization template for an OIDC subject claim for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Status response
- 400:
- 404:

PUT - /repos/{owner}/{repo}/actions/oidc/customization/sub

Set the customization template for an OIDC subject claim for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Empty response
- 404:
- 400:
- 422:

GET - /repos/{owner}/{repo}/actions/organization-secrets

List repository organization secrets

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/organization-variables

List repository organization variables

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/permissions

Get GitHub Actions permissions for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/permissions

Set GitHub Actions permissions for a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/permissions/access

Get the level of access for workflows outside of the repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/permissions/access

Set the level of access for workflows outside of the repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/permissions/selected-actions

Get allowed actions and reusable workflows for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/permissions/selected-actions

Set allowed actions and reusable workflows for a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/permissions/workflow

Get default workflow permissions for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/permissions/workflow

Set default workflow permissions for a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Success response
- 409: Conflict response when changing a setting is prevented by the owning organization

GET - /repos/{owner}/{repo}/actions/runners

List self-hosted runners for a repository

Parameters:

- name (query): The name of a self-hosted runner.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/runners/downloads

List runner applications for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/actions/runners/generate-jitconfig

Create configuration for a just-in-time runner for a repository

Parameters:

- {}:
- {}:

Responses:

- 201:
- 404:
- 422:
- 409:

POST - /repos/{owner}/{repo}/actions/runners/registration-token

Create a registration token for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Response

POST - /repos/{owner}/{repo}/actions/runners/remove-token

Create a remove token for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/actions/runners/{runner_id}

Get a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/actions/runners/{runner_id}

Delete a self-hosted runner from a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/runners/{runner_id}/labels

List labels for a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200:
- 404:

POST - /repos/{owner}/{repo}/actions/runners/{runner_id}/labels

Add custom labels to a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200:
- 404:
- 422:

PUT - /repos/{owner}/{repo}/actions/runners/{runner_id}/labels

Set custom labels for a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200:

- 404:
- 422:

DELETE - /repos/{owner}/{repo}/actions/runners/{runner_id}/labels

Remove all custom labels from a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200:
- 404:

DELETE - /repos/{owner}/{repo}/actions/runners/{runner_id}/labels/{name}

Remove a custom label from a self-hosted runner for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200:
- 404:
- 422:

GET - /repos/{owner}/{repo}/actions/runs

List workflow runs for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}

Get a workflow run

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/actions/runs/{run_id}

Delete a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/approvals

Get the review history for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/approve

Approve a workflow run for a fork pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 404:
- 403:

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/artifacts

List workflow run artifacts

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/attempts/{attempt_number}

Get a workflow run attempt

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/attempts/{attempt_number}/jobs

List jobs for a workflow run attempt

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/attempts/{attempt_number}/logs

Download workflow run attempt logs

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 302: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/cancel

Cancel a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202: Response
- 409:

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/deployment_protection_rule

Review custom deployment protection rules for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/force-cancel

Force cancel a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202: Response
- 409:

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/jobs

List jobs for a workflow run

Parameters:

- {}:
- {}:
- {}:
- filter (query): Filters jobs by their `completed_at` timestamp. `latest` returns jobs from the most recent execution of the workflow run. `all` returns all jobs for a workflow run, including from old executions of the workflow run.
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/logs

Download workflow run logs

Parameters:

- {}:
- {}:
- {}:

Responses:

- 302: Response

DELETE - /repos/{owner}/{repo}/actions/runs/{run_id}/logs

Delete workflow run logs

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403:
- 500:

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/pending_deployments

Get pending deployments for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/pending_deployments

Review pending deployments for a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/rerun

Re-run a workflow

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

POST - /repos/{owner}/{repo}/actions/runs/{run_id}/rerun-failed-jobs

Re-run failed jobs from a workflow run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/actions/runs/{run_id}/timing

Get workflow run usage

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/secrets

List repository secrets

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/secrets/public-key

Get a repository public key

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/secrets/{secret_name}

Get a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/secrets/{secret_name}

Create or update a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /repos/{owner}/{repo}/actions/secrets/{secret_name}

Delete a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/variables

List repository variables

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/actions/variables

Create a repository variable

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/actions/variables/{name}

Get a repository variable

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/actions/variables/{name}

Update a repository variable

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /repos/{owner}/{repo}/actions/variables/{name}

Delete a repository variable

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/workflows

List repository workflows

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/workflows/{workflow_id}

Get a workflow

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/actions/workflows/{workflow_id}/disable

Disable a workflow

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

POST - /repos/{owner}/{repo}/actions/workflows/{workflow_id}/dispatches

Create a workflow dispatch event

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

PUT - /repos/{owner}/{repo}/actions/workflows/{workflow_id}/enable

Enable a workflow

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/actions/workflows/{workflow_id}/runs

List workflow runs for a workflow

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/actions/workflows/{workflow_id}/timing

Get workflow usage

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/activity

List repository activities

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- ref(query): The Git reference for the activities you want to list. The `ref` for a branch can be formatted either as `refs/heads/BRANCH_NAME` or `BRANCH_NAME`, where `BRANCH_NAME` is the name of your branch.
- actor (query): The GitHub username to use to filter by the actor who performed the activity.
- time_period (query): The time period to filter by. For example, `day` will filter for activity that occurred in the past 24 hours, and `week` will filter for activity that occurred in the past 7 days (168 hours).

- activity_type (query): The activity type to filter by. For example, you can choose to filter by "force_push", to see all force pushes to the repository.

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/assignees

List assignees

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/assignees/{assignee}

Check if a user can be assigned

Parameters:

- {}:
- {}:
- assignee (path):

Responses:

- 204: If the `assignee` can be assigned to issues in the repository, a `204` header with no content is returned.
- 404: Otherwise a `404` status code is returned.

POST - /repos/{owner}/{repo}/attestations

Create an attestation

Parameters:

- {}:
- {}:

Responses:

- 201: response
- 403:
- 422:

GET - /repos/{owner}/{repo}/attestations/{subject_digest}

List attestations

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- subject_digest (path): The parameter should be set to the attestation's subject's SHA256 digest, in the form `sha256:HEX_DIGEST`.
- predicate_type (query): Optional filter for fetching attestations with a given predicate type. This option accepts `provenance`, `sbom`, or freeform text for custom predicate types.

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/autolinks

Get all autolinks of a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/autolinks

Create an autolink reference for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: response
- 422:

GET - /repos/{owner}/{repo}/autolinks/{autolink_id}

Get an autolink reference of a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/autolinks/{autolink_id}

Delete an autolink reference from a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/automated-security-fixes

Check if Dependabot security updates are enabled for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response if Dependabot is enabled
- 404: Not Found if Dependabot is not enabled for the repository

PUT - /repos/{owner}/{repo}/automated-security-fixes

Enable Dependabot security updates

Parameters:

- {}:
- {}:

Responses:

- 204: Response

DELETE - /repos/{owner}/{repo}/automated-security-fixes

Disable Dependabot security updates

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/branches

List branches

Parameters:

- {}:

- {}:
- protected (query): Setting to `true` returns only branches protected by branch protections or rulesets. When set to `false`, only unprotected branches are returned. Omitting this parameter returns all branches.
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/branches/{branch}

Get a branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:

GET - /repos/{owner}/{repo}/branches/{branch}/protection

Get branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /repos/{owner}/{repo}/branches/{branch}/protection

Update branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 422:
- 404:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection

Delete branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/enforce_admins

Get admin branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/branches/{branch}/protection/enforce_admins

Set admin branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/enforce_admins

Delete admin branch protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/required_pull_request_reviews

Get pull request review protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/branches/{branch}/protection/required_pull_request_reviews

Update pull request review protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/required_pull_request_reviews

Delete pull request review protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/required_signatures

Get commit signature protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/branches/{branch}/protection/required_signatures

Create commit signature protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/required_signatures

Delete commit signature protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks

Get status checks protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks

Update status check protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks

Remove status check protection

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks/contexts

Get all status check contexts

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks/contexts

Add status check contexts

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 403:
- 404:

PUT - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks/contexts

Set status check contexts

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/required_status_checks/contexts

Remove status check contexts

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions

Get access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions

Delete access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/apps

Get apps with access to the protected branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/apps

Add app access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

PUT - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/apps

Set app access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/apps

Remove app access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/teams

Get teams with access to the protected branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/teams

Add team access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

PUT - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/teams

Set team access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/teams

Remove team access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/users

Get users with access to the protected branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/users

Add user access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

PUT - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/users

Set user access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/branches/{branch}/protection/restrictions/users

Remove user access restrictions

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

POST - /repos/{owner}/{repo}/branches/{branch}/rename

Rename a branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403:
- 404:
- 422:

POST - /repos/{owner}/{repo}/check-runs

Create a check run

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/check-runs/{check_run_id}

Get a check run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/check-runs/{check_run_id}

Update a check run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/check-runs/{check_run_id}/annotations

List check run annotations

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/check-runs/{check_run_id}/rerequest

Rerequest a check run

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403: Forbidden if the check run is not rerequestable or doesn't belong to the authenticated GitHub App
- 422: Validation error if the check run is not rerequestable
- 404:

POST - /repos/{owner}/{repo}/check-suites

Create a check suite

Parameters:

- {}:
- {}:

Responses:

- 200: Response when the suite already exists
- 201: Response when the suite was created

PATCH - /repos/{owner}/{repo}/check-suites/preferences

Update repository preferences for check suites

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/check-suites/{check_suite_id}

Get a check suite

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/check-suites/{check_suite_id}/check-runs

List check runs in a check suite

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- filter (query): Filters check runs by their `completed_at` timestamp. `latest` returns the most recent check runs.
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/check-suites/{check_suite_id}/rerequest

Rerequest a check suite

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/code-scanning/alerts

List code scanning alerts for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

- {}:
- {}:
- {}:
- {}:
- {}:
- sort (query): The property by which to sort the results.
- state (query): If specified, only code scanning alerts with this state will be returned.
- severity (query): If specified, only code scanning alerts with this severity will be returned.

Responses:

- 200: Response
- 304:
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}

Get a code scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 404:
- 503:

PATCH - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}

Update a code scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}/autofix

Get the status of an autofix for a code scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 403:
- 404:
- 503:

POST - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}/autofix

Create an autofix for a code scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: OK
- 202: Accepted
- 400:
- 403:
- 404:

- 422: Unprocessable Entity
- 503:

POST - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}/autofix/commits

Commit an autofix for a code scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Created
- 400:
- 403:
- 404:
- 422: Unprocessable Entity
- 503:

GET - /repos/{owner}/{repo}/code-scanning/alerts/{alert_number}/instances

List instances of a code scanning alert

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/analyses

List code scanning analyses for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- ref(query): The Git reference for the analyses you want to list. The `ref` for a branch can be formatted either as `refs/heads/` or simply ``. To reference a pull request use `refs/pull//merge`.
- sarif_id(query): Filter analyses belonging to the same SARIF upload.
- {}:
- sort(query): The property by which to sort the results.

Responses:

- 200: Response
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/analyses/{analysis_id}

Get a code scanning analysis for a repository

Parameters:

- {}:
- {}:
- analysis_id(path): The ID of the analysis, as returned from the `GET /repos/{owner}/{repo}/code-scanning/analyses` operation.

Responses:

- 200: Response
- 403:
- 404:
- 503:

DELETE - /repos/{owner}/{repo}/code-scanning/analyses/{analysis_id}

Delete a code scanning analysis from a repository

Parameters:

- {}:
- {}:
- analysis_id (path): The ID of the analysis, as returned from the `GET /repos/{owner}/{repo}/code-scanning/analyses` operation.
- confirm_delete (query): Allow deletion if the specified analysis is the last in a set. If you attempt to delete the final analysis in a set without setting this parameter to `true`, you'll get a 400 response with the message: `Analysis is last of its type and deletion may result in the loss of historical alert data. Please specify confirm_delete.`

Responses:

- 200: Response
- 400:
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/codeql/databases

List CodeQL databases for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/codeql/databases/{language}

Get a CodeQL database for a repository

Parameters:

- {}:
- {}:
- language (path): The language of the CodeQL database.

Responses:

- 200: Response
- 302:
- 403:
- 404:
- 503:

DELETE - /repos/{owner}/{repo}/code-scanning/codeql/databases/{language}

Delete a CodeQL database

Parameters:

- {}:
- {}:
- language (path): The language of the CodeQL database.

Responses:

- 204: Response
- 403:
- 404:
- 503:

POST - /repos/{owner}/{repo}/code-scanning/codeql/variant-analyses

Create a CodeQL variant analysis

Parameters:

- {}:
- {}:

Responses:

- 201: Variant analysis submitted for processing
- 404:
- 422: Unable to process variant analysis submission
- 503:

GET - /repos/{owner}/{repo}/code-scanning/codeql/variant-analyses/{codeql_variant_analysis_id}

Get the summary of a CodeQL variant analysis

Parameters:

- {}:
- {}:
- codeql_variant_analysis_id (path): The unique identifier of the variant analysis.

Responses:

- 200: Response
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/codeql/variant-analyses/{codeql_variant_analysis_id}/repos/{repo_owner}/{repo_name}

Get the analysis status of a repository in a CodeQL variant analysis

Parameters:

- {}:
- repo (path): The name of the controller repository.
- codeql_variant_analysis_id (path): The ID of the variant analysis.
- repo_owner (path): The account owner of the variant analysis repository. The name is not case sensitive.
- repo_name (path): The name of the variant analysis repository.

Responses:

- 200: Response
- 404:
- 503:

GET - /repos/{owner}/{repo}/code-scanning/default-setup

Get a code scanning default setup configuration

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 503:

PATCH - /repos/{owner}/{repo}/code-scanning/default-setup

Update a code scanning default setup configuration

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 202: Response
- 403:
- 404:
- 409:
- 503:

POST - /repos/{owner}/{repo}/code-scanning/sarifs

Upload an analysis as SARIF data

Parameters:

- {}:
- {}:

Responses:

- 202: Response
- 400: Bad Request if the sarif field is invalid
- 403:
- 404:
- 413: Payload Too Large if the sarif field is too large
- 503:

GET - /repos/{owner}/{repo}/code-scanning/sarifs/{sarif_id}

Get information about a SARIF upload

Parameters:

- {}:
- {}:
- sarif_id (path): The SARIF ID obtained after uploading.

Responses:

- 200: Response
- 403:
- 404: Not Found if the sarif id does not match any upload
- 503:

GET - /repos/{owner}/{repo}/code-security-configuration

Get the code security configuration associated with a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 204:
- 304:
- 403:
- 404:

GET - /repos/{owner}/{repo}/codeowners/errors

List CODEOWNERS errors

Parameters:

- {}:
- {}:
- ref (query): A branch, tag or commit name used to determine which version of the CODEOWNERS file to use. Default: the repository's default branch (e.g. `main`)

Responses:

- 200: Response
- 404: Resource not found

GET - /repos/{owner}/{repo}/codespaces

List codespaces in a repository for the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 500:
- 401:
- 403:
- 404:

POST - /repos/{owner}/{repo}/codespaces

Create a codespace in a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Response when the codespace was successfully created
- 202: Response when the codespace creation partially failed but is being retried in the background
- 400:
- 401:
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/codespaces/devcontainers

List devcontainer configurations in a repository for the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 500:
- 400:
- 401:
- 403:
- 404:

GET - /repos/{owner}/{repo}/codespaces/machines

List available machine types for a repository

Parameters:

- {}:
- {}:
- location (query): The location to check for available machines. Assigned by IP if not provided.
- client_ip (query): IP for location auto-detection when proxying a request
- ref (query): The branch or commit to check for prebuild availability and devcontainer restrictions.

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

GET - /repos/{owner}/{repo}/codespaces/new

Get default attributes for a codespace

Parameters:

- {}:
- {}:
- ref (query): The branch or commit to check for a default devcontainer path. If not specified, the default branch will be checked.
- client_ip (query): An alternative IP for default location auto-detection, such as when proxying a request.

Responses:

- 200: Response when a user is able to create codespaces from the repository.
- 401:
- 403:
- 404:

GET - /repos/{owner}/{repo}/codespaces/permissions_check

Check if permissions defined by a devcontainer have been accepted by the authenticated user

Parameters:

- {}:
- {}:
- ref (query): The git reference that points to the location of the devcontainer configuration to use for the permission check. The value of `ref` will typically be a branch name ('heads/BRANCH_NAME'). For more information, see "[Git References](https://git-scm.com/book/en/v2/Git-Internals-Git-References)" in the Git documentation.
- devcontainer_path (query): Path to the devcontainer.json configuration to use for the permission check.

Responses:

- 200: Response when the permission check is successful
- 401:
- 403:
- 404:
- 422:
- 503:

GET - /repos/{owner}/{repo}/codespaces/secrets

List repository secrets

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/codespaces/secrets/public-key

Get a repository public key

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/codespaces/secrets/{secret_name}

Get a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/codespaces/secrets/{secret_name}

Create or update a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /repos/{owner}/{repo}/codespaces/secrets/{secret_name}

Delete a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/collaborators

List repository collaborators

Parameters:

- {}:
- {}:
- affiliation (query): Filter collaborators returned by their affiliation. `outside` means all outside collaborators of an organization-owned repository. `direct` means all collaborators with permissions to an organization-owned repository, regardless of organization membership status. `all` means all collaborators the authenticated user can see.
- permission (query): Filter collaborators by the permissions they have on the repository. If not specified, all collaborators will be returned.
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/collaborators/{username}

Check if a user is a repository collaborator

Parameters:

- {}:
- {}:

- {}:

Responses:

- 204: Response if user is a collaborator
- 404: Not Found if user is not a collaborator

PUT - /repos/{owner}/{repo}/collaborators/{username}

Add a repository collaborator

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response when a new invitation is created
- 204: Response when: - an existing collaborator is added as a collaborator - an organization member is added as an individual collaborator - an existing team member (whose team is also a repository collaborator) is added as an individual collaborator
- 422:
- 403:

DELETE - /repos/{owner}/{repo}/collaborators/{username}

Remove a repository collaborator

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: No Content when collaborator was removed from the repository.
- 422:
- 403:

GET - /repos/{owner}/{repo}/collaborators/{username}/permission

Get repository permissions for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: if user has admin permissions
- 404:

GET - /repos/{owner}/{repo}/comments

List commit comments for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/comments/{comment_id}

Get a commit comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/comments/{comment_id}

Update a commit comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/comments/{comment_id}

Delete a commit comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/comments/{comment_id}/reactions

List reactions for a commit comment

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a commit comment.
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/comments/{comment_id}/reactions

Create reaction for a commit comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Reaction exists
- 201: Reaction created
- 422:

DELETE - /repos/{owner}/{repo}/comments/{comment_id}/reactions/{reaction_id}

Delete a commit comment reaction

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/commits

List commits

Parameters:

- {}:
- {}:
- sha (query): SHA or branch to start listing commits from. Default: the repositoryâ€™s default branch (usually `main`).
- path (query): Only commits containing this file path will be returned.

- author (query): GitHub username or email address to use to filter by commit author.
- committer (query): GitHub username or email address to use to filter by commit committer.
- since (query): Only show results that were last updated after the given time. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format: `YYYY-MM-DDTHH:MM:SSZ`. Due to limitations of Git, timestamps must be between 1970-01-01 and 2099-12-31 (inclusive) or unexpected results may be returned.
- until (query): Only commits before this date will be returned. This is a timestamp in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format: `YYYY-MM-DDTHH:MM:SSZ`. Due to limitations of Git, timestamps must be between 1970-01-01 and 2099-12-31 (inclusive) or unexpected results may be returned.
- {}:
- {}:

Responses:

- 200: Response
- 500:
- 400:
- 404:
- 409:

GET - /repos/{owner}/{repo}/commits/{commit_sha}/branches-where-head

List branches for HEAD commit

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 409:

GET - /repos/{owner}/{repo}/commits/{commit_sha}/comments

List commit comments

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/commits/{commit_sha}/comments

Create a commit comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403:
- 422:

GET - /repos/{owner}/{repo}/commits/{commit_sha}/pulls

List pull requests associated with a commit

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 409:

GET - /repos/{owner}/{repo}/commits/{ref}

Get a commit

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:
- 500:
- 503:
- 409:

GET - /repos/{owner}/{repo}/commits/{ref}/check-runs

List check runs for a Git reference

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- filter (query): Filters check runs by their `completed_at` timestamp. `latest` returns the most recent check runs.
- {}:
- {}:
- app_id (query):

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/commits/{ref}/check-suites

List check suites for a Git reference

Parameters:

- {}:
- {}:
- {}:
- app_id (query): Filters check suites by GitHub App `id`.
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/commits/{ref}/status

Get the combined status for a specific reference

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/commits/{ref}/statuses

List commit statuses for a reference

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:

GET - /repos/{owner}/{repo}/community/profile

Get community profile metrics

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/compare/{basehead}

Compare two commits

Parameters:

- {}:
- {}:
- {}:
- {}:
- basehead (path): The base branch and head branch to compare. This parameter expects the format 'BASE...HEAD'. Both must be branch names in 'repo'. To compare with a branch that exists in a different repository in the same network as 'repo', the 'basehead' parameter expects the format 'USERNAME:BASE...USERNAME:HEAD'.

Responses:

- 200: Response
- 404:
- 500:
- 503:

GET - /repos/{owner}/{repo}/contents/{path}

Get repository content

Parameters:

- {}:
- {}:
- path (path): path parameter
- ref (query): The name of the commit/branch/tag. Default: the repository's default branch.

Responses:

- 200: Response
- 404:
- 403:
- 302:
- 304:

PUT - /repos/{owner}/{repo}/contents/{path}

Create or update file contents

Parameters:

- {}:
- {}:
- path (path): path parameter

Responses:

- 200: Response
- 201: Response
- 404:
- 422:
- 409: Conflict

DELETE - /repos/{owner}/{repo}/contents/{path}

Delete a file

Parameters:

- {}:
- {}:
- path (path): path parameter

Responses:

- 200: Response
- 422:
- 404:
- 409:
- 503:

GET - /repos/{owner}/{repo}/contributors

List repository contributors

Parameters:

- {}:
- {}:
- anon (query): Set to '1' or 'true' to include anonymous contributors in results.
- {}:
- {}:

Responses:

- 200: If repository contains content
- 204: Response if repository is empty
- 403:
- 404:

GET - /repos/{owner}/{repo}/dependabot/alerts

List Dependabot alerts for a repository

Parameters:

- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- 0:
- page (query): ****Closing down notice****. Page number of the results to fetch. Use cursor-based pagination with `before` or `after` instead.
- per_page (query): The number of results per page (max 100). For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- 0:
- 0:
- 0:
- 0:

Responses:

- 200: Response
- 304:
- 400:
- 403:
- 404:
- 422:

GET - /repos/{owner}/{repo}/dependabot/alerts/{alert_number}

Get a Dependabot alert

Parameters:

- 0:
- 0:
- 0:

Responses:

- 200: Response
- 304:
- 403:
- 404:

PATCH - /repos/{owner}/{repo}/dependabot/alerts/{alert_number}

Update a Dependabot alert

Parameters:

- $O:$
- $O:$
- $O:$

Responses:

- 200: Response
- 400:
- 403:
- 404:
- 409:
- 422:

GET - /repos/{owner}/{repo}/dependabot/secrets

List repository secrets

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/dependabot/secrets/public-key

Get a repository public key

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/dependabot/secrets/{secret_name}

Get a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/dependabot/secrets/{secret_name}

Create or update a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /repos/{owner}/{repo}/dependabot/secrets/{secret_name}

Delete a repository secret

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/dependency-graph/compare/{basehead}

Get a diff of the dependencies between commits

Parameters:

- {}:
- {}:
- basehead (path): The base and head Git revisions to compare. The Git revisions will be resolved to commit SHAs. Named revisions will be resolved to their corresponding HEAD commits, and an appropriate merge base will be determined. This parameter expects the format `{base}...{head}`.
- {}:

Responses:

- 200: Response

- 404:
- 403:

GET - /repos/{owner}/{repo}/dependency-graph/sbom

Export a software bill of materials (SBOM) for a repository.

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 403:

POST - /repos/{owner}/{repo}/dependency-graph/snapshots

Create a snapshot of dependencies for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/deployments

List deployments

Parameters:

- {}:
- {}:
- sha (query): The SHA recorded at creation time.
- ref (query): The name of the ref. This can be a branch, tag, or SHA.
- task (query): The name of the task for the deployment (e.g., 'deploy' or 'deploy:migrations').
- environment (query): The name of the environment that was deployed to (e.g., 'staging' or 'production').
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/deployments

Create a deployment

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 202: Merged branch response
- 409: Conflict when there is a merge conflict or the commit's status checks failed
- 422:

GET - /repos/{owner}/{repo}/deployments/{deployment_id}

Get a deployment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/deployments/{deployment_id}

Delete a deployment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/deployments/{deployment_id}/statuses

List deployment statuses

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/deployments/{deployment_id}/statuses

Create a deployment status

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 422:

GET - /repos/{owner}/{repo}/deployments/{deployment_id}/statuses/{status_id}

Get a deployment status

Parameters:

- {}:
- {}:
- {}:
- status_id (path):

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/dispatches

Create a repository dispatch event

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/environments

List environments

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}

Get an environment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/environments/{environment_name}

Create or update an environment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422: Validation error when the environment name is invalid or when 'protected_branches' and 'custom_branch_policies' in 'deployment_branch_policy' are set to the same value

DELETE - /repos/{owner}/{repo}/environments/{environment_name}

Delete an environment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Default response

GET - /repos/{owner}/{repo}/environments/{environment_name}/deployment-branch-policies

List deployment branch policies

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/environments/{environment_name}/deployment-branch-policies

Create a deployment branch policy

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: Not Found or 'deployment_branch_policy.custom_branch_policies' property for the environment is set to false
- 303: Response if the same branch name pattern already exists

GET - /repos/{owner}/{repo}/environments/{environment_name}/deployment-branch-policies/{branch_policy_id}

Get a deployment branch policy

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/environments/{environment_name}/deployment-branch-policies/{branch_policy_id}

Update a deployment branch policy

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/environments/{environment_name}/deployment-branch-policies/{branch_policy_id}

Delete a deployment branch policy

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}/deployment_protection_rules

Get all deployment protection rules for an environment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: List of deployment protection rules

POST - /repos/{owner}/{repo}/environments/{environment_name}/deployment_protection_rules

Create a custom deployment protection rule on an environment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: The enabled custom deployment protection rule

GET - /repos/{owner}/{repo}/environments/{environment_name}/deployment_protection_rules/apps

List custom deployment rule integrations available for an environment

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: A list of custom deployment rule integrations available for this environment.

GET - /repos/{owner}/{repo}/environments/{environment_name}/deployment_protection_rules/{protection_rule_id}

Get a custom deployment protection rule

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/environments/{environment_name}/deployment_protection_rules/{protection_rule_id}

Disable a custom protection rule for an environment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}/secrets

List environment secrets

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}/secrets/public-key

Get an environment public key

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}/secrets/{secret_name}

Get an environment secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/environments/{environment_name}/secrets/{secret_name}

Create or update an environment secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 201: Response when creating a secret
- 204: Response when updating a secret

DELETE - /repos/{owner}/{repo}/environments/{environment_name}/secrets/{secret_name}

Delete an environment secret

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Default response

GET - /repos/{owner}/{repo}/environments/{environment_name}/variables

List environment variables

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/environments/{environment_name}/variables

Create an environment variable

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/environments/{environment_name}/variables/{name}

Get an environment variable

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/environments/{environment_name}/variables/{name}

Update an environment variable

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /repos/{owner}/{repo}/environments/{environment_name}/variables/{name}

Delete an environment variable

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/events

List repository events

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/forks

List forks

Parameters:

- {}:
- {}:
- sort (query): The sort order. `stargazers` will sort by star count.
- {}:
- {}:

Responses:

- 200: Response
- 400:

POST - /repos/{owner}/{repo}/forks

Create a fork

Parameters:

- {}:
- {}:

Responses:

- 202: Response
- 400:
- 422:
- 403:
- 404:

POST - /repos/{owner}/{repo}/git/blobs

Create a blob

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 404:
- 409:
- 403:
- 422: Validation failed

GET - /repos/{owner}/{repo}/git/blobs/{file_sha}

Get a blob

Parameters:

- {}:
- {}:
- file_sha (path):

Responses:

- 200: Response
- 404:
- 422:
- 403:
- 409:

POST - /repos/{owner}/{repo}/git/commits

Create a commit

Parameters:

- {}:

- {}:

Responses:

- 201: Response
- 422:
- 404:
- 409:

GET - /repos/{owner}/{repo}/git/commits/{commit_sha}

Get a commit object

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 409:

GET - /repos/{owner}/{repo}/git/matching-refs/{ref}

List matching references

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 409:

GET - /repos/{owner}/{repo}/git/ref/{ref}

Get a reference

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 409:

POST - /repos/{owner}/{repo}/git/refs

Create a reference

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 409:

PATCH - /repos/{owner}/{repo}/git/refs/{ref}

Update a reference

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 409:

DELETE - /repos/{owner}/{repo}/git/refs/{ref}

Delete a reference

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 422:
- 409:

POST - /repos/{owner}/{repo}/git/tags

Create a tag object

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 409:

GET - /repos/{owner}/{repo}/git/tags/{tag_sha}

Get a tag

Parameters:

- {}:
- {}:
- tag_sha (path):

Responses:

- 200: Response
- 404:
- 409:

POST - /repos/{owner}/{repo}/git/trees

Create a tree

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 404:
- 403:
- 409:

GET - /repos/{owner}/{repo}/git/trees/{tree_sha}

Get a tree

Parameters:

- {}:
- {}:
- tree_sha (path): The SHA1 value or ref(branch or tag) name of the tree.
- recursive (query): Setting this parameter to any value returns the objects or subtrees referenced by the tree specified in `tree_sha`. For example, setting `recursive` to any of the following will enable returning objects or subtrees: `0`, `1`, `true`, and `false`. Omit this parameter to prevent recursively returning objects or subtrees.

Responses:

- 200: Response
- 422:
- 404:
- 409:

GET - /repos/{owner}/{repo}/hooks

List repository webhooks

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/hooks

Create a repository webhook

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 404:
- 422:
- 403:

GET - /repos/{owner}/{repo}/hooks/{hook_id}

Get a repository webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/hooks/{hook_id}

Update a repository webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:

DELETE - /repos/{owner}/{repo}/hooks/{hook_id}

Delete a repository webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/hooks/{hook_id}/config

Get a webhook configuration for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/hooks/{hook_id}/config

Update a webhook configuration for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/hooks/{hook_id}/deliveries

List deliveries for a repository webhook

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:

GET - /repos/{owner}/{repo}/hooks/{hook_id}/deliveries/{delivery_id}

Get a delivery for a repository webhook

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:

POST - /repos/{owner}/{repo}/hooks/{hook_id}/deliveries/{delivery_id}/attempts

Redeliver a delivery for a repository webhook

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 202:
- 400:
- 422:

POST - /repos/{owner}/{repo}/hooks/{hook_id}/pings

Ping a repository webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

POST - /repos/{owner}/{repo}/hooks/{hook_id}/tests

Test the push repository webhook

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/import

Get an import status

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 503:

PUT - /repos/{owner}/{repo}/import

Start an import

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 404:
- 503:

PATCH - /repos/{owner}/{repo}/import

Update an import

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 503:

DELETE - /repos/{owner}/{repo}/import

Cancel an import

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 503:

GET - /repos/{owner}/{repo}/import/authors

Get commit authors

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 503:

PATCH - /repos/{owner}/{repo}/import/authors/{author_id}

Map a commit author

Parameters:

- {}:
- {}:
- author_id (path):

Responses:

- 200: Response
- 422:
- 404:
- 503:

GET - /repos/{owner}/{repo}/import/large_files

Get large files

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 503:

PATCH - /repos/{owner}/{repo}/import/lfs

Update Git LFS preference

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 422:
- 503:

GET - /repos/{owner}/{repo}/installation

Get a repository installation for the authenticated app

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:

GET - /repos/{owner}/{repo}/interaction-limits

Get interaction restrictions for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/interaction-limits

Set interaction restrictions for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 409: Response

DELETE - /repos/{owner}/{repo}/interaction-limits

Remove interaction restrictions for a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 409: Response

GET - /repos/{owner}/{repo}/invitations

List repository invitations

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /repos/{owner}/{repo}/invitations/{invitation_id}

Update a repository invitation

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/invitations/{invitation_id}

Delete a repository invitation

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/issues

List repository issues

Parameters:

- {}:
- {}:
- milestone (query): If an `integer` is passed, it should refer to a milestone by its `number` field. If the string `*` is passed, issues with any milestone are accepted. If the string `none` is passed, issues without milestones are returned.
- state (query): Indicates the state of the issues to return.
- assignee (query): Can be the name of a user. Pass in `none` for issues with no assigned user, and `*` for issues assigned to any user.
- type (query): Can be the name of an issue type. If the string `*` is passed, issues with any type are accepted. If the string `none` is passed, issues without type are returned.
- creator (query): The user that created the issue.
- mentioned (query): A user that's mentioned in the issue.
- {}:
- sort (query): What to sort results by.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 422:
- 404:

POST - /repos/{owner}/{repo}/issues

Create an issue

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 400:
- 403:
- 422:
- 503:
- 404:
- 410:

GET - /repos/{owner}/{repo}/issues/comments

List issue comments for a repository

Parameters:

- {}:
- {}:
- {}:
- direction (query): Either `asc` or `desc`. Ignored without the `sort` parameter.
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:

GET - /repos/{owner}/{repo}/issues/comments/{comment_id}

Get an issue comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/issues/comments/{comment_id}

Update an issue comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/issues/comments/{comment_id}

Delete an issue comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/issues/comments/{comment_id}/reactions

List reactions for an issue comment

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to an issue comment.

- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/issues/comments/{comment_id}/reactions

Create reaction for an issue comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Reaction exists
- 201: Reaction created
- 422:

DELETE - /repos/{owner}/{repo}/issues/comments/{comment_id}/reactions/{reaction_id}

Delete an issue comment reaction

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/issues/events

List issue events for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/issues/events/{event_id}

Get an issue event

Parameters:

- {}:
- {}:
- event_id (path):

Responses:

- 200: Response
- 404:
- 410:
- 403:

GET - /repos/{owner}/{repo}/issues/{issue_number}

Get an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:

- 410:
- 304:

PATCH - /repos/{owner}/{repo}/issues/{issue_number}

Update an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 503:
- 403:
- 301:
- 404:
- 410:

POST - /repos/{owner}/{repo}/issues/{issue_number}/assignees

Add assignees to an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/assignees

Remove assignees from an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/issues/{issue_number}/assignees/{assignee}

Check if a user can be assigned to a issue

Parameters:

- {}:
- {}:
- {}:
- assignee (path):

Responses:

- 204: Response if `assignee` can be assigned to `issue_number`
- 404: Response if `assignee` can not be assigned to `issue_number`

GET - /repos/{owner}/{repo}/issues/{issue_number}/comments

List issue comments

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 410:

POST - /repos/{owner}/{repo}/issues/{issue_number}/comments

Create an issue comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403:
- 410:
- 422:
- 404:

GET - /repos/{owner}/{repo}/issues/{issue_number}/events

List issue events

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 410:

GET - /repos/{owner}/{repo}/issues/{issue_number}/labels

List labels for an issue

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:
- 410:

POST - /repos/{owner}/{repo}/issues/{issue_number}/labels

Add labels to an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:
- 410:
- 422:

PUT - /repos/{owner}/{repo}/issues/{issue_number}/labels

Set labels for an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 301:
- 404:
- 410:
- 422:

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/labels

Remove all labels from an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 301:
- 404:
- 410:

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/labels/{name}

Remove a label from an issue

Parameters:

- {}:
- {}:
- {}:
- name (path):

Responses:

- 200: Response
- 301:
- 404:
- 410:

PUT - /repos/{owner}/{repo}/issues/{issue_number}/lock

Lock an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403:
- 410:
- 404:
- 422:

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/lock

Unlock an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403:
- 404:

GET - /repos/{owner}/{repo}/issues/{issue_number}/reactions

List reactions for an issue

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to an issue.
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 410:

POST - /repos/{owner}/{repo}/issues/{issue_number}/reactions

Create reaction for an issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 201: Response
- 422:

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/reactions/{reaction_id}

Delete an issue reaction

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

DELETE - /repos/{owner}/{repo}/issues/{issue_number}/sub_issue

Remove sub-issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:
- 404:

GET - /repos/{owner}/{repo}/issues/{issue_number}/sub_issues

List sub-issues

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 410:

POST - /repos/{owner}/{repo}/issues/{issue_number}/sub_issues

Add sub-issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 403:
- 410:
- 422:
- 404:

PATCH - /repos/{owner}/{repo}/issues/{issue_number}/sub_issues/priority

Reprioritize sub-issue

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 422:
- 503:

GET - /repos/{owner}/{repo}/issues/{issue_number}/timeline

List timeline events for an issue

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 410:

GET - /repos/{owner}/{repo}/keys

List deploy keys

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/keys

Create a deploy key

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:

GET - /repos/{owner}/{repo}/keys/{key_id}

Get a deploy key

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

DELETE - /repos/{owner}/{repo}/keys/{key_id}

Delete a deploy key

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/labels

List labels for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/labels

Create a label

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 404:

GET - /repos/{owner}/{repo}/labels/{name}

Get a label

Parameters:

- {}:
- {}:
- name (path):

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/labels/{name}

Update a label

Parameters:

- {}:
- {}:
- name (path):

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/labels/{name}

Delete a label

Parameters:

- {}:
- {}:
- name (path):

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/languages

List repository languages

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/license

Get the license for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/merge-upstream

Sync a fork branch with the upstream repository

Parameters:

- {}:
- {}:

Responses:

- 200: The branch has been successfully synced with the upstream repository
- 409: The branch could not be synced because of a merge conflict
- 422: The branch could not be synced for some other reason

POST - /repos/{owner}/{repo}/merges

Merge a branch

Parameters:

- {}:
- {}:

Responses:

- 201: Successful Response (The resulting merge commit)
- 204: Response when already merged
- 404: Not Found when the base or head does not exist
- 409: Conflict when there is a merge conflict
- 403:
- 422:

GET - /repos/{owner}/{repo}/milestones

List milestones

Parameters:

- {}:
- {}:
- state (query): The state of the milestone. Either `open`, `closed`, or `all`.
- sort (query): What to sort results by. Either `due_on` or `completeness`.
- direction (query): The direction of the sort. Either `asc` or `desc`.
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/milestones

Create a milestone

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/milestones/{milestone_number}

Get a milestone

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/milestones/{milestone_number}

Update a milestone

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/milestones/{milestone_number}

Delete a milestone

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/milestones/{milestone_number}/labels

List labels for issues in a milestone

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/notifications

List repository notifications for the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

PUT - /repos/{owner}/{repo}/notifications

Mark repository notifications as read

Parameters:

- {}:
- {}:

Responses:

- 202: Response

- 205: Reset Content

GET - /repos/{owner}/{repo}/pages

Get a GitHub Pages site

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/pages

Create a GitHub Pages site

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 422:
- 409:

PUT - /repos/{owner}/{repo}/pages

Update information about a GitHub Pages site

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422:
- 400:
- 409:

DELETE - /repos/{owner}/{repo}/pages

Delete a GitHub Pages site

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 422:
- 404:
- 409:

GET - /repos/{owner}/{repo}/pages/builds

List GitHub Pages builds

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/pages/builds

Request a GitHub Pages build

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/pages/builds/latest

Get latest Pages build

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/pages/builds/{build_id}

Get GitHub Pages build

Parameters:

- {}:
- {}:
- build_id (path):

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/pages/deployments

Create a GitHub Pages deployment

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 400:
- 422:
- 404:

GET - /repos/{owner}/{repo}/pages/deployments/{pages_deployment_id}

Get the status of a GitHub Pages deployment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/pages/deployments/{pages_deployment_id}/cancel

Cancel a GitHub Pages deployment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204:
- 404:

GET - /repos/{owner}/{repo}/pages/health

Get a DNS health check for GitHub Pages

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 202: Empty response
- 400: Custom domains are not available for GitHub Pages
- 422: There isn't a CNAME for this page
- 404:

GET - /repos/{owner}/{repo}/private-vulnerability-reporting

Check if private vulnerability reporting is enabled for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Private vulnerability reporting status
- 422:

PUT - /repos/{owner}/{repo}/private-vulnerability-reporting

Enable private vulnerability reporting for a repository

Parameters:

- {}:
- {}:

Responses:

- 204:
- 422:

DELETE - /repos/{owner}/{repo}/private-vulnerability-reporting

Disable private vulnerability reporting for a repository

Parameters:

- {}:
- {}:

Responses:

- 204:
- 422:

GET - /repos/{owner}/{repo}/projects

List repository projects

Parameters:

- {}:
- {}:
- state (query): Indicates the state of the projects to return.
- {}:
- {}:

Responses:

- 200: Response
- 401:
- 403:
- 404:
- 410:
- 422:

POST - /repos/{owner}/{repo}/projects

Create a repository project

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 401:
- 403:
- 404:
- 410:
- 422:

GET - /repos/{owner}/{repo}/properties/values

Get all custom property values for a repository

Parameters:

- ():
- ():

Responses:

- 200: Response
- 403:
- 404:

PATCH - /repos/{owner}/{repo}/properties/values

Create or update custom property values for a repository

Parameters:

- ():
- ():

Responses:

- 204: No Content when custom property values are successfully created or updated
- 403:
- 404:
- 422:

GET - /repos/{owner}/{repo}/pulls

List pull requests

Parameters:

- ():
- ():
- state (query): Either `open`, `closed`, or `all` to filter by state.
- head (query): Filter pulls by head user or head organization and branch name in the format of `user:ref-name` or `organization:ref-name`. For example: `github:new-script-format` or `octocat:test-branch`.
- base (query): Filter pulls by base branch name. Example: `gh-pages`.
- sort (query): What to sort results by. `popularity` will sort by the number of comments. `long-running` will sort by date created and will limit the results to pull requests that have been open for more than a month and have had activity within the past month.
- direction (query): The direction of the sort. Default: `desc` when sort is `created` or sort is not specified, otherwise `asc`.
- ():
- ():

Responses:

- 200: Response
- 304:
- 422:

POST - /repos/{owner}/{repo}/pulls

Create a pull request

Parameters:

- ():
- ():

Responses:

- 201: Response
- 403:
- 422:

GET - /repos/{owner}/{repo}/pulls/comments

List review comments in a repository

Parameters:

- ():
- ():
- sort (query):
- direction (query): The direction to sort results. Ignored without `sort` parameter.
- ():
- ():
- ():

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/pulls/comments/{comment_id}

Get a review comment for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PATCH - /repos/{owner}/{repo}/pulls/comments/{comment_id}

Update a review comment for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/pulls/comments/{comment_id}

Delete a review comment for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:

GET - /repos/{owner}/{repo}/pulls/comments/{comment_id}/reactions

List reactions for a pull request review comment

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a pull request review comment.
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/pulls/comments/{comment_id}/reactions

Create reaction for a pull request review comment

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Reaction exists
- 201: Reaction created
- 422:

DELETE - /repos/{owner}/{repo}/pulls/comments/{comment_id}/reactions/{reaction_id}

Delete a pull request comment reaction

Parameters:

- {}:
- {}:
- {}:

- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/pulls/{pull_number}

Get a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Pass the appropriate [media type](https://docs.github.com/rest/using-the-rest-api/getting-started-with-the-rest-api#media-types) to fetch diff and patch formats.
- 304:
- 404:
- 406:
- 500:
- 503:

PATCH - /repos/{owner}/{repo}/pulls/{pull_number}

Update a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 403:

POST - /repos/{owner}/{repo}/pulls/{pull_number}/codespaces

Create a codespace from a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response when the codespace was successfully created
- 202: Response when the codespace creation partially failed but is being retried in the background
- 401:
- 403:
- 404:
- 503:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/comments

List review comments on a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:
- direction (query): The direction to sort results. Ignored without `sort` parameter.
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/pulls/{pull_number}/comments

Create a review comment for a pull request

Parameters:

- {}:
- {}:

- {}:

Responses:

- 201: Response
- 422:
- 403:

POST - /repos/{owner}/{repo}/pulls/{pull_number}/comments/{comment_id}/replies

Create a reply for a review comment

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 201: Response
- 404:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/commits

List commits on a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/pulls/{pull_number}/files

List pull requests files

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 500:
- 503:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/merge

Check if a pull request has been merged

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response if pull request has been merged
- 404: Not Found if pull request has not been merged

PUT - /repos/{owner}/{repo}/pulls/{pull_number}/merge

Merge a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: if merge was successful

- 405: Method Not Allowed if merge cannot be performed
- 409: Conflict if sha was provided and pull request head did not match
- 422:
- 403:
- 404:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/requested_reviewers

Get all requested reviewers for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/pulls/{pull_number}/requested_reviewers

Request reviewers for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response
- 422: Unprocessable Entity if user is not a collaborator
- 403:

DELETE - /repos/{owner}/{repo}/pulls/{pull_number}/requested_reviewers

Remove requested reviewers from a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/reviews

List reviews for a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: The list of reviews returns in chronological order.

POST - /repos/{owner}/{repo}/pulls/{pull_number}/reviews

Create a review for a pull request

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 403:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}

Get a review for a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}

Update a review for a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

DELETE - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}

Delete a pending review for a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 404:

GET - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}/comments

List comments for a pull request review

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}/dismissals

Dismiss a review for a pull request

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

POST - /repos/{owner}/{repo}/pulls/{pull_number}/reviews/{review_id}/events

Submit a review for a pull request

Parameters:

- {}:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:
- 403:

PUT - /repos/{owner}/{repo}/pulls/{pull_number}/update-branch

Update a pull request branch

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202: Response
- 422:
- 403:

GET - /repos/{owner}/{repo}/readme

Get a repository README

Parameters:

- {}:
- {}:
- ref(query): The name of the commit/branch/tag. Default: the repositoryâ€™s default branch.

Responses:

- 200: Response
- 304:
- 404:
- 422:

GET - /repos/{owner}/{repo}/readme/{dir}

Get a repository README for a directory

Parameters:

- {}:
- {}:
- dir (path): The alternate path to look for a README file
- ref(query): The name of the commit/branch/tag. Default: the repositoryâ€™s default branch.

Responses:

- 200: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/releases

List releases

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/releases

Create a release

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 404: Not Found if the discussion category name is invalid
- 422:

GET - /repos/{owner}/{repo}/releases/assets/{asset_id}

Get a release asset

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 302:

PATCH - /repos/{owner}/{repo}/releases/assets/{asset_id}

Update a release asset

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/releases/assets/{asset_id}

Delete a release asset

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

POST - /repos/{owner}/{repo}/releases/generate-notes

Generate release notes content for a release

Parameters:

- {}:
- {}:

Responses:

- 200: Name and body of generated release notes
- 404:

GET - /repos/{owner}/{repo}/releases/latest

Get the latest release

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/releases/tags/{tag}

Get a release by tag name

Parameters:

- {}:
- {}:
- tag (path): tag parameter

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/releases/{release_id}

Get a release

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: **Note:** This returns an `upload_url` key corresponding to the endpoint for uploading release assets. This key is a hypermedia resource. For more information, see "[Getting started with the REST API](https://docs.github.com/rest/using-the-rest-api/getting-started-with-the-rest-api#hypermedia)."
- 401: Unauthorized

PATCH - /repos/{owner}/{repo}/releases/{release_id}

Update a release

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: Not Found if the discussion category name is invalid

DELETE - /repos/{owner}/{repo}/releases/{release_id}

Delete a release

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/releases/{release_id}/assets

List release assets

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /repos/{owner}/{repo}/releases/{release_id}/assets

Upload a release asset

Parameters:

- {}:
- {}:
- {}:
- name (query):
- label (query):

Responses:

- 201: Response for successful upload
- 422: Response if you upload an asset with the same filename as another uploaded asset

GET - /repos/{owner}/{repo}/releases/{release_id}/reactions

List reactions for a release

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a release.
- {}:
- {}:

Responses:

- 200: Response
- 404:

POST - /repos/{owner}/{repo}/releases/{release_id}/reactions

Create reaction for a release

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Reaction exists
- 201: Reaction created
- 422:

DELETE - /repos/{owner}/{repo}/releases/{release_id}/reactions/{reaction_id}

Delete a release reaction

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/rules/branches/{branch}

Get rules for a branch

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/rulesets

Get all repository rulesets

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- includes_parents (query): Include rulesets configured at higher levels that apply to this repository
- {}:

Responses:

- 200: Response
- 404:
- 500:

POST - /repos/{owner}/{repo}/rulesets

Create a repository ruleset

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/rulesets/rule-suites

List repository rule suites

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/rulesets/rule-suites/{rule_suite_id}

Get a repository rule suite

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/rulesets/{ruleset_id}

Get a repository ruleset

Parameters:

- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.
- includes_parents (query): Include rulesets configured at higher levels that apply to this repository

Responses:

- 200: Response
- 404:
- 500:

PUT - /repos/{owner}/{repo}/rulesets/{ruleset_id}

Update a repository ruleset

Parameters:

- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 200: Response
- 404:
- 500:

DELETE - /repos/{owner}/{repo}/rulesets/{ruleset_id}

Delete a repository ruleset

Parameters:

- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 204: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/rulesets/{ruleset_id}/history

Get repository ruleset history

Parameters:

- {}:
- {}:
- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.

Responses:

- 200: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/rulesets/{ruleset_id}/history/{version_id}

Get repository ruleset version

Parameters:

- {}:
- {}:
- ruleset_id (path): The ID of the ruleset.
- version_id (path): The ID of the version

Responses:

- 200: Response
- 404:
- 500:

GET - /repos/{owner}/{repo}/secret-scanning/alerts

List secret scanning alerts for a repository

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: Repository is public or secret scanning is disabled for the repository
- 503:

GET - /repos/{owner}/{repo}/secret-scanning/alerts/{alert_number}

Get a secret scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404: Repository is public, or secret scanning is disabled for the repository, or the resource is not found
- 503:

PATCH - /repos/{owner}/{repo}/secret-scanning/alerts/{alert_number}

Update a secret scanning alert

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400: Bad request, resolution comment is invalid or the resolution was not changed.
- 404: Repository is public, or secret scanning is disabled for the repository, or the resource is not found
- 422: State does not match the resolution or resolution comment
- 503:

GET - /repos/{owner}/{repo}/secret-scanning/alerts/{alert_number}/locations

List locations for a secret scanning alert

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404: Repository is public, or secret scanning is disabled for the repository, or the resource is not found
- 503:

POST - /repos/{owner}/{repo}/secret-scanning/push-protection-bypasses

Create a push protection bypass

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403: User does not have enough permissions to perform this action.
- 404: Placeholder ID not found, or push protection is disabled on this repository.
- 422: Bad request, input data missing or incorrect.
- 503:

GET - /repos/{owner}/{repo}/secret-scanning/scan-history

Get secret scanning scan history for a repository

Parameters:

- {}:
- {}:

Responses:

- 404: Repository does not have GitHub Advanced Security or secret scanning enabled
- 503:
- 200: Response

GET - /repos/{owner}/{repo}/security-advisories

List repository security advisories

Parameters:

- {}:
- {}:
- {}:
- sort (query): The property to sort the results by.
- {}:
- {}:
- per_page (query): The number of advisories to return per page. For more information, see "[Using pagination in the REST API](https://docs.github.com/rest/using-the-rest-api/using-pagination-in-the-rest-api)."
- state (query): Filter by state of the repository advisories. Only advisories of this state will be returned.

Responses:

- 200: Response
- 400:
- 404:

POST - /repos/{owner}/{repo}/security-advisories

Create a repository security advisory

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 403:
- 404:
- 422:

POST - /repos/{owner}/{repo}/security-advisories/reports

Privately report a security vulnerability

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 403:
- 404:
- 422:

GET - /repos/{owner}/{repo}/security-advisories/{ghsa_id}

Get a repository security advisory

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

PATCH - /repos/{owner}/{repo}/security-advisories/{ghsa_id}

Update a repository security advisory

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:
- 422: Validation failed, or the endpoint has been spanned.

POST - /repos/{owner}/{repo}/security-advisories/{ghsa_id}/cve

Request a CVE for a repository security advisory

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202:
- 400:
- 403:
- 404:
- 422:

POST - /repos/{owner}/{repo}/security-advisories/{ghsa_id}/forks

Create a temporary private fork

Parameters:

- {}:
- {}:
- {}:

Responses:

- 202: Response
- 400:
- 422:
- 403:
- 404:

GET - /repos/{owner}/{repo}/stargazers

List stargazers

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /repos/{owner}/{repo}/stats/code_frequency

Get the weekly commit activity

Parameters:

- {}:
- {}:

Responses:

- 200: Returns a weekly aggregate of the number of additions and deletions pushed to a repository.
- 202:
- 204:
- 422: Repository contains more than 10,000 commits

GET - /repos/{owner}/{repo}/stats/commit_activity

Get the last year of commit activity

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 202:
- 204:

GET - /repos/{owner}/{repo}/stats/contributors

Get all contributor commit activity

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 202:
- 204:

GET - /repos/{owner}/{repo}/stats/participation

Get the weekly commit count

Parameters:

- {}:
- {}:

Responses:

- 200: The array order is oldest week (index 0) to most recent week.
- 404:

GET - /repos/{owner}/{repo}/stats/punch_card

Get the hourly commit count for each day

Parameters:

- {}:
- {}:

Responses:

- 200: For example, `[2, 14, 25]` indicates that there were 25 total commits, during the 2:00pm hour on Tuesdays. All times are based on the time zone of individual commits.
- 204:

POST - /repos/{owner}/{repo}/statuses/{sha}

Create a commit status

Parameters:

- {}:
- {}:
- sha (path):

Responses:

- 201: Response

GET - /repos/{owner}/{repo}/subscribers

List watchers

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/subscription

Get a repository subscription

Parameters:

- {}:
- {}:

Responses:

- 200: if you subscribe to the repository
- 404: Not Found if you don't subscribe to the repository
- 403:

PUT - /repos/{owner}/{repo}/subscription

Set a repository subscription

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /repos/{owner}/{repo}/subscription

Delete a repository subscription

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/tags

List repository tags

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /repos/{owner}/{repo}/tags/protection

Closing down - List tag protection states for a repository

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:
- 404:

POST - /repos/{owner}/{repo}/tags/protection

Closing down - Create a tag protection state for a repository

Parameters:

- {}:
- {}:

Responses:

- 201: Response
- 403:
- 404:

DELETE - /repos/{owner}/{repo}/tags/protection/{tag_protection_id}

Closing down - Delete a tag protection state for a repository

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 403:
- 404:

GET - /repos/{owner}/{repo}/tarball/{ref}

Download a repository archive (tar)

Parameters:

- {}:
- {}:
- ref (path):

Responses:

- 302: Response

GET - /repos/{owner}/{repo}/teams

List repository teams

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /repos/{owner}/{repo}/topics

Get all repository topics

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /repos/{owner}/{repo}/topics

Replace all repository topics

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:
- 422:

GET - /repos/{owner}/{repo}/traffic/clones

Get repository clones

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:

GET - /repos/{owner}/{repo}/traffic/popular/paths

Get top referral paths

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:

GET - /repos/{owner}/{repo}/traffic/popular/referrers

Get top referral sources

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403:

GET - /repos/{owner}/{repo}/traffic/views

Get page views

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

- 403:

POST - /repos/{owner}/{repo}/transfer

Transfer a repository

Parameters:

- {}:
- {}:

Responses:

- 202: Response

GET - /repos/{owner}/{repo}/vulnerability-alerts

Check if vulnerability alerts are enabled for a repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response if repository is enabled with vulnerability alerts
- 404: Not Found if repository is not enabled with vulnerability alerts

PUT - /repos/{owner}/{repo}/vulnerability-alerts

Enable vulnerability alerts

Parameters:

- {}:
- {}:

Responses:

- 204: Response

DELETE - /repos/{owner}/{repo}/vulnerability-alerts

Disable vulnerability alerts

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /repos/{owner}/{repo}/zipball/{ref}

Download a repository archive (zip)

Parameters:

- {}:
- {}:
- ref(path):

Responses:

- 302: Response

POST - /repos/{template_owner}/{template_repo}/generate

Create a repository using a template

Parameters:

- template_owner (path): The account owner of the template repository. The name is not case sensitive.
- template_repo (path): The name of the template repository without the '.git' extension. The name is not case sensitive.

Responses:

- 201: Response

GET - /repositories

List public repositories

Parameters:

- {}:

Responses:

- 200: Response
- 422:
- 304:

GET - /search/code

Search code

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query). See "[Searching code](https://docs.github.com/search-github/searching-on-github/searching-code)" for a detailed list of qualifiers.
- sort (query): ****This field is closing down.**** Sorts the results of your query. Can only be `indexed`, which indicates how recently a file has been indexed by the GitHub search infrastructure. Default: [best match](https://docs.github.com/rest/search/search#ranking-search-results)
- order (query): ****This field is closing down.**** Determines whether the first search result returned is the highest number of matches (`desc`) or lowest number of matches (`asc`). This parameter is ignored unless you provide `sort`.
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 503:
- 422:
- 403:

GET - /search/commits

Search commits

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query). See "[Searching commits](https://docs.github.com/search-github/searching-on-github/searching-commits)" for a detailed list of qualifiers.
- sort (query): Sorts the results of your query by `author-date` or `committer-date`. Default: [best match](https://docs.github.com/rest/search/search#ranking-search-results)
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:

GET - /search/issues

Search issues and pull requests

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query). See "[Searching issues and pull requests](https://docs.github.com/search-github/searching-on-github/searching-issues-and-pull-requests)" for a detailed list of qualifiers.
- sort (query): Sorts the results of your query by the number of `comments`, `reactions`, `reactions+1`, `reactions-1`, `reactions-smile`, `reactions-thinking_face`, `reactions-heart`, `reactions-tada`, or `interactions`. You can also sort results by how recently the items were `created` or `updated`. Default: [best match](https://docs.github.com/rest/search/search#ranking-search-results)
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 503:
- 422:
- 304:
- 403:

GET - /search/labels

Search labels

Parameters:

- repository_id (query): The id of the repository.
- q (query): The search keywords. This endpoint does not accept qualifiers in the query. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query).
- sort (query): Sorts the results of your query by when the label was `created` or `updated`. Default: [best match](https://docs.github.com/rest/search/search#ranking-search-results)
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 422:

GET - /search/repositories

Search repositories

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query). See "[Searching for repositories](https://docs.github.com/articles/searching-for-repositories/)" for a detailed list of qualifiers.
- sort (query): Sorts the results of your query by number of `stars`, `forks`, or `help-wanted-issues` or how recently the items were `updated`. Default: [best match] (https://docs.github.com/rest/search/search#ranking-search-results)
- ():
- ():
- ():

Responses:

- 200: Response
- 503:
- 422:
- 304:

GET - /search/topics

Search topics

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query).
- ():
- ():

Responses:

- 200: Response
- 304:

GET - /search/users

Search users

Parameters:

- q (query): The query contains one or more search keywords and qualifiers. Qualifiers allow you to limit your search to specific areas of GitHub. The REST API supports the same qualifiers as the web interface for GitHub. To learn more about the format of the query, see [Constructing a search query](https://docs.github.com/rest/search/search#constructing-a-search-query). See "[Searching users](https://docs.github.com/search-github/searching-on-github/searching-users)" for a detailed list of qualifiers.
- sort (query): Sorts the results of your query by number of `followers` or `repositories`, or when the person `joined` GitHub. Default: [best match] (https://docs.github.com/rest/search/search#ranking-search-results)
- ():
- ():
- ():

Responses:

- 200: Response
- 304:
- 503:
- 422:

GET - /teams/{team_id}

Get a team (Legacy)

Parameters:

- ():

Responses:

- 200: Response
- 404:

PATCH - /teams/{team_id}

Update a team (Legacy)

Parameters:

- {}:

Responses:

- 200: Response when the updated information already exists
- 201: Response
- 404:
- 422:
- 403:

DELETE - /teams/{team_id}

Delete a team (Legacy)

Parameters:

- {}:

Responses:

- 204: Response
- 404:
- 422:

GET - /teams/{team_id}/discussions

List discussions (Legacy)

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /teams/{team_id}/discussions

Create a discussion (Legacy)

Parameters:

- {}:

Responses:

- 201: Response

GET - /teams/{team_id}/discussions/{discussion_number}

Get a discussion (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 200: Response

PATCH - /teams/{team_id}/discussions/{discussion_number}

Update a discussion (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /teams/{team_id}/discussions/{discussion_number}

Delete a discussion (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response

GET - /teams/{team_id}/discussions/{discussion_number}/comments

List discussion comments (Legacy)

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

POST - /teams/{team_id}/discussions/{discussion_number}/comments

Create a discussion comment (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /teams/{team_id}/discussions/{discussion_number}/comments/{comment_number}

Get a discussion comment (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

PATCH - /teams/{team_id}/discussions/{discussion_number}/comments/{comment_number}

Update a discussion comment (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /teams/{team_id}/discussions/{discussion_number}/comments/{comment_number}

Delete a discussion comment (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /teams/{team_id}/discussions/{discussion_number}/comments/{comment_number}/reactions

List reactions for a team discussion comment (Legacy)

Parameters:

- {}:
- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a team discussion comment.
- {}:
- {}:

Responses:

- 200: Response

POST - /teams/{team_id}/discussions/{discussion_number}/comments/{comment_number}/reactions

Create reaction for a team discussion comment (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 201: Response

GET - /teams/{team_id}/discussions/{discussion_number}/reactions

List reactions for a team discussion (Legacy)

Parameters:

- {}:
- {}:
- content (query): Returns a single [reaction type](https://docs.github.com/rest/reactions/reactions#about-reactions). Omit this parameter to list all reactions to a team discussion.
- {}:
- {}:

Responses:

- 200: Response

POST - /teams/{team_id}/discussions/{discussion_number}/reactions

Create reaction for a team discussion (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 201: Response

GET - /teams/{team_id}/invitations

List pending team invitations (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /teams/{team_id}/members

List team members (Legacy)

Parameters:

- {}:
- role (query): Filters members returned by their role in the team.
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /teams/{team_id}/members/{username}

Get team member (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: if user is a member

- 404: if user is not a member

PUT - /teams/{team_id}/members/{username}

Add team member (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404: Not Found if team synchronization is set up
- 422: Unprocessable Entity if you attempt to add an organization to a team or you attempt to add a user to a team when they are not a member of at least one other team in the same organization
- 403:

DELETE - /teams/{team_id}/members/{username}

Remove team member (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404: Not Found if team synchronization is setup

GET - /teams/{team_id}/memberships/{username}

Get team membership for a user (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

PUT - /teams/{team_id}/memberships/{username}

Add or update team membership for a user (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 403: Forbidden if team synchronization is set up
- 422: Unprocessable Entity if you attempt to add an organization to a team
- 404:

DELETE - /teams/{team_id}/memberships/{username}

Remove team membership for a user (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403: if team synchronization is set up

GET - /teams/{team_id}/projects

List team projects (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /teams/{team_id}/projects/{project_id}

Check team permissions for a project (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404: Not Found if project is not managed by this team

PUT - /teams/{team_id}/projects/{project_id}

Add or update team project permissions (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403: Forbidden if the project is not owned by the organization
- 404:
- 422:

DELETE - /teams/{team_id}/projects/{project_id}

Remove a project from a team (Legacy)

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 422:

GET - /teams/{team_id}/repos

List team repositories (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /teams/{team_id}/repos/{owner}/{repo}

Check team permissions for a repository (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Alternative response with extra repository information
- 204: Response if repository is managed by this team
- 404: Not Found if repository is not managed by this team

PUT - /teams/{team_id}/repos/{owner}/{repo}

Add or update team repository permissions (Legacy)

Parameters:

- {}:
- {}:

- {}:

Responses:

- 204: Response
- 403:
- 422:

DELETE - /teams/{team_id}/repos/{owner}/{repo}

Remove a repository from a team (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response

GET - /teams/{team_id}/teams

List child teams (Legacy)

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: if child teams exist
- 404:
- 403:
- 422:

GET - /user

Get the authenticated user

Responses:

- 200: Response
- 304:
- 403:
- 401:

PATCH - /user

Update the authenticated user

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:
- 422:

GET - /user/blocks

List users blocked by the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/blocks/{username}

Check if a user is blocked by the authenticated user

Parameters:

- {}:

Responses:

- 204: If the user is blocked
- 404: If the user is not blocked
- 304:
- 403:
- 401:

PUT - /user/blocks/{username}

Block a user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:
- 422:

DELETE - /user/blocks/{username}

Unblock a user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 403:
- 401:
- 404:

GET - /user/codespaces

List codespaces for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

POST - /user/codespaces

Create a codespace for the authenticated user

Responses:

- 201: Response when the codespace was successfully created
- 202: Response when the codespace creation partially failed but is being retried in the background
- 401:
- 403:
- 404:
- 503:

GET - /user/codespaces/secrets

List secrets for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response

GET - /user/codespaces/secrets/public-key

Get public key for the authenticated user

Responses:

- 200: Response

GET - /user/codespaces/secrets/{secret_name}

Get a secret for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response

PUT - /user/codespaces/secrets/{secret_name}

Create or update a secret for the authenticated user

Parameters:

- {}:

Responses:

- 201: Response after successfully creating a secret
- 204: Response after successfully updating a secret
- 422:
- 404:

DELETE - /user/codespaces/secrets/{secret_name}

Delete a secret for the authenticated user

Parameters:

- {}:

Responses:

- 204: Response

GET - /user/codespaces/secrets/{secret_name}/repositories

List selected repositories for a user secret

Parameters:

- {}:

Responses:

- 200: Response
- 401:
- 403:
- 404:
- 500:

PUT - /user/codespaces/secrets/{secret_name}/repositories

Set selected repositories for a user secret

Parameters:

- {}:

Responses:

- 204: No Content when repositories were added to the selected list
- 401:
- 403:
- 404:
- 500:

PUT - /user/codespaces/secrets/{secret_name}/repositories/{repository_id}

Add a selected repository to a user secret

Parameters:

- {}:
- repository_id (path):

Responses:

- 204: No Content when repository was added to the selected list
- 401:
- 403:
- 404:
- 500:

DELETE - /user/codespaces/secrets/{secret_name}/repositories/{repository_id}

Remove a selected repository from a user secret

Parameters:

- {}:
- repository_id (path):

Responses:

- 204: No Content when repository was removed from the selected list
- 401:
- 403:
- 404:
- 500:

GET - /user/codespaces/{codespace_name}

Get a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

PATCH - /user/codespaces/{codespace_name}

Update a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 401:
- 403:
- 404:

DELETE - /user/codespaces/{codespace_name}

Delete a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 202:
- 304:
- 500:
- 401:
- 403:
- 404:

POST - /user/codespaces/{codespace_name}/exports

Export a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 202: Response
- 500:
- 401:
- 403:
- 404:

- 422:

GET - /user/codespaces/{codespace_name}/exports/{export_id}

Get details about a codespace export

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /user/codespaces/{codespace_name}/machines

List machine types for a codespace

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 500:
- 401:
- 403:
- 404:

POST - /user/codespaces/{codespace_name}/publish

Create a repository from an unpublished codespace

Parameters:

- {}:

Responses:

- 201: Response
- 401:
- 403:
- 404:
- 422:

POST - /user/codespaces/{codespace_name}/start

Start a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 304:
- 500:
- 400:
- 401:
- 402: Payment required
- 403:
- 404:
- 409:

POST - /user/codespaces/{codespace_name}/stop

Stop a codespace for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 500:
- 401:
- 403:
- 404:

GET - /user/docker/conflicts

Get list of conflicting packages during Docker migration for authenticated-user

Responses:

- 200: Response

PATCH - /user/email/visibility

Set primary email visibility for the authenticated user

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:
- 422:

GET - /user/emails

List email addresses for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

POST - /user/emails

Add an email address for the authenticated user

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:
- 401:

DELETE - /user/emails

Delete an email address for the authenticated user

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:
- 422:

GET - /user/followers

List followers of the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

GET - /user/following

List the people the authenticated user follows

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

GET - /user/following/{username}

Check if a person is followed by the authenticated user

Parameters:

- {}:

Responses:

- 204: if the person is followed by the authenticated user
- 404: if the person is not followed by the authenticated user
- 304:
- 403:
- 401:

PUT - /user/following/{username}

Follow a user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:
- 422:

DELETE - /user/following/{username}

Unfollow a user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/gpg_keys

List GPG keys for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

POST - /user/gpg_keys

Create a GPG key for the authenticated user

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:
- 401:

GET - /user/gpg_keys/{gpg_key_id}

Get a GPG key for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 404:
- 304:
- 403:
- 401:

DELETE - /user/gpg_keys/{gpg_key_id}

Delete a GPG key for the authenticated user

Parameters:

- {}:

Responses:

- 204: Response
- 404:
- 422:
- 304:
- 403:
- 401:

GET - /user/installations

List app installations accessible to the user access token

Parameters:

- {}:
- {}:

Responses:

- 200: You can find the permissions for the installation under the `permissions` key.
- 304:
- 403:
- 401:

GET - /user/installations/{installation_id}/repositories

List repositories accessible to the user access token

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: The access the user has to each repository is included in the hash under the `permissions` key.
- 404:
- 403:
- 304:

PUT - /user/installations/{installation_id}/repositories/{repository_id}

Add a repository to an app installation

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403:
- 304:
- 404:

DELETE - /user/installations/{installation_id}/repositories/{repository_id}

Remove a repository from an app installation

Parameters:

- {}:

- {}:

Responses:

- 204: Response
- 403:
- 304:
- 404:
- 422: Returned when the application is installed on 'all' repositories in the organization, or if this request would remove the last repository that the application has access to in the organization.

GET - /user/interaction-limits

Get interaction restrictions for your public repositories

Responses:

- 200: Default response
- 204: Response when there are no restrictions

PUT - /user/interaction-limits

Set interaction restrictions for your public repositories

Responses:

- 200: Response
- 422:

DELETE - /user/interaction-limits

Remove interaction restrictions from your public repositories

Responses:

- 204: Response

GET - /user/issues

List user account issues assigned to the authenticated user

Parameters:

- filter (query): Indicates which sorts of issues to return. 'assigned' means issues assigned to you. 'created' means issues created by you. 'mentioned' means issues mentioning you. 'subscribed' means issues you're subscribed to updates for. 'all' or 'repos' means all issues you can see, regardless of participation or creation.
- state (query): Indicates the state of the issues to return.
- {}:
- sort (query): What to sort results by.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 304:

GET - /user/keys

List public SSH keys for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

POST - /user/keys

Create a public SSH key for the authenticated user

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:
- 401:

GET - /user/keys/{key_id}

Get a public SSH key for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 404:
- 304:
- 403:
- 401:

DELETE - /user/keys/{key_id}

Delete a public SSH key for the authenticated user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/marketplace_purchases

List subscriptions for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 401:
- 404:

GET - /user/marketplace_purchases/stubbed

List subscriptions for the authenticated user (stubbed)

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 401:

GET - /user/memberships/orgs

List organization memberships for the authenticated user

Parameters:

- state (query): Indicates the state of the memberships to return. If not specified, the API returns both active and pending memberships.
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:
- 422:

GET - /user/memberships/orgs/{org}

Get an organization membership for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:

PATCH - /user/memberships/orgs/{org}

Update an organization membership for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 404:
- 422:

GET - /user/migrations

List user migrations

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

POST - /user/migrations

Start a user migration

Responses:

- 201: Response
- 422:
- 304:
- 403:
- 401:

GET - /user/migrations/{migration_id}

Get a user migration status

Parameters:

- {}:
- exclude (query):

Responses:

- 200: Response
- 404:
- 304:
- 403:
- 401:

GET - /user/migrations/{migration_id}/archive

Download a user migration archive

Parameters:

- {}:

Responses:

- 302: Response
- 304:
- 403:
- 401:

DELETE - /user/migrations/{migration_id}/archive

Delete a user migration archive

Parameters:

- {}:

Responses:

- 204: Response
- 404:
- 304:
- 403:
- 401:

DELETE - /user/migrations/{migration_id}/repos/{repo_name}/lock

Unlock a user repository

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/migrations/{migration_id}/repositories

List repositories for a user migration

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:

GET - /user/orgs

List organizations for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

GET - /user/packages

List packages for the authenticated user's namespace

Parameters:

- package_type (query): The type of supported package. Packages in GitHub's Gradle registry have the type `maven`. Docker images pushed to GitHub's Container registry (`ghcr.io`) have the type `container`. You can use the type `docker` to find images that were pushed to GitHub's Docker registry (`docker.pkg.github.com`), even if these have now been migrated to the Container registry.
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 400:

GET - /user/packages/{package_type}/{package_name}

Get a package for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response

DELETE - /user/packages/{package_type}/{package_name}

Delete a package for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /user/packages/{package_type}/{package_name}/restore

Restore a package for the authenticated user

Parameters:

- {}:
- {}:
- token (query): package token

Responses:

- 204: Response
- 404:
- 403:
- 401:

GET - /user/packages/{package_type}/{package_name}/versions

List package versions for a package owned by the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:
- state (query): The state of the package, either active or deleted.

Responses:

- 200: Response
- 404:
- 403:
- 401:

GET - /user/packages/{package_type}/{package_name}/versions/{package_version_id}

Get a package version for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /user/packages/{package_type}/{package_name}/versions/{package_version_id}

Delete a package version for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /user/packages/{package_type}/{package_name}/versions/{package_version_id}/restore

Restore a package version for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /user/projects

Create a user project

Responses:

- 201: Response
- 304:
- 403:
- 401:
- 422:

GET - /user/public_emails

List public email addresses for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/repos

List repositories for the authenticated user

Parameters:

- visibility (query): Limit results to repositories with the specified visibility.
- affiliation (query): Comma-separated list of values. Can include: * `owner`: Repositories that are owned by the authenticated user. * `collaborator`: Repositories that the user has been added to as a collaborator. * `organization_member`: Repositories that the user has access to through being a member of an organization. This includes every repository on every team that the user is on.
- type (query): Limit results to repositories of the specified type. Will cause a `422` error if used in the same request as `**visibility**` or `**affiliation**`.
- sort (query): The property to sort the results by.
- direction (query): The order to sort by. Default: `asc` when using `full_name`, otherwise `desc`.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:
- 304:
- 403:
- 401:

POST - /user/repos

Create a repository for the authenticated user

Responses:

- 201: Response
- 401:
- 304:
- 404:
- 403:
- 422:
- 400:

GET - /user/repository_invitations

List repository invitations for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

PATCH - /user/repository_invitations/{invitation_id}

Accept a repository invitation

Parameters:

- {}:

Responses:

- 204: Response
- 403:
- 409:
- 404:
- 304:

DELETE - /user/repository_invitations/{invitation_id}

Decline a repository invitation

Parameters:

- {}:

Responses:

- 204: Response
- 409:
- 304:
- 404:
- 403:

GET - /user/social_accounts

List social accounts for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

POST - /user/social_accounts

Add social accounts for the authenticated user

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:
- 401:

DELETE - /user/social_accounts

Delete social accounts for the authenticated user

Responses:

- 204: Response
- 422:
- 304:
- 404:

- 403:
- 401:

GET - /user/ssh_signing_keys

List SSH signing keys for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:
- 401:

POST - /user/ssh_signing_keys

Create a SSH signing key for the authenticated user

Responses:

- 201: Response
- 422:
- 304:
- 404:
- 403:
- 401:

GET - /user/ssh_signing_keys/{ssh_signing_key_id}

Get an SSH signing key for the authenticated user

Parameters:

- {}:

Responses:

- 200: Response
- 404:
- 304:
- 403:
- 401:

DELETE - /user/ssh_signing_keys/{ssh_signing_key_id}

Delete an SSH signing key for the authenticated user

Parameters:

- {}:

Responses:

- 204: Response
- 304:
- 404:
- 403:
- 401:

GET - /user/starred

List repositories starred by the authenticated user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

GET - /user/starred/{owner}/{repo}

Check if a repository is starred by the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response if this repository is starred by you
- 404: Not Found if this repository is not starred by you
- 401:
- 304:
- 403:

PUT - /user/starred/{owner}/{repo}

Star a repository for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 403:
- 404:
- 401:
- 304:

DELETE - /user/starred/{owner}/{repo}

Unstar a repository for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 401:
- 304:
- 403:

GET - /user/subscriptions

List repositories watched by the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 403:
- 401:

GET - /user/teams

List teams for the authenticated user

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:
- 404:
- 403:

GET - /user/{account_id}

Get a user using their ID

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /users

List users

Parameters:

- {}:
- {}:

Responses:

- 200: Response
- 304:

GET - /users/{username}

Get a user

Parameters:

- {}:

Responses:

- 200: Response
- 404:

GET - /users/{username}/attestations/{subject_digest}

List attestations

Parameters:

- {}:
- {}:
- {}:
- {}:
- subject_digest (path): Subject Digest
- predicate_type (query): Optional filter for fetching attestations with a given predicate type. This option accepts 'provenance', 'sbom', or freeform text for custom predicate types.

Responses:

- 200: Response
- 201: Response
- 204: Response
- 404:

GET - /users/{username}/docker/conflicts

Get list of conflicting packages during Docker migration for user

Parameters:

- {}:

Responses:

- 200: Response
- 403:
- 401:

GET - /users/{username}/events

List events for the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/events/orgs/{org}

List organization events for the authenticated user

Parameters:

- {}:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/events/public

List public events for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/followers

List followers of a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/following

List the people a user follows

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/following/{target_user}

Check if a user follows another user

Parameters:

- {}:
- target_user (path):

Responses:

- 204: if the user follows the target user
- 404: if the user does not follow the target user

GET - /users/{username}/gists

List gists for a user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /users/{username}/gpg_keys

List GPG keys for a user

Parameters:

- {}:

- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/hovercard

Get contextual information for a user

Parameters:

- {}:
- subject_type (query): Identifies which additional information you'd like to receive about the person's hovercard. Can be `organization`, `repository`, `issue`, `pull_request`. ****Required**** when using `subject_id`.
- subject_id (query): Uses the ID for the `subject_type` you specified. ****Required**** when using `subject_type`.

Responses:

- 200: Response
- 404:
- 422:

GET - /users/{username}/installation

Get a user installation for the authenticated app

Parameters:

- {}:

Responses:

- 200: Response

GET - /users/{username}/keys

List public keys for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/orgs

List organizations for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/packages

List packages for a user

Parameters:

- package_type (query): The type of supported package. Packages in GitHub's Gradle registry have the type `maven`. Docker images pushed to GitHub's Container registry (`ghcr.io`) have the type `container`. You can use the type `docker` to find images that were pushed to GitHub's Docker registry (`docker.pkg.github.com`), even if these have now been migrated to the Container registry.
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response
- 403:
- 401:
- 400:

GET - /users/{username}/packages/{package_type}/{package_name}

Get a package for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /users/{username}/packages/{package_type}/{package_name}

Delete a package for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /users/{username}/packages/{package_type}/{package_name}/restore

Restore a package for a user

Parameters:

- {}:
- {}:
- {}:
- token (query): package token

Responses:

- 204: Response
- 404:
- 403:
- 401:

GET - /users/{username}/packages/{package_type}/{package_name}/versions

List package versions for a package owned by a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response
- 404:
- 403:
- 401:

GET - /users/{username}/packages/{package_type}/{package_name}/versions/{package_version_id}

Get a package version for a user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

DELETE - /users/{username}/packages/{package_type}/{package_name}/versions/{package_version_id}

Delete package version for a user

Parameters:

- {}:
- {}:

- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

POST - /users/{username}/packages/{package_type}/{package_name}/versions/{package_version_id}/restore

Restore package version for a user

Parameters:

- {}:
- {}:
- {}:
- {}:

Responses:

- 204: Response
- 404:
- 403:
- 401:

GET - /users/{username}/projects

List user projects

Parameters:

- {}:
- state (query): Indicates the state of the projects to return.
- {}:
- {}:

Responses:

- 200: Response
- 422:

GET - /users/{username}/received_events

List events received by the authenticated user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/received_events/public

List public events received by a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/repos

List repositories for a user

Parameters:

- {}:
- type (query): Limit results to repositories of the specified type.
- sort (query): The property to sort the results by.
- direction (query): The order to sort by. Default: 'asc' when using 'full_name', otherwise 'desc'.
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/settings/billing/actions

Get GitHub Actions billing for a user

Parameters:

- {}:

Responses:

- 200: Response

GET - /users/{username}/settings/billing/packages

Get GitHub Packages billing for a user

Parameters:

- {}:

Responses:

- 200: Response

GET - /users/{username}/settings/billing/shared-storage

Get shared storage billing for a user

Parameters:

- {}:

Responses:

- 200: Response

GET - /users/{username}/social_accounts

List social accounts for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/ssh_signing_keys

List SSH signing keys for a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/starred

List repositories starred by a user

Parameters:

- {}:
- {}:
- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /users/{username}/subscriptions

List repositories watched by a user

Parameters:

- {}:
- {}:
- {}:

Responses:

- 200: Response

GET - /versions

Get all API versions

Responses:

- 200: Response
- 404:

GET - /zen

Get the Zen of GitHub

Responses:

- 200: Response