Mutations

In this article

createEnterpriseOrganization

createEnvironment

createIpAllowListEntry

createIssue

createLabel

createLinkedBranch

createMigrationSource

createProject

createProjectV2

createProjectV2Field

createPullRequest

createRef

createRepository

createTeamDiscussion

createTeamDiscussionComment

deleteBranchProtectionRule

deleteDeployment

deleteDiscussion

deleteDiscussionComment

deleteEnvironment

deleteIpAllowListEntry

deleteIssue

deleteIssueComment

deleteLabel

deleteLinkedBranch

deletePackageVersion

deleteProject

deleteProjectCard

deleteProjectColumn

deleteProjectV2

deleteProjectV2Field

deleteProjectV2Item

deleteProjectV2Workflow

deletePullRequestReview

deletePullRequestReviewComment

deleteRef

deleteTeamDiscussion

deleteTeamDiscussionComment

deleteVerifiableDomain

disablePullRequestAutoMerge

dismissPullRequestReview

dismissRepositoryVulnerabilityAlert

enablePullRequestAutoMerge

followOrganization

followUser

grantEnterpriseOrganizationsMigratorRole

grantMigratorRole

importProject

linkProjectV2ToRepository

linkProjectV2ToTeam

linkRepositoryToProject

lockLockable

markDiscussionCommentAsAnswer

markFileAsViewed

markProjectV2AsTemplate

markPullRequestReadyForReview

mergeBranch

mergePullRequest

minimizeComment

moveProjectCard

moveProjectColumn

pinIssue

regenerateVerifiableDomainToken

rejectDeployments

removeAssigneesFromAssignable

removeEnterpriseAdmin

removeLabelsFromLabelable

removeOutsideCollaborator

removeReaction

removeStar

removeUpvote

reopenDiscussion

reopenIssue

reopenPullRequest

requestReviews

rerequestCheckSuite

resolveReviewThread

revertPullRequest

revokeEnterpriseOrganizationsMigratorRole

revokeMigratorRole

startOrganizationMigration

startRepositoryMigration

submitPullRequestReview

transferIssue

unarchiveProjectV2Item

unarchiveRepository

unfollowOrganization

unfollowUser

unlinkProjectV2FromRepository

unlinkProjectV2FromTeam

unlinkRepositoryFromProject

unlockLockable

unmarkDiscussionCommentAsAnswer

unmarkFileAsViewed

unmarkIssueAsDuplicate

unmarkProjectV2AsTemplate

unminimizeComment

unpinIssue

unresolveReviewThread

updateBranchProtectionRule

updateCheckRun

updateCheckSuitePreferences

updateDiscussion

updateDiscussionComment

updateEnterpriseAllowPrivateRepositoryForkingSetting

updateEnterpriseDefaultRepositoryPermissionSetting

updateEnterpriseMembersCanChangeRepositoryVisibilitySetting

updateEnterpriseMembersCanCreateRepositoriesSetting

updateEnterpriseMembersCanDeleteIssuesSetting

updateEnterpriseMembersCanDeleteRepositoriesSetting

updateEnterpriseMembersCanInviteCollaboratorsSetting

updateEnterpriseMembersCanMakePurchasesSetting

updateEnterpriseMembersCanUpdateProtectedBranchesSetting

updateEnterpriseMembersCanViewDependencyInsightsSetting

updateEnterpriseOrganizationProjectsSetting

updateEnterpriseOwnerOrganizationRole

updateEnterpriseProfile

updateEnterpriseRepositoryProjectsSetting

updateEnterpriseTeamDiscussionsSetting

updateEnterpriseTwoFactorAuthenticationRequiredSetting

updateEnvironment

updateIpAllowListEnabledSetting

updateIpAllowListEntry

updateIpAllowListForInstalledAppsEnabledSetting

updateIssue

updateIssueComment

updateLabel

updateNotificationRestrictionSetting

updateOrganizationAllowPrivateRepositoryForkingSetting

updateOrganizationWebCommitSignoffSetting

updateProject

updateProjectCard

updateProjectColumn

updateProjectV2

updateProjectV2Collaborators

updateProjectV2DraftIssue

updateProjectV2ItemFieldValue

updateProjectV2ItemPosition

updatePullRequest

updatePullRequestBranch

updatePullRequestReview

updatePullRequestReviewComment

updateRef

updateRefs

updateRepository

updateRepositoryWebCommitSignoffSetting

updateSubscription

updateTeamDiscussion

updateTeamDiscussionComment

updateTeamReviewAssignment

updateTeamsRepository

updateTopics

verifyVerifiableDomain

The mutation type defines GraphQL operations that change data on the server.

About mutations ⊘

Every GraphQL schema has a root type for both queries and mutations. The [mutation type](#) defines GraphQL operations that change data on the server. It is analogous to performing HTTP verbs such as `POST`, `PATCH`, and `DELETE`.

For more information, see "[Forming calls with GraphQL](#)."

abortQueuedMigrations ⊘

Clear all of a customer's queued migrations.

Input fields for `abortQueuedMigrations`

- `input` ([AbortQueuedMigrationsInput!](#))

Return fields for `abortQueuedMigrations`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`success` ([Boolean](#))	Did the operation succeed?.

addAssigneesToAssignable ⊘

Adds assignees to an assignable object.

Input fields for `addAssigneesToAssignable`

- `input` ([AddAssigneesToAssignableInput!](#))

Return fields for `addAssigneesToAssignable`

Name	Description
`assignable` ([Assignable](#))	The item that was assigned.
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

addComment ⊘

Adds a comment to an Issue or Pull Request.

Input fields for `addComment`

- `input` (`AddCommentInput!`)

Return fields for `addComment`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`commentEdge` (`IssueCommentEdge`)	The edge from the subject's comment connection.
`subject` (`Node`)	The subject.
`timelineEdge` (`IssueTimelineItemEdge`)	The edge from the subject's timeline connection.

addDiscussionComment 🔗

Adds a comment to a Discussion, possibly as a reply to another comment.

Input fields for `addDiscussionComment`

- `input` (`AddDiscussionCommentInput!`)

Return fields for `addDiscussionComment`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`comment` (`DiscussionComment`)	The newly created discussion comment.

addDiscussionPollVote 🔗

Vote for an option in a discussion poll.

Input fields for `addDiscussionPollVote`

- `input` (`AddDiscussionPollVoteInput!`)

Return fields for `addDiscussionPollVote`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`pollOption` (`DiscussionPollOption`)	The poll option that a vote was added to.

addEnterpriseAdmin 🔗

Adds an administrator to the global enterprise account.

Input fields for `addEnterpriseAdmin`

- `input` (`AddEnterpriseAdminInput!`)

Return fields for `addEnterpriseAdmin`

Name	Description
`admin` (`User`)	The user who was added as an administrator.
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`enterprise` (`Enterprise`)	The updated enterprise.
`role` (`EnterpriseAdministratorRole`)	The role of the administrator.
`viewer` (`User`)	The viewer performing the mutation.

addEnterpriseOrganizationMember 🔗

Adds enterprise members to an organization within the enterprise.

Input fields for `addEnterpriseOrganizationMember`

- `input` (`AddEnterpriseOrganizationMemberInput!`)

Return fields for `addEnterpriseOrganizationMember`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`users` (`[User!]`)	The users who were added to the organization.

addLabelsToLabelable 🔗

Adds labels to a labelable object.

Input fields for `addLabelsToLabelable`

- `input` (`AddLabelsToLabelableInput!`)

Return fields for `addLabelsToLabelable`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.

`labelable` (<u>Labelable</u>)	The item that was labeled.

addProjectCard 🔗

Adds a card to a ProjectColumn. Either `contentId` or `note` must be provided but **not** both.

Input fields for `addProjectCard`

- `input` (<u>AddProjectCardInput!</u>)

Return fields for `addProjectCard`

Name	Description
`cardEdge` (<u>ProjectCardEdge</u>)	The edge from the ProjectColumn's card connection.
`clientMutationId` (<u>String</u>)	A unique identifier for the client performing the mutation.
`projectColumn` (<u>ProjectColumn</u>)	The ProjectColumn.

addProjectColumn 🔗

Adds a column to a Project.

Input fields for `addProjectColumn`

- `input` (<u>AddProjectColumnInput!</u>)

Return fields for `addProjectColumn`

Name	Description
`clientMutationId` (<u>String</u>)	A unique identifier for the client performing the mutation.
`columnEdge` (<u>ProjectColumnEdge</u>)	The edge from the project's column connection.
`project` (<u>Project</u>)	The project.

addProjectV2DraftIssue 🔗

Creates a new draft issue and add it to a Project.

Input fields for `addProjectV2DraftIssue`

- `input` (<u>AddProjectV2DraftIssueInput!</u>)

Return fields for `addProjectV2DraftIssue`

Name	Description

clientMutationId (`String`)	A unique identifier for the client performing the mutation.
projectItem (`ProjectV2Item`)	The draft issue added to the project.

addProjectV2ItemById 🔗

Links an existing content instance to a Project.

Input fields for `addProjectV2ItemById`

- input (`AddProjectV2ItemByIdInput!`)

Return fields for `addProjectV2ItemById`

Name	Description
clientMutationId (`String`)	A unique identifier for the client performing the mutation.
item (`ProjectV2Item`)	The item added to the project.

addPullRequestReview 🔗

Adds a review to a Pull Request.

Input fields for `addPullRequestReview`

- input (`AddPullRequestReviewInput!`)

Return fields for `addPullRequestReview`

Name	Description
clientMutationId (`String`)	A unique identifier for the client performing the mutation.
pullRequestReview (`PullRequestReview`)	The newly created pull request review.
reviewEdge (`PullRequestReviewEdge`)	The edge from the pull request's review connection.

addPullRequestReviewComment 🔗

Adds a comment to a review.

Input fields for `addPullRequestReviewComment`

- input (`AddPullRequestReviewCommentInput!`)

Return fields for `addPullRequestReviewComment`

Name	Description

clientMutationId (String)	A unique identifier for the client performing the mutation.
comment (PullRequestReviewComment)	The newly created comment.
commentEdge (PullRequestReviewCommentEdge)	The edge from the review's comment connection.

addPullRequestReviewThread 🔗

Adds a new thread to a pending Pull Request Review.

Input fields for addPullRequestReviewThread

- input (AddPullRequestReviewThreadInput!)

Return fields for addPullRequestReviewThread

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
thread (PullRequestReviewThread)	The newly created thread.

addReaction 🔗

Adds a reaction to a subject.

Input fields for addReaction

- input (AddReactionInput!)

Return fields for addReaction

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
reaction (Reaction)	The reaction object.
reactionGroups ([ReactionGroup!])	The reaction groups for the subject.
subject (Reactable)	The reactable subject.

addStar 🔗

Adds a star to a Starrable.

Input fields for addStar

- input (AddStarInput!)

Return fields for `addStar`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
starrable (Starrable)	The starrable.

addUpvote 🔗

Add an upvote to a discussion or discussion comment.

Input fields for `addUpvote`

- input (AddUpvoteInput!)

Return fields for `addUpvote`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
subject (Votable)	The votable subject.

addVerifiableDomain 🔗

Adds a verifiable domain to an owning account.

Input fields for `addVerifiableDomain`

- input (AddVerifiableDomainInput!)

Return fields for `addVerifiableDomain`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
domain (VerifiableDomain)	The verifiable domain that was added.

approveDeployments 🔗

Approve all pending deployments under one or more environments.

Input fields for `approveDeployments`

- input (ApproveDeploymentsInput!)

Return fields for `approveDeployments`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
deployments ([Deployment!])	The affected deployments.

approveVerifiableDomain 🔗

Approve a verifiable domain for notification delivery.

Input fields for approveVerifiableDomain

- input (ApproveVerifiableDomainInput!)

Return fields for approveVerifiableDomain

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
domain (VerifiableDomain)	The verifiable domain that was approved.

archiveProjectV2Item 🔗

Archives a ProjectV2Item.

Input fields for archiveProjectV2Item

- input (ArchiveProjectV2ItemInput!)

Return fields for archiveProjectV2Item

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
item (ProjectV2Item)	The item archived from the project.

archiveRepository 🔗

Marks a repository as archived.

Input fields for archiveRepository

- input (ArchiveRepositoryInput!)

Return fields for archiveRepository

Name	Description
clientMutationId (String)	A unique identifier for the client performing the

mutation.

| repository (Repository) | The repository that was marked as archived. |

changeUserStatus 🔗

Update your status on GitHub.

Input fields for `changeUserStatus`

- input (ChangeUserStatusInput!)

Return fields for `changeUserStatus`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
status (UserStatus)	Your updated status.

clearLabelsFromLabelable 🔗

Clears all labels from a labelable object.

Input fields for `clearLabelsFromLabelable`

- input (ClearLabelsFromLabelableInput!)

Return fields for `clearLabelsFromLabelable`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
labelable (Labelable)	The item that was unlabeled.

clearProjectV2ItemFieldValue 🔗

This mutation clears the value of a field for an item in a Project. Currently only text, number, date, assignees, labels, single-select, iteration and milestone fields are supported.

Input fields for `clearProjectV2ItemFieldValue`

- input (ClearProjectV2ItemFieldValueInput!)

Return fields for `clearProjectV2ItemFieldValue`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.

projectV2Item (ProjectV2Item)	The updated item.

cloneProject 🔗

Creates a new project by cloning configuration from an existing project.

Input fields for cloneProject

- input (CloneProjectInput!)

Return fields for cloneProject

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
jobStatusId (String)	The id of the JobStatus for populating cloned fields.
project (Project)	The new cloned project.

cloneTemplateRepository 🔗

Create a new repository with the same files and directory structure as a template repository.

Input fields for cloneTemplateRepository

- input (CloneTemplateRepositoryInput!)

Return fields for cloneTemplateRepository

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
repository (Repository)	The new repository.

closeDiscussion 🔗

Close a discussion.

Input fields for closeDiscussion

- input (CloseDiscussionInput!)

Return fields for closeDiscussion

Name	Description
clientMutationId (String)	A unique identifier for the client performing the

	mutation.
discussion (`Discussion`)	The discussion that was closed.

closeIssue 🔗

Close an issue.

Input fields for `closeIssue`

- input (`CloseIssueInput!`)

Return fields for `closeIssue`

Name	Description
clientMutationId (`String`)	A unique identifier for the client performing the mutation.
issue (`Issue`)	The issue that was closed.

closePullRequest 🔗

Close a pull request.

Input fields for `closePullRequest`

- input (`ClosePullRequestInput!`)

Return fields for `closePullRequest`

Name	Description
clientMutationId (`String`)	A unique identifier for the client performing the mutation.
pullRequest (`PullRequest`)	The pull request that was closed.

convertProjectCardNoteToIssue 🔗

Convert a project note card to one associated with a newly created issue.

Input fields for `convertProjectCardNoteToIssue`

- input (`ConvertProjectCardNoteToIssueInput!`)

Return fields for `convertProjectCardNoteToIssue`

Name	Description
clientMutationId (`String`)	A unique identifier for the client performing the mutation.
projectCard (`ProjectCard`)	The updated ProjectCard.

convertPullRequestToDraft 🔗

Converts a pull request to draft.

Input fields for `convertPullRequestToDraft`

- `input` (`ConvertPullRequestToDraftInput!`)

Return fields for `convertPullRequestToDraft`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`pullRequest` (`PullRequest`)	The pull request that is now a draft.

copyProjectV2 🔗

Copy a project.

Input fields for `copyProjectV2`

- `input` (`CopyProjectV2Input!`)

Return fields for `copyProjectV2`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`projectV2` (`ProjectV2`)	The copied project.

createAttributionInvitation 🔗

Invites a user to claim reattributable data.

Input fields for `createAttributionInvitation`

- `input` (`CreateAttributionInvitationInput!`)

Return fields for `createAttributionInvitation`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`owner` (`Organization`)	The owner scoping the reattributable data.
`source` (`Claimable`)	The account owning the data to reattribute.

target (Claimable)	The account which may claim the data.

createBranchProtectionRule 🔗

Create a new branch protection rule.

Input fields for `createBranchProtectionRule`

- input (CreateBranchProtectionRuleInput!)

Return fields for `createBranchProtectionRule`

Name	Description
branchProtectionRule (BranchProtectionRule)	The newly created BranchProtectionRule.
clientMutationId (String)	A unique identifier for the client performing the mutation.

createCheckRun 🔗

Create a check run.

Input fields for `createCheckRun`

- input (CreateCheckRunInput!)

Return fields for `createCheckRun`

Name	Description
checkRun (CheckRun)	The newly created check run.
clientMutationId (String)	A unique identifier for the client performing the mutation.

createCheckSuite 🔗

Create a check suite.

Input fields for `createCheckSuite`

- input (CreateCheckSuiteInput!)

Return fields for `createCheckSuite`

Name	Description
checkSuite (CheckSuite)	The newly created check suite.
clientMutationId (String)	A unique identifier for the client performing the mutation.

createCommitOnBranch 🔗

Appends a commit to the given branch as the authenticated user.

This mutation creates a commit whose parent is the HEAD of the provided branch and also updates that branch to point to the new commit. It can be thought of as similar to `git commit`.

Locating a Branch 🔗

Commits are appended to a `branch` of type `Ref`. This must refer to a git branch (i.e. the fully qualified path must begin with `refs/heads/`, although including this prefix is optional.

Callers may specify the `branch` to commit to either by its global node ID or by passing both of `repositoryNameWithOwner` and `refName`. For more details see the documentation for `CommittableBranch`.

Describing Changes 🔗

`fileChanges` are specified as a `FilesChanges` object describing `FileAdditions` and `FileDeletions`.

Please see the documentation for `FileChanges` for more information on how to use this argument to describe any set of file changes.

Authorship 🔗

Similar to the web commit interface, this mutation does not support specifying the author or committer of the commit and will not add support for this in the future.

A commit created by a successful execution of this mutation will be authored by the owner of the credential which authenticates the API request. The committer will be identical to that of commits authored using the web interface.

If you need full control over author and committer information, please use the Git Database REST API instead.

Commit Signing 🔗

Commits made using this mutation are automatically signed by GitHub if supported and will be marked as verified in the user interface.

Input fields for `createCommitOnBranch`

- `input` (`CreateCommitOnBranchInput!`)

Return fields for `createCommitOnBranch`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`commit` (`Commit`)	The new commit.
`ref` (`Ref`)	The ref which has been updated to point to the new commit.

createDeployment 🔗

Creates a new deployment event.

> **Preview notice**
>
> `createDeployment` is available under the [Deployments preview](). During the preview period, the API may change without notice.

Input fields for `createDeployment`

- `input` ([CreateDeploymentInput!]())

Return fields for `createDeployment`

Name	Description
`autoMerged` ([Boolean]())	True if the default branch has been auto-merged into the deployment ref.
`clientMutationId` ([String]())	A unique identifier for the client performing the mutation.
`deployment` ([Deployment]())	The new deployment.

createDeploymentStatus 🔗

Create a deployment status.

> **Preview notice**
>
> `createDeploymentStatus` is available under the [Deployments preview](). During the preview period, the API may change without notice.

Input fields for `createDeploymentStatus`

- `input` ([CreateDeploymentStatusInput!]())

Return fields for `createDeploymentStatus`

Name	Description
`clientMutationId` ([String]())	A unique identifier for the client performing the mutation.
`deploymentStatus` ([DeploymentStatus]())	The new deployment status.

createDiscussion 🔗

Create a discussion.

Input fields for `createDiscussion`

- `input` ([CreateDiscussionInput!]())

Return fields for `createDiscussion`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
discussion ([Discussion](#))	The discussion that was just created.

createEnterpriseOrganization 🔗

Creates an organization as part of an enterprise account.

Input fields for `createEnterpriseOrganization`

- input ([CreateEnterpriseOrganizationInput!](#))

Return fields for `createEnterpriseOrganization`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
enterprise ([Enterprise](#))	The enterprise that owns the created organization.
organization ([Organization](#))	The organization that was created.

createEnvironment 🔗

Creates an environment or simply returns it if already exists.

Input fields for `createEnvironment`

- input ([CreateEnvironmentInput!](#))

Return fields for `createEnvironment`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
environment ([Environment](#))	The new or existing environment.

createIpAllowListEntry 🔗

Creates a new IP allow list entry.

Input fields for `createIpAllowListEntry`

- input ([CreateIpAllowListEntryInput!](#))

Return fields for `createIpAllowListEntry`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
ipAllowListEntry (IpAllowListEntry)	The IP allow list entry that was created.

createIssue 🔗

Creates a new issue.

Input fields for `createIssue`

- input (CreateIssueInput!)

Return fields for `createIssue`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
issue (Issue)	The new issue.

createLabel 🔗

Creates a new label.

> **Preview notice**
>
> `createLabel` is available under the Labels preview. During the preview period, the API may change without notice.

Input fields for `createLabel`

- input (CreateLabelInput!)

Return fields for `createLabel`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
label (Label)	The new label.

createLinkedBranch 🔗

Create a branch linked to an issue.

Input fields for `createLinkedBranch`

- `input` ([CreateLinkedBranchInput!](#))

Return fields for `createLinkedBranch`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`linkedBranch` ([LinkedBranch](#))	The new branch issue reference.

createMigrationSource 🔗

Creates a GitHub Enterprise Importer (GEI) migration source.

Input fields for `createMigrationSource`

- `input` ([CreateMigrationSourceInput!](#))

Return fields for `createMigrationSource`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`migrationSource` ([MigrationSource](#))	The created migration source.

createProject 🔗

Creates a new project.

Input fields for `createProject`

- `input` ([CreateProjectInput!](#))

Return fields for `createProject`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`project` ([Project](#))	The new project.

createProjectV2 🔗

Creates a new project.

Input fields for `createProjectV2`

- input ([CreateProjectV2Input!](#))

Return fields for `createProjectV2`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
projectV2 ([ProjectV2](#))	The new project.

createProjectV2Field 🔗

Create a new project field.

Input fields for `createProjectV2Field`

- input ([CreateProjectV2FieldInput!](#))

Return fields for `createProjectV2Field`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
projectV2Field ([ProjectV2FieldConfiguration](#))	The new field.

createPullRequest 🔗

Create a new pull request.

Input fields for `createPullRequest`

- input ([CreatePullRequestInput!](#))

Return fields for `createPullRequest`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
pullRequest ([PullRequest](#))	The new pull request.

createRef 🔗

Create a new Git Ref.

Input fields for `createRef`

- input ([CreateRefInput!](#))

Return fields for `createRef`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
ref (Ref)	The newly created ref.

createRepository 🔗

Create a new repository.

Input fields for `createRepository`

- input (CreateRepositoryInput!)

Return fields for `createRepository`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
repository (Repository)	The new repository.

createTeamDiscussion 🔗

Creates a new team discussion.

Input fields for `createTeamDiscussion`

- input (CreateTeamDiscussionInput!)

Return fields for `createTeamDiscussion`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
teamDiscussion (TeamDiscussion)	The new discussion.

createTeamDiscussionComment 🔗

Creates a new team discussion comment.

Input fields for `createTeamDiscussionComment`

- input (CreateTeamDiscussionCommentInput!)

Return fields for `createTeamDiscussionComment`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
teamDiscussionComment (TeamDiscussionComment)	The new comment.

deleteBranchProtectionRule 🔗

Delete a branch protection rule.

Input fields for deleteBranchProtectionRule

- input (DeleteBranchProtectionRuleInput!)

Return fields for deleteBranchProtectionRule

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.

deleteDeployment 🔗

Deletes a deployment.

Input fields for deleteDeployment

- input (DeleteDeploymentInput!)

Return fields for deleteDeployment

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.

deleteDiscussion 🔗

Delete a discussion and all of its replies.

Input fields for deleteDiscussion

- input (DeleteDiscussionInput!)

Return fields for deleteDiscussion

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
discussion (Discussion)	The discussion that was just deleted.

deleteDiscussionComment 🔗

Delete a discussion comment. If it has replies, wipe it instead.

Input fields for `deleteDiscussionComment`

- `input` ([DeleteDiscussionCommentInput!](#))

Return fields for `deleteDiscussionComment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`comment` ([DiscussionComment](#))	The discussion comment that was just deleted.

deleteEnvironment 🔗

Deletes an environment.

Input fields for `deleteEnvironment`

- `input` ([DeleteEnvironmentInput!](#))

Return fields for `deleteEnvironment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

deleteIpAllowListEntry 🔗

Deletes an IP allow list entry.

Input fields for `deleteIpAllowListEntry`

- `input` ([DeleteIpAllowListEntryInput!](#))

Return fields for `deleteIpAllowListEntry`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`ipAllowListEntry` ([IpAllowListEntry](#))	The IP allow list entry that was deleted.

deleteIssue 🔗

Deletes an Issue object.

Input fields for `deleteIssue`

- `input` ([DeleteIssueInput!](#))

Return fields for `deleteIssue`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`repository` ([Repository](#))	The repository the issue belonged to.

deleteIssueComment 🔗

Deletes an IssueComment object.

Input fields for `deleteIssueComment`

- `input` ([DeleteIssueCommentInput!](#))

Return fields for `deleteIssueComment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

deleteLabel 🔗

Deletes a label.

> **Preview notice**
>
> `deleteLabel` is available under the [Labels preview](#). During the preview period, the API may change without notice.

Input fields for `deleteLabel`

- `input` ([DeleteLabelInput!](#))

Return fields for `deleteLabel`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

deleteLinkedBranch 🔗

Unlink a branch from an issue.

Input fields for `deleteLinkedBranch`

- `input` (`DeleteLinkedBranchInput!`)

Return fields for `deleteLinkedBranch`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`issue` (`Issue`)	The issue the linked branch was unlinked from.

deletePackageVersion 🔗

Delete a package version.

> **Preview notice**
>
> `deletePackageVersion` is available under the [Access to package version deletion preview](). During the preview period, the API may change without notice.

Input fields for `deletePackageVersion`

- `input` (`DeletePackageVersionInput!`)

Return fields for `deletePackageVersion`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`success` (`Boolean`)	Whether or not the operation succeeded.

deleteProject 🔗

Deletes a project.

Input fields for `deleteProject`

- `input` (`DeleteProjectInput!`)

Return fields for `deleteProject`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`owner` (`ProjectOwner`)	The repository or organization the project was removed from.

deleteProjectCard 🔗

Deletes a project card.

Input fields for `deleteProjectCard`

- `input` ([DeleteProjectCardInput!](#))

Return fields for `deleteProjectCard`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`column` ([ProjectColumn](#))	The column the deleted card was in.
`deletedCardId` ([ID](#))	The deleted card ID.

deleteProjectColumn 🔗

Deletes a project column.

Input fields for `deleteProjectColumn`

- `input` ([DeleteProjectColumnInput!](#))

Return fields for `deleteProjectColumn`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`deletedColumnId` ([ID](#))	The deleted column ID.
`project` ([Project](#))	The project the deleted column was in.

deleteProjectV2 🔗

Delete a project.

Input fields for `deleteProjectV2`

- `input` ([DeleteProjectV2Input!](#))

Return fields for `deleteProjectV2`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`projectV2` ([ProjectV2](#))	The deleted Project.

deleteProjectV2Field 🔗

Delete a project field.

Input fields for `deleteProjectV2Field`

- `input` ([DeleteProjectV2FieldInput!](#))

Return fields for `deleteProjectV2Field`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`projectV2Field` ([ProjectV2FieldConfiguration](#))	The deleted field.

deleteProjectV2Item 🔗

Deletes an item from a Project.

Input fields for `deleteProjectV2Item`

- `input` ([DeleteProjectV2ItemInput!](#))

Return fields for `deleteProjectV2Item`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`deletedItemId` ([ID](#))	The ID of the deleted item.

deleteProjectV2Workflow 🔗

Deletes a project workflow.

Input fields for `deleteProjectV2Workflow`

- `input` ([DeleteProjectV2WorkflowInput!](#))

Return fields for `deleteProjectV2Workflow`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`deletedWorkflowId` ([ID](#))	The ID of the deleted workflow.
`projectV2` ([ProjectV2](#))	The project the deleted workflow was in.

deletePullRequestReview 🔗

Deletes a pull request review.

Input fields for `deletePullRequestReview`

- `input` ([DeletePullRequestReviewInput!](#))

Return fields for `deletePullRequestReview`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequestReview` ([PullRequestReview](#))	The deleted pull request review.

deletePullRequestReviewComment 🔗

Deletes a pull request review comment.

Input fields for `deletePullRequestReviewComment`

- `input` ([DeletePullRequestReviewCommentInput!](#))

Return fields for `deletePullRequestReviewComment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequestReview` ([PullRequestReview](#))	The pull request review the deleted comment belonged to.
`pullRequestReviewComment` ([PullRequestReviewComment](#))	The deleted pull request review comment.

deleteRef 🔗

Delete a Git Ref.

Input fields for `deleteRef`

- `input` ([DeleteRefInput!](#))

Return fields for `deleteRef`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

deleteTeamDiscussion 🔗

Deletes a team discussion.

Input fields for `deleteTeamDiscussion`

- `input` ([DeleteTeamDiscussionInput!](#))

Return fields for `deleteTeamDiscussion`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.

deleteTeamDiscussionComment 🔗

Deletes a team discussion comment.

Input fields for `deleteTeamDiscussionComment`

- `input` ([DeleteTeamDiscussionCommentInput!](#))

Return fields for `deleteTeamDiscussionComment`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.

deleteVerifiableDomain 🔗

Deletes a verifiable domain.

Input fields for `deleteVerifiableDomain`

- `input` ([DeleteVerifiableDomainInput!](#))

Return fields for `deleteVerifiableDomain`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
owner ([VerifiableDomainOwner](#))	The owning account from which the domain was deleted.

disablePullRequestAutoMerge 🔗

Disable auto merge on the given pull request.

Input fields for `disablePullRequestAutoMerge`

- input ([DisablePullRequestAutoMergeInput!](#))

Return fields for `disablePullRequestAutoMerge`

Name	Description
actor ([Actor](#))	Identifies the actor who performed the event.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
pullRequest ([PullRequest](#))	The pull request auto merge was disabled on.

dismissPullRequestReview 🔗

Dismisses an approved or rejected pull request review.

Input fields for `dismissPullRequestReview`

- input ([DismissPullRequestReviewInput!](#))

Return fields for `dismissPullRequestReview`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
pullRequestReview ([PullRequestReview](#))	The dismissed pull request review.

dismissRepositoryVulnerabilityAlert 🔗

Dismisses the Dependabot alert.

Input fields for `dismissRepositoryVulnerabilityAlert`

- input ([DismissRepositoryVulnerabilityAlertInput!](#))

Return fields for `dismissRepositoryVulnerabilityAlert`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
repositoryVulnerabilityAlert ([RepositoryVulnerabilityAlert](#))	The Dependabot alert that was dismissed.

enablePullRequestAutoMerge 🔗

Enable the default auto-merge on a pull request.

Input fields for `enablePullRequestAutoMerge`

- `input` ([EnablePullRequestAutoMergeInput!](#))

Return fields for `enablePullRequestAutoMerge`

Name	Description
`actor` ([Actor](#))	Identifies the actor who performed the event.
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The pull request auto-merge was enabled on.

followOrganization 🔗

Follow an organization.

Input fields for `followOrganization`

- `input` ([FollowOrganizationInput!](#))

Return fields for `followOrganization`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`organization` ([Organization](#))	The organization that was followed.

followUser 🔗

Follow a user.

Input fields for `followUser`

- `input` ([FollowUserInput!](#))

Return fields for `followUser`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`user` ([User](#))	The user that was followed.

grantEnterpriseOrganizationsMigratorRole 🔗

Grant the migrator role to a user for all organizations under an enterprise account.

Input fields for `grantEnterpriseOrganizationsMigratorRole`

- `input` (GrantEnterpriseOrganizationsMigratorRoleInput!)

Return fields for `grantEnterpriseOrganizationsMigratorRole`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`organizations` (OrganizationConnection)	The organizations that had the migrator role applied to for the given user.

grantMigratorRole ∅

Grant the migrator role to a user or a team.

Input fields for `grantMigratorRole`

- `input` (GrantMigratorRoleInput!)

Return fields for `grantMigratorRole`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`success` (Boolean)	Did the operation succeed?.

importProject ∅

Creates a new project by importing columns and a list of issues/PRs.

> **Preview notice**
>
> `importProject` is available under the Import project preview. During the preview period, the API may change without notice.

Input fields for `importProject`

- `input` (ImportProjectInput!)

Return fields for `importProject`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`project` (Project)	The new Project!.

linkProjectV2ToRepository 🔗

Links a project to a repository.

Input fields for `linkProjectV2ToRepository`

- `input` ([LinkProjectV2ToRepositoryInput!](#))

Return fields for `linkProjectV2ToRepository`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`repository` ([Repository](#))	The repository the project is linked to.

linkProjectV2ToTeam 🔗

Links a project to a team.

Input fields for `linkProjectV2ToTeam`

- `input` ([LinkProjectV2ToTeamInput!](#))

Return fields for `linkProjectV2ToTeam`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`team` ([Team](#))	The team the project is linked to.

linkRepositoryToProject 🔗

Creates a repository link for a project.

Input fields for `linkRepositoryToProject`

- `input` ([LinkRepositoryToProjectInput!](#))

Return fields for `linkRepositoryToProject`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`project` ([Project](#))	The linked Project.
`repository` ([Repository](#))	The linked Repository.

lockLockable 🔗

Lock a lockable object.

Input fields for `lockLockable`

- `input` (LockLockableInput!)

Return fields for `lockLockable`

Name	Description
`actor` (Actor)	Identifies the actor who performed the event.
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`lockedRecord` (Lockable)	The item that was locked.

markDiscussionCommentAsAnswer 🔗

Mark a discussion comment as the chosen answer for discussions in an answerable category.

Input fields for `markDiscussionCommentAsAnswer`

- `input` (MarkDiscussionCommentAsAnswerInput!)

Return fields for `markDiscussionCommentAsAnswer`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`discussion` (Discussion)	The discussion that includes the chosen comment.

markFileAsViewed 🔗

Mark a pull request file as viewed.

Input fields for `markFileAsViewed`

- `input` (MarkFileAsViewedInput!)

Return fields for `markFileAsViewed`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`pullRequest` (PullRequest)	The updated pull request.

markProjectV2AsTemplate 🔗

Mark a project as a template. Note that only projects which are owned by an Organization can be marked as a template.

Input fields for `markProjectV2AsTemplate`

- `input` ([MarkProjectV2AsTemplateInput!](#))

Return fields for `markProjectV2AsTemplate`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
projectV2 ([ProjectV2](#))	The project.

markPullRequestReadyForReview 🔗

Marks a pull request ready for review.

Input fields for `markPullRequestReadyForReview`

- `input` ([MarkPullRequestReadyForReviewInput!](#))

Return fields for `markPullRequestReadyForReview`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
pullRequest ([PullRequest](#))	The pull request that is ready for review.

mergeBranch 🔗

Merge a head into a branch.

Input fields for `mergeBranch`

- `input` ([MergeBranchInput!](#))

Return fields for `mergeBranch`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
mergeCommit ([Commit](#))	The resulting merge Commit.

mergePullRequest 🔗

Merge a pull request.

Input fields for `mergePullRequest`

- input ([MergePullRequestInput!](#))

Return fields for `mergePullRequest`

Name	Description
actor ([Actor](#))	Identifies the actor who performed the event.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
pullRequest ([PullRequest](#))	The pull request that was merged.

minimizeComment 🔗

Minimizes a comment on an Issue, Commit, Pull Request, or Gist.

Input fields for `minimizeComment`

- input ([MinimizeCommentInput!](#))

Return fields for `minimizeComment`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
minimizedComment ([Minimizable](#))	The comment that was minimized.

moveProjectCard 🔗

Moves a project card to another place.

Input fields for `moveProjectCard`

- input ([MoveProjectCardInput!](#))

Return fields for `moveProjectCard`

Name	Description
cardEdge ([ProjectCardEdge](#))	The new edge of the moved card.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.

moveProjectColumn 🔗

Moves a project column to another place.

Input fields for `moveProjectColumn`

- `input` ([MoveProjectColumnInput!](#))

Return fields for `moveProjectColumn`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
columnEdge ([ProjectColumnEdge](#))	The new edge of the moved column.

pinIssue 🔗

Pin an issue to a repository.

Input fields for `pinIssue`

- `input` ([PinIssueInput!](#))

Return fields for `pinIssue`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
issue ([Issue](#))	The issue that was pinned.

regenerateVerifiableDomainToken 🔗

Regenerates a verifiable domain's verification token.

Input fields for `regenerateVerifiableDomainToken`

- `input` ([RegenerateVerifiableDomainTokenInput!](#))

Return fields for `regenerateVerifiableDomainToken`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
verificationToken ([String](#))	The verification token that was generated.

rejectDeployments 🔗

Reject all pending deployments under one or more environments.

Input fields for `rejectDeployments`

- `input` (`RejectDeploymentsInput!`)

Return fields for `rejectDeployments`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`deployments` (`[Deployment!]`)	The affected deployments.

removeAssigneesFromAssignable 🔗

Removes assignees from an assignable object.

Input fields for `removeAssigneesFromAssignable`

- `input` (`RemoveAssigneesFromAssignableInput!`)

Return fields for `removeAssigneesFromAssignable`

Name	Description
`assignable` (`Assignable`)	The item that was unassigned.
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.

removeEnterpriseAdmin 🔗

Removes an administrator from the enterprise.

Input fields for `removeEnterpriseAdmin`

- `input` (`RemoveEnterpriseAdminInput!`)

Return fields for `removeEnterpriseAdmin`

Name	Description
`admin` (`User`)	The user who was removed as an administrator.
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`enterprise` (`Enterprise`)	The updated enterprise.
`message` (`String`)	A message confirming the result of removing an administrator.
`viewer` (`User`)	The viewer performing the mutation.

removeLabelsFromLabelable 🔗

Removes labels from a Labelable object.

Input fields for `removeLabelsFromLabelable`

- `input` ([RemoveLabelsFromLabelableInput!](#))

Return fields for `removeLabelsFromLabelable`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`labelable` ([Labelable](#))	The Labelable the labels were removed from.

removeOutsideCollaborator 🔗

Removes outside collaborator from all repositories in an organization.

Input fields for `removeOutsideCollaborator`

- `input` ([RemoveOutsideCollaboratorInput!](#))

Return fields for `removeOutsideCollaborator`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`removedUser` ([User](#))	The user that was removed as an outside collaborator.

removeReaction 🔗

Removes a reaction from a subject.

Input fields for `removeReaction`

- `input` ([RemoveReactionInput!](#))

Return fields for `removeReaction`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`reaction` ([Reaction](#))	The reaction object.
`reactionGroups` ([[ReactionGroup!]](#))	The reaction groups for the subject.

subject ([Reactable](#))	The reactable subject.

removeStar 🔗

Removes a star from a Starrable.

Input fields for `removeStar`

- input ([RemoveStarInput!](#))

Return fields for `removeStar`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
starrable ([Starrable](#))	The starrable.

removeUpvote 🔗

Remove an upvote to a discussion or discussion comment.

Input fields for `removeUpvote`

- input ([RemoveUpvoteInput!](#))

Return fields for `removeUpvote`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
subject ([Votable](#))	The votable subject.

reopenDiscussion 🔗

Reopen a discussion.

Input fields for `reopenDiscussion`

- input ([ReopenDiscussionInput!](#))

Return fields for `reopenDiscussion`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
discussion ([Discussion](#))	The discussion that was reopened.

reopenIssue 🔗

Reopen a issue.

Input fields for `reopenIssue`

- `input` ([ReopenIssueInput!](#))

Return fields for `reopenIssue`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`issue` ([Issue](#))	The issue that was opened.

reopenPullRequest 🔗

Reopen a pull request.

Input fields for `reopenPullRequest`

- `input` ([ReopenPullRequestInput!](#))

Return fields for `reopenPullRequest`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The pull request that was reopened.

requestReviews 🔗

Set review requests on a pull request.

Input fields for `requestReviews`

- `input` ([RequestReviewsInput!](#))

Return fields for `requestReviews`

Name	Description
`actor` ([Actor](#))	Identifies the actor who performed the event.
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The pull request that is getting requests.
`requestedReviewersEdge` ([UserEdge](#))	The edge from the pull request to the requested reviewers.

rerequestCheckSuite 🔗

Rerequests an existing check suite.

Input fields for `rerequestCheckSuite`

- `input` ([RerequestCheckSuiteInput!](#))

Return fields for `rerequestCheckSuite`

Name	Description
`checkSuite` ([CheckSuite](#))	The requested check suite.
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

resolveReviewThread 🔗

Marks a review thread as resolved.

Input fields for `resolveReviewThread`

- `input` ([ResolveReviewThreadInput!](#))

Return fields for `resolveReviewThread`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`thread` ([PullRequestReviewThread](#))	The thread to resolve.

revertPullRequest 🔗

Create a pull request that reverts the changes from a merged pull request.

Input fields for `revertPullRequest`

- `input` ([RevertPullRequestInput!](#))

Return fields for `revertPullRequest`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The pull request that was reverted.
`revertPullRequest` ([PullRequest](#))	The new pull request that reverts the input pull request.

revokeEnterpriseOrganizationsMigratorRole 🔗

Revoke the migrator role to a user for all organizations under an enterprise account.

Input fields for `revokeEnterpriseOrganizationsMigratorRole`

- `input` ([RevokeEnterpriseOrganizationsMigratorRoleInput!](#))

Return fields for `revokeEnterpriseOrganizationsMigratorRole`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`organizations` ([OrganizationConnection](#))	The organizations that had the migrator role revoked for the given user.

revokeMigratorRole 🔗

Revoke the migrator role from a user or a team.

Input fields for `revokeMigratorRole`

- `input` ([RevokeMigratorRoleInput!](#))

Return fields for `revokeMigratorRole`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`success` ([Boolean](#))	Did the operation succeed?.

startOrganizationMigration 🔗

Starts a GitHub Enterprise Importer organization migration.

Input fields for `startOrganizationMigration`

- `input` ([StartOrganizationMigrationInput!](#))

Return fields for `startOrganizationMigration`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`orgMigration` ([OrganizationMigration](#))	The new organization migration.

startRepositoryMigration 🔗

Starts a GitHub Enterprise Importer (GEI) repository migration.

Input fields for `startRepositoryMigration`

- `input` ([StartRepositoryMigrationInput!](#))

Return fields for `startRepositoryMigration`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`repositoryMigration` ([RepositoryMigration](#))	The new repository migration.

submitPullRequestReview 🔗

Submits a pending pull request review.

Input fields for `submitPullRequestReview`

- `input` ([SubmitPullRequestReviewInput!](#))

Return fields for `submitPullRequestReview`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequestReview` ([PullRequestReview](#))	The submitted pull request review.

transferIssue 🔗

Transfer an issue to a different repository.

Input fields for `transferIssue`

- `input` ([TransferIssueInput!](#))

Return fields for `transferIssue`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`issue` ([Issue](#))	The issue that was transferred.

unarchiveProjectV2Item 🔗

Unarchives a ProjectV2Item.

Input fields for `unarchiveProjectV2Item`

- `input` ([UnarchiveProjectV2ItemInput!](#))

Return fields for `unarchiveProjectV2Item`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`item` ([ProjectV2Item](#))	The item unarchived from the project.

unarchiveRepository 🔗

Unarchives a repository.

Input fields for `unarchiveRepository`

- `input` ([UnarchiveRepositoryInput!](#))

Return fields for `unarchiveRepository`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`repository` ([Repository](#))	The repository that was unarchived.

unfollowOrganization 🔗

Unfollow an organization.

Input fields for `unfollowOrganization`

- `input` ([UnfollowOrganizationInput!](#))

Return fields for `unfollowOrganization`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`organization` ([Organization](#))	The organization that was unfollowed.

unfollowUser 🔗

Unfollow a user.

Input fields for `unfollowUser`

- `input` (UnfollowUserInput!)

Return fields for `unfollowUser`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`user` (User)	The user that was unfollowed.

unlinkProjectV2FromRepository 🔗

Unlinks a project from a repository.

Input fields for `unlinkProjectV2FromRepository`

- `input` (UnlinkProjectV2FromRepositoryInput!)

Return fields for `unlinkProjectV2FromRepository`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`repository` (Repository)	The repository the project is no longer linked to.

unlinkProjectV2FromTeam 🔗

Unlinks a project to a team.

Input fields for `unlinkProjectV2FromTeam`

- `input` (UnlinkProjectV2FromTeamInput!)

Return fields for `unlinkProjectV2FromTeam`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`team` (Team)	The team the project is unlinked from.

unlinkRepositoryFromProject 🔗

Deletes a repository link from a project.

Input fields for `unlinkRepositoryFromProject`

- input ([UnlinkRepositoryFromProjectInput!](#))

Return fields for `unlinkRepositoryFromProject`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
project ([Project](#))	The linked Project.
repository ([Repository](#))	The linked Repository.

unlockLockable 🔗

Unlock a lockable object.

Input fields for `unlockLockable`

- input ([UnlockLockableInput!](#))

Return fields for `unlockLockable`

Name	Description
actor ([Actor](#))	Identifies the actor who performed the event.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
unlockedRecord ([Lockable](#))	The item that was unlocked.

unmarkDiscussionCommentAsAnswer 🔗

Unmark a discussion comment as the chosen answer for discussions in an answerable category.

Input fields for `unmarkDiscussionCommentAsAnswer`

- input ([UnmarkDiscussionCommentAsAnswerInput!](#))

Return fields for `unmarkDiscussionCommentAsAnswer`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
discussion ([Discussion](#))	The discussion that includes the comment.

unmarkFileAsViewed 🔗

Unmark a pull request file as viewed.

Input fields for `unmarkFileAsViewed`

- `input` ([UnmarkFileAsViewedInput!](#))

Return fields for `unmarkFileAsViewed`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The updated pull request.

unmarkIssueAsDuplicate 🔗

Unmark an issue as a duplicate of another issue.

Input fields for `unmarkIssueAsDuplicate`

- `input` ([UnmarkIssueAsDuplicateInput!](#))

Return fields for `unmarkIssueAsDuplicate`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`duplicate` ([IssueOrPullRequest](#))	The issue or pull request that was marked as a duplicate.

unmarkProjectV2AsTemplate 🔗

Unmark a project as a template.

Input fields for `unmarkProjectV2AsTemplate`

- `input` ([UnmarkProjectV2AsTemplateInput!](#))

Return fields for `unmarkProjectV2AsTemplate`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`projectV2` ([ProjectV2](#))	The project.

unminimizeComment 🔗

Unminimizes a comment on an Issue, Commit, Pull Request, or Gist.

Input fields for `unminimizeComment`

- `input` (UnminimizeCommentInput!)

Return fields for `unminimizeComment`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`unminimizedComment` (Minimizable)	The comment that was unminimized.

unpinIssue 🔗

Unpin a pinned issue from a repository.

Input fields for `unpinIssue`

- `input` (UnpinIssueInput!)

Return fields for `unpinIssue`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`issue` (Issue)	The issue that was unpinned.

unresolveReviewThread 🔗

Marks a review thread as unresolved.

Input fields for `unresolveReviewThread`

- `input` (UnresolveReviewThreadInput!)

Return fields for `unresolveReviewThread`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`thread` (PullRequestReviewThread)	The thread to resolve.

updateBranchProtectionRule 🔗

Update a branch protection rule.

Input fields for `updateBranchProtectionRule`

- input ([UpdateBranchProtectionRuleInput!](#))

Return fields for `updateBranchProtectionRule`

Name	Description
branchProtectionRule ([BranchProtectionRule](#))	The newly created BranchProtectionRule.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.

updateCheckRun 🔗

Update a check run.

Input fields for `updateCheckRun`

- input ([UpdateCheckRunInput!](#))

Return fields for `updateCheckRun`

Name	Description
checkRun ([CheckRun](#))	The updated check run.
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.

updateCheckSuitePreferences 🔗

Modifies the settings of an existing check suite.

Input fields for `updateCheckSuitePreferences`

- input ([UpdateCheckSuitePreferencesInput!](#))

Return fields for `updateCheckSuitePreferences`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
repository ([Repository](#))	The updated repository.

updateDiscussion 🔗

Update a discussion.

Input fields for `updateDiscussion`

- input ([UpdateDiscussionInput!](#))

Return fields for `updateDiscussion`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
discussion (Discussion)	The modified discussion.

updateDiscussionComment 🔗

Update the contents of a comment on a Discussion.

Input fields for `updateDiscussionComment`

- input (UpdateDiscussionCommentInput!)

Return fields for `updateDiscussionComment`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
comment (DiscussionComment)	The modified discussion comment.

updateEnterpriseAllowPrivateRepositoryForkingSetting 🔗

Sets whether private repository forks are enabled for an enterprise.

Input fields for `updateEnterpriseAllowPrivateRepositoryForkingSetting`

- input (UpdateEnterpriseAllowPrivateRepositoryForkingSettingInput!)

Return fields for `updateEnterpriseAllowPrivateRepositoryForkingSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated allow private repository forking setting.
message (String)	A message confirming the result of updating the allow private repository forking setting.

updateEnterpriseDefaultRepositoryPermissionSetting 🔗

Sets the base repository permission for organizations in an enterprise.

Input fields for `updateEnterpriseDefaultRepositoryPermissionSetting`

- input (UpdateEnterpriseDefaultRepositoryPermissionSettingInput!)

Return fields for `updateEnterpriseDefaultRepositoryPermissionSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated base repository permission setting.
message (String)	A message confirming the result of updating the base repository permission setting.

updateEnterpriseMembersCanChangeRepositoryVisibilitySetting 🔗

Sets whether organization members with admin permissions on a repository can change repository visibility.

Input fields for `updateEnterpriseMembersCanChangeRepositoryVisibilitySetting`

- input (UpdateEnterpriseMembersCanChangeRepositoryVisibilitySettingInput!)

Return fields for `updateEnterpriseMembersCanChangeRepositoryVisibilitySetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated members can change repository visibility setting.
message (String)	A message confirming the result of updating the members can change repository visibility setting.

updateEnterpriseMembersCanCreateRepositoriesSetting 🔗

Sets the members can create repositories setting for an enterprise.

Input fields for `updateEnterpriseMembersCanCreateRepositoriesSetting`

- input (UpdateEnterpriseMembersCanCreateRepositoriesSettingInput!)

Return fields for `updateEnterpriseMembersCanCreateRepositoriesSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the

	mutation.
enterprise (Enterprise)	The enterprise with the updated members can create repositories setting.
message (String)	A message confirming the result of updating the members can create repositories setting.

updateEnterpriseMembersCanDeleteIssuesSetting 🔗

Sets the members can delete issues setting for an enterprise.

Input fields for `updateEnterpriseMembersCanDeleteIssuesSetting`

- input (UpdateEnterpriseMembersCanDeleteIssuesSettingInput!)

Return fields for `updateEnterpriseMembersCanDeleteIssuesSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated members can delete issues setting.
message (String)	A message confirming the result of updating the members can delete issues setting.

updateEnterpriseMembersCanDeleteRepositoriesSetting 🔗

Sets the members can delete repositories setting for an enterprise.

Input fields for `updateEnterpriseMembersCanDeleteRepositoriesSetting`

- input (UpdateEnterpriseMembersCanDeleteRepositoriesSettingInput!)

Return fields for `updateEnterpriseMembersCanDeleteRepositoriesSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated members can delete repositories setting.
message (String)	A message confirming the result of updating the members can delete repositories setting.

updateEnterpriseMembersCanInviteCollaboratorsSetting 🔗

Sets whether members can invite collaborators are enabled for an enterprise.

Input fields for `updateEnterpriseMembersCanInviteCollaboratorsSetting`

- `input` (`UpdateEnterpriseMembersCanInviteCollaboratorsSettingInput!`)

Return fields for `updateEnterpriseMembersCanInviteCollaboratorsSetting`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`enterprise` (`Enterprise`)	The enterprise with the updated members can invite collaborators setting.
`message` (`String`)	A message confirming the result of updating the members can invite collaborators setting.

updateEnterpriseMembersCanMakePurchasesSetting 🔗

Sets whether or not an organization admin can make purchases.

Input fields for `updateEnterpriseMembersCanMakePurchasesSetting`

- `input` (`UpdateEnterpriseMembersCanMakePurchasesSettingInput!`)

Return fields for `updateEnterpriseMembersCanMakePurchasesSetting`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`enterprise` (`Enterprise`)	The enterprise with the updated members can make purchases setting.
`message` (`String`)	A message confirming the result of updating the members can make purchases setting.

updateEnterpriseMembersCanUpdateProtectedBranchesSetting 🔗

Sets the members can update protected branches setting for an enterprise.

Input fields for `updateEnterpriseMembersCanUpdateProtectedBranchesSetting`

- `input` (`UpdateEnterpriseMembersCanUpdateProtectedBranchesSettingInput!`)

Return fields for `updateEnterpriseMembersCanUpdateProtectedBranchesSetting`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.

	mutation.
enterprise (Enterprise)	The enterprise with the updated members can update protected branches setting.
message (String)	A message confirming the result of updating the members can update protected branches setting.

updateEnterpriseMembersCanViewDependencyInsighthtsSetting ⧉

Sets the members can view dependency insights for an enterprise.

Input fields for `updateEnterpriseMembersCanViewDependencyInsightsSetting`

- input (UpdateEnterpriseMembersCanViewDependencyInsightsSettingInput!)

Return fields for `updateEnterpriseMembersCanViewDependencyInsightsSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated members can view dependency insights setting.
message (String)	A message confirming the result of updating the members can view dependency insights setting.

updateEnterpriseOrganizationProjectsSetting ⧉

Sets whether organization projects are enabled for an enterprise.

Input fields for `updateEnterpriseOrganizationProjectsSetting`

- input (UpdateEnterpriseOrganizationProjectsSettingInput!)

Return fields for `updateEnterpriseOrganizationProjectsSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
enterprise (Enterprise)	The enterprise with the updated organization projects setting.
message (String)	A message confirming the result of updating the organization projects setting.

updateEnterpriseOwnerOrganizationRole ⧉

Updates the role of an enterprise owner with an organization.

Input fields for `updateEnterpriseOwnerOrganizationRole`

- `input` ([UpdateEnterpriseOwnerOrganizationRoleInput!](#))

Return fields for `updateEnterpriseOwnerOrganizationRole`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`message` ([String](#))	A message confirming the result of changing the owner's organization role.

updateEnterpriseProfile 🔗

Updates an enterprise's profile.

Input fields for `updateEnterpriseProfile`

- `input` ([UpdateEnterpriseProfileInput!](#))

Return fields for `updateEnterpriseProfile`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`enterprise` ([Enterprise](#))	The updated enterprise.

updateEnterpriseRepositoryProjectsSetting 🔗

Sets whether repository projects are enabled for a enterprise.

Input fields for `updateEnterpriseRepositoryProjectsSetting`

- `input` ([UpdateEnterpriseRepositoryProjectsSettingInput!](#))

Return fields for `updateEnterpriseRepositoryProjectsSetting`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`enterprise` ([Enterprise](#))	The enterprise with the updated repository projects setting.
`message` ([String](#))	A message confirming the result of updating the repository projects setting.

updateEnterpriseTeamDiscussionsSetting 🔗

Sets whether team discussions are enabled for an enterprise.

Input fields for `updateEnterpriseTeamDiscussionsSetting`

- `input` ([UpdateEnterpriseTeamDiscussionsSettingInput!](#))

Return fields for `updateEnterpriseTeamDiscussionsSetting`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`enterprise` ([Enterprise](#))	The enterprise with the updated team discussions setting.
`message` ([String](#))	A message confirming the result of updating the team discussions setting.

updateEnterpriseTwoFactorAuthenticationRequiredSetting 🔗

Sets whether two factor authentication is required for all users in an enterprise.

Input fields for `updateEnterpriseTwoFactorAuthenticationRequiredSetting`

- `input` ([UpdateEnterpriseTwoFactorAuthenticationRequiredSettingInput!](#))

Return fields for `updateEnterpriseTwoFactorAuthenticationRequiredSetting`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`enterprise` ([Enterprise](#))	The enterprise with the updated two factor authentication required setting.
`message` ([String](#))	A message confirming the result of updating the two factor authentication required setting.

updateEnvironment 🔗

Updates an environment.

Input fields for `updateEnvironment`

- `input` ([UpdateEnvironmentInput!](#))

Return fields for `updateEnvironment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.

| environment (Environment) | The updated environment. |

updateIpAllowListEnabledSetting 🔗

Sets whether an IP allow list is enabled on an owner.

Input fields for `updateIpAllowListEnabledSetting`

- input (UpdateIpAllowListEnabledSettingInput!)

Return fields for `updateIpAllowListEnabledSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
owner (IpAllowListOwner)	The IP allow list owner on which the setting was updated.

updateIpAllowListEntry 🔗

Updates an IP allow list entry.

Input fields for `updateIpAllowListEntry`

- input (UpdateIpAllowListEntryInput!)

Return fields for `updateIpAllowListEntry`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
ipAllowListEntry (IpAllowListEntry)	The IP allow list entry that was updated.

updateIpAllowListForInstalledAppsEnabledSetting 🔗

Sets whether IP allow list configuration for installed GitHub Apps is enabled on an owner.

Input fields for `updateIpAllowListForInstalledAppsEnabledSetting`

- input (UpdateIpAllowListForInstalledAppsEnabledSettingInput!)

Return fields for `updateIpAllowListForInstalledAppsEnabledSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
owner (IpAllowListOwner)	The IP allow list owner on which the setting was updated.

updateIssue 🔗

Updates an Issue.

Input fields for `updateIssue`

- `input` (`UpdateIssueInput!`)

Return fields for `updateIssue`

Name	Description
`actor` (`Actor`)	Identifies the actor who performed the event.
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`issue` (`Issue`)	The issue.

updateIssueComment 🔗

Updates an IssueComment object.

Input fields for `updateIssueComment`

- `input` (`UpdateIssueCommentInput!`)

Return fields for `updateIssueComment`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`issueComment` (`IssueComment`)	The updated comment.

updateLabel 🔗

Updates an existing label.

> **Preview notice**
>
> `updateLabel` is available under the Labels preview. During the preview period, the API may change without notice.

Input fields for `updateLabel`

- `input` (`UpdateLabelInput!`)

Return fields for `updateLabel`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
label (Label)	The updated label.

updateNotificationRestrictionSetting 🔗

Update the setting to restrict notifications to only verified or approved domains available to an owner.

Input fields for updateNotificationRestrictionSetting

- input (UpdateNotificationRestrictionSettingInput!)

Return fields for updateNotificationRestrictionSetting

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
owner (VerifiableDomainOwner)	The owner on which the setting was updated.

updateOrganizationAllowPrivateRepositoryForkingSetting 🔗

Sets whether private repository forks are enabled for an organization.

Input fields for updateOrganizationAllowPrivateRepositoryForkingSetting

- input (UpdateOrganizationAllowPrivateRepositoryForkingSettingInput!)

Return fields for updateOrganizationAllowPrivateRepositoryForkingSetting

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
message (String)	A message confirming the result of updating the allow private repository forking setting.
organization (Organization)	The organization with the updated allow private repository forking setting.

updateOrganizationWebCommitSignoffSetting 🔗

Sets whether contributors are required to sign off on web-based commits for repositories in an organization.

Input fields for updateOrganizationWebCommitSignoffSetting

- input ([UpdateOrganizationWebCommitSignoffSettingInput!](#))

Return fields for `updateOrganizationWebCommitSignoffSetting`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
message ([String](#))	A message confirming the result of updating the web commit signoff setting.
organization ([Organization](#))	The organization with the updated web commit signoff setting.

updateProject 🔗

Updates an existing project.

Input fields for `updateProject`

- input ([UpdateProjectInput!](#))

Return fields for `updateProject`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
project ([Project](#))	The updated project.

updateProjectCard 🔗

Updates an existing project card.

Input fields for `updateProjectCard`

- input ([UpdateProjectCardInput!](#))

Return fields for `updateProjectCard`

Name	Description
clientMutationId ([String](#))	A unique identifier for the client performing the mutation.
projectCard ([ProjectCard](#))	The updated ProjectCard.

updateProjectColumn 🔗

Updates an existing project column.

Input fields for `updateProjectColumn`

- `input` (UpdateProjectColumnInput!)

Return fields for `updateProjectColumn`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`projectColumn` (ProjectColumn)	The updated project column.

updateProjectV2 🔗

Updates an existing project (beta).

Input fields for `updateProjectV2`

- `input` (UpdateProjectV2Input!)

Return fields for `updateProjectV2`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`projectV2` (ProjectV2)	The updated Project.

updateProjectV2Collaborators 🔗

Update the collaborators on a team or a project.

Input fields for `updateProjectV2Collaborators`

- `input` (UpdateProjectV2CollaboratorsInput!)

Return fields for `updateProjectV2Collaborators`

Name	Description
`clientMutationId` (String)	A unique identifier for the client performing the mutation.
`collaborators` (ProjectV2ActorConnection)	The collaborators granted a role.

updateProjectV2DraftIssue 🔗

Updates a draft issue within a Project.

Input fields for `updateProjectV2DraftIssue`

- `input` (UpdateProjectV2DraftIssueInput!)

Return fields for `updateProjectV2DraftIssue`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
draftIssue (DraftIssue)	The draft issue updated in the project.

updateProjectV2ItemFieldValue 🔗

This mutation updates the value of a field for an item in a Project. Currently only single-select, text, number, date, and iteration fields are supported.

Input fields for `updateProjectV2ItemFieldValue`

- input (UpdateProjectV2ItemFieldValueInput!)

Return fields for `updateProjectV2ItemFieldValue`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
projectV2Item (ProjectV2Item)	The updated item.

updateProjectV2ItemPosition 🔗

This mutation updates the position of the item in the project, where the position represents the priority of an item.

Input fields for `updateProjectV2ItemPosition`

- input (UpdateProjectV2ItemPositionInput!)

Return fields for `updateProjectV2ItemPosition`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
items (ProjectV2ItemConnection)	The items in the new order.

updatePullRequest 🔗

Update a pull request.

Input fields for `updatePullRequest`

- input (UpdatePullRequestInput!)

Return fields for `updatePullRequest`

Name	Description
`actor` ([Actor](#))	Identifies the actor who performed the event.
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The updated pull request.

updatePullRequestBranch 🔗

Merge or Rebase HEAD from upstream branch into pull request branch.

Input fields for `updatePullRequestBranch`

- `input` ([UpdatePullRequestBranchInput!](#))

Return fields for `updatePullRequestBranch`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequest` ([PullRequest](#))	The updated pull request.

updatePullRequestReview 🔗

Updates the body of a pull request review.

Input fields for `updatePullRequestReview`

- `input` ([UpdatePullRequestReviewInput!](#))

Return fields for `updatePullRequestReview`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`pullRequestReview` ([PullRequestReview](#))	The updated pull request review.

updatePullRequestReviewComment 🔗

Updates a pull request review comment.

Input fields for `updatePullRequestReviewComment`

- `input` ([UpdatePullRequestReviewCommentInput!](#))

Return fields for `updatePullRequestReviewComment`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`pullRequestReviewComment` (`PullRequestReviewComment`)	The updated comment.

updateRef 🔗

Update a Git Ref.

Input fields for `updateRef`

- `input` (`UpdateRefInput!`)

Return fields for `updateRef`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`ref` (`Ref`)	The updated Ref.

updateRefs 🔗

Creates, updates and/or deletes multiple refs in a repository.

This mutation takes a list of `RefUpdate` s and performs these updates on the repository. All updates are performed atomically, meaning that if one of them is rejected, no other ref will be modified.

`RefUpdate.beforeOid` specifies that the given reference needs to point to the given value before performing any updates. A value of `0000000000000000000000000000000000000000` can be used to verify that the references should not exist.

`RefUpdate.afterOid` specifies the value that the given reference will point to after performing all updates. A value of `0000000000000000000000000000000000000000` can be used to delete a reference.

If `RefUpdate.force` is set to `true` , a non-fast-forward updates for the given reference will be allowed.

> **Preview notice**
>
> `updateRefs` is available under the [Update refs preview update multiple refs in a single operation preview](). During the preview period, the API may change without notice.

Input fields for `updateRefs`

- `input` (`UpdateRefsInput!`)

Return fields for `updateRefs`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.

updateRepository 🔗

Update information about a repository.

Input fields for `updateRepository`

- input (UpdateRepositoryInput!)

Return fields for `updateRepository`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
repository (Repository)	The updated repository.

updateRepositoryWebCommitSignoffSetting 🔗

Sets whether contributors are required to sign off on web-based commits for a repository.

Input fields for `updateRepositoryWebCommitSignoffSetting`

- input (UpdateRepositoryWebCommitSignoffSettingInput!)

Return fields for `updateRepositoryWebCommitSignoffSetting`

Name	Description
clientMutationId (String)	A unique identifier for the client performing the mutation.
message (String)	A message confirming the result of updating the web commit signoff setting.
repository (Repository)	The updated repository.

updateSubscription 🔗

Updates the state for subscribable subjects.

Input fields for `updateSubscription`

- input (UpdateSubscriptionInput!)

Return fields for `updateSubscription`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`subscribable` (`Subscribable`)	The input subscribable entity.

updateTeamDiscussion 🔗

Updates a team discussion.

Input fields for `updateTeamDiscussion`

- `input` (`UpdateTeamDiscussionInput!`)

Return fields for `updateTeamDiscussion`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`teamDiscussion` (`TeamDiscussion`)	The updated discussion.

updateTeamDiscussionComment 🔗

Updates a discussion comment.

Input fields for `updateTeamDiscussionComment`

- `input` (`UpdateTeamDiscussionCommentInput!`)

Return fields for `updateTeamDiscussionComment`

Name	Description
`clientMutationId` (`String`)	A unique identifier for the client performing the mutation.
`teamDiscussionComment` (`TeamDiscussionComment`)	The updated comment.

updateTeamReviewAssignment 🔗

Updates team review assignment.

> **Preview notice**
>
> `updateTeamReviewAssignment` is available under the Team review assignments preview. During the preview period, the API may change without notice.

Input fields for `updateTeamReviewAssignment`

- `input` ([UpdateTeamReviewAssignmentInput!](#))

Return fields for `updateTeamReviewAssignment`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`team` ([Team](#))	The team that was modified.

updateTeamsRepository 🔗

Update team repository.

Input fields for `updateTeamsRepository`

- `input` ([UpdateTeamsRepositoryInput!](#))

Return fields for `updateTeamsRepository`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`repository` ([Repository](#))	The repository that was updated.
`teams` ([[Team!]](#))	The teams granted permission on the repository.

updateTopics 🔗

Replaces the repository's topics with the given topics.

Input fields for `updateTopics`

- `input` ([UpdateTopicsInput!](#))

Return fields for `updateTopics`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`invalidTopicNames` ([[String!]](#))	Names of the provided topics that are not valid.
`repository` ([Repository](#))	The updated repository.

verifyVerifiableDomain 🔗

Verify that a verifiable domain has the expected DNS record.

Input fields for `verifyVerifiableDomain`

- `input` ([VerifyVerifiableDomainInput!](#))

Return fields for `verifyVerifiableDomain`

Name	Description
`clientMutationId` ([String](#))	A unique identifier for the client performing the mutation.
`domain` ([VerifiableDomain](#))	The verifiable domain that was verified.