# **Queries**

#### In this article

About queries

codeOfConduct

codesOfConduct

enterprise

enterpriseAdministratorInvitation

enterprise Administrator Invitation By Token

license

licenses

marketplaceCategories

marketplaceCategory

marketplaceListing

marketplaceListings

meta

node

nodes

organization

rateLimit

relay

repository

repositoryOwner

resource

search

securityAdvisories

securityAdvisory

securityVulnerabilities

sponsorables

topic

user

viewer

The query type defines GraphQL operations that retrieve data from the server.

## **About queries 3**

Every GraphQL schema has a root type for both queries and mutations. The <u>query type</u> defines GraphQL operations that retrieve data from the server.

For more information, see "Forming calls with GraphQL."

Note: For GitHub App requests made with user access tokens, you should use separate queries

for issues and pull requests. For example, use the <code>is:issue</code> or <code>is:pull-request</code> filters and their equivalents. Using the <code>search</code> connection to return a combination of issues and pull requests in a single query will result in an empty set of nodes.

#### codeOfConduct @

Look up a code of conduct by its key.

**Type:** CodeOfConduct

#### **Arguments for codeOfConduct**

Name	Description
key ( <u>String!</u> )	The code of conduct's key.

#### codesOfConduct @

Look up a code of conduct by its key.

Type: [CodeOfConduct]

### enterprise &

Look up an enterprise by URL slug.

Type: Enterprise

#### **Arguments for enterprise**

Name	Description
<pre>invitationToken ( <u>String</u> )</pre>	The enterprise invitation token.
slug ( <u>String!</u> )	The enterprise URL slug.

## enterpriseAdministratorInvitation $\mathscr O$

Look up a pending enterprise administrator invitation by invitee, enterprise and role.

**Type:** EnterpriseAdministratorInvitation

#### **Arguments for** enterpriseAdministratorInvitation

Name	Description
<pre>enterpriseSlug ( String! )</pre>	The slug of the enterprise the user was invited to join.
<pre>role ( EnterpriseAdministratorRole! )</pre>	The role for the business member invitation.
userLogin ( <a href="String!">String!</a> )	The login of the user invited to join the business.

### enterpriseAdministratorInvitationByToken &

Look up a pending enterprise administrator invitation by invitation token.

**Type:** EnterpriseAdministratorInvitation

#### **Arguments for** enterpriseAdministratorInvitationByToken

Name	Description
<pre>invitationToken ( String! )</pre>	The invitation token sent with the invitation email.

#### license ₽

Look up an open source license by its key.

Type: <u>License</u>

#### **Arguments for license**

Name	Description
key (String!)	The license's downcased SPDX ID.

#### licenses &

Return a list of known open source licenses.

Type: [License]!

### marketplaceCategories @

Get alphabetically sorted list of Marketplace categories.

**Type:** [MarketplaceCategory!]!

#### **Arguments for marketplaceCategories**

Name	Description
excludeEmpty ( <u>Boolean</u> )	Exclude categories with no listings.
excludeSubcategories ( <u>Boolean</u> )	Returns top level categories only, excluding any subcategories.
<pre>includeCategories ( [String!] )</pre>	Return only the specified categories.

## marketplaceCategory &

Look up a Marketplace category by its slug.

Type: MarketplaceCategory

#### **Arguments for marketplaceCategory**

Name	Description
slug ( <u>String!</u> )	The URL slug of the category.
useTopicAliases ( <u>Boolean</u> )	Also check topic aliases for the category slug.

## $\mathbf{marketplaceListing}~\mathscr{O}$

Look up a single Marketplace listing.

**Type:** MarketplaceListing

### **Arguments for marketplaceListing**

Name	Description
<pre>slug ( String! )</pre>	Select the listing that matches this slug. It's the short name of the listing used in its URL.

## marketplaceListings &

Look up Marketplace listings.

**Type:** MarketplaceListingConnection!

### **Arguments for marketplaceListings**

Name	Description
adminId ( ID )	Select listings that can be administered by the specified user.
after ( <u>String</u> )	Returns the elements in the list that come after the specified cursor.
allStates ( Boolean )	Select listings visible to the viewer even if they are not approved. If omitted or false, only approved listings will be returned.
before ( <u>String</u> )	Returns the elements in the list that come before the specified cursor.
<pre>categorySlug ( <u>String</u> )</pre>	Select only listings with the given category.
first ( <u>Int</u> )	Returns the first $n$ elements from the list.
last ( <u>Int</u> )	Returns the last $n$ elements from the list.
organizationId ( <u>ID</u> )	Select listings for products owned by the specified organization.
<pre>primaryCategoryOnly ( Boolean )</pre>	Select only listings where the primary category matches the given category slug.  The default value is false.
slugs ( <a href="[String]">[String]</a> )	Select the listings with these slugs, if they are visible to the viewer.
usaTaniaAliassa / Baalaan \	Also shock tonic aliasas for the category slug

useropicAtlases ( <u>Bootean</u> )	Also check topic allases for the category slug.
viewerCanAdmin ( <u>Boolean</u> )	Select listings to which user has admin access. If omitted, listings visible to the viewer are returned.
withFreeTrialsOnly ( Boolean )	Select only listings that offer a free trial.  The default value is false.

### meta 🔗

Return information about the GitHub instance.

**Type:** <u>GitHubMetadata!</u>

### node 🔗

Fetches an object given its ID.

Type: Node

### Arguments for node

Name	Description
id ( <u>ID!</u> )	ID of the object.

### nodes 🔗

Lookup nodes by a list of IDs.

Type: [Node]!

#### **Arguments for nodes**

Name	Description
ids ( [ID!]! )	The list of node IDs.

## organization $\mathscr O$

Lookup a organization by login.

**Type:** Organization

### **Arguments for organization**

Name	Description
<pre>login (<u>String!</u>)</pre>	The organization's login.

### rateLimit 🔗

The client's rate limit information.

**Type:** RateLimit

#### **Arguments for rateLimit**

Name	Description
dryRun ( <u>Boolean</u> )	If true, calculate the cost for the query without evaluating it.  The default value is false.

## relay 🔗

Workaround for re-exposing the root query object. (Refer to <a href="https://github.com/facebook/relay/issues/112">https://github.com/facebook/relay/issues/112</a> for more information.).

Type: Query!

### repository &

Lookup a given repository by the owner and repository name.

**Type:** Repository

#### **Arguments for repository**

Name	Description
followRenames ( <u>Boolean</u> )	Follow repository renames. If disabled, a repository referenced by its old name will return an error.  The default value is true.
name ( <u>String!</u> )	The name of the repository.
owner (String!)	The login field of a user or organization.

## repositoryOwner 🔗

Lookup a repository owner (ie. either a User or an Organization) by login.

Type: RepositoryOwner

#### **Arguments for repository0wner**

Name	Description
login ( <u>String!</u> )	The username to lookup the owner by.

#### resource @

Lookup resource by a URL.

Type: <u>UniformResourceLocatable</u>

#### **Arguments for resource**

Name	Description
url (URI!)	The URL.

#### search 🔗

Perform a search across resources, returning a maximum of 1,000 results.

**Type:** <u>SearchResultItemConnection!</u>

### **Arguments for search**

Name	Description
after ( <u>String</u> )	Returns the elements in the list that come after the specified cursor.
before ( <u>String</u> )	Returns the elements in the list that come before the specified cursor.
first ( <u>Int</u> )	Returns the first $n$ elements from the list.
last ( <u>Int</u> )	Returns the last $n$ elements from the list.
query ( <u>String!</u> )	The search string to look for.
type ( <u>SearchType!</u> )	The types of search items to search within.

## securityAdvisories 🔗

GitHub Security Advisories.

**Type:** <u>SecurityAdvisoryConnection!</u>

## **Arguments for** securityAdvisories

Name	Description
after ( String )	Returns the elements in the list that come after the specified cursor.
before (String)	Returns the elements in the list that come before the specified cursor.
<pre>classifications ( [SecurityAdvisoryClassification!] )</pre>	A list of classifications to filter advisories by.
first ( Int )	Returns the first $n$ elements from the list.
$identifier \ \ ( \ \underline{SecurityAdvisoryIdentifierFilter} \ )$	Filter advisories by identifier, e.g. GHSA or CVE.
last ( <u>Int</u> )	Returns the last $n$ elements from the list.
orderBy ( <u>SecurityAdvisoryOrder</u> )	Ordering options for the returned topics.
<pre>publishedSince ( DateTime )</pre>	Filter advisories to those published since a time in the past.
<pre>updatedSince ( DateTime )</pre>	Filter advisories to those updated since a time in the past.

## securityAdvisory &

Fetch a Security Advisory by its GHSA ID.

**Type:** <u>SecurityAdvisory</u>

#### **Arguments for** securityAdvisory

Name	Description
<pre>ghsaId ( String! )</pre>	GitHub Security Advisory ID.

## securityVulnerabilities 🔗

Software Vulnerabilities documented by GitHub Security Advisories.

**Type:** <u>SecurityVulnerabilityConnection!</u>

### **Arguments for** securityVulnerabilities

Name	Description
after ( <u>String</u> )	Returns the elements in the list that come after the specified cursor.
before ( <u>String</u> )	Returns the elements in the list that come before the specified cursor.
<pre>classifications ( [SecurityAdvisoryClassification!] )</pre>	A list of advisory classifications to filter vulnerabilities by.
ecosystem ( <u>SecurityAdvisoryEcosystem</u> )	An ecosystem to filter vulnerabilities by.
first ( <u>Int</u> )	Returns the first <i>n</i> elements from the list.
last ( <u>Int</u> )	Returns the last $n$ elements from the list.
<pre>orderBy ( <u>SecurityVulnerabilityOrder</u> )</pre>	Ordering options for the returned topics.
package ( <u>String</u> )	A package name to filter vulnerabilities by.
severities ( [SecurityAdvisorySeverity!] )	A list of severities to filter vulnerabilities by.

## sponsorables @

Users and organizations who can be sponsored via GitHub Sponsors.

**Type:** <u>SponsorableItemConnection!</u>

#### **Arguments for sponsorables**

Name	Description
after ( <u>String</u> )	Returns the elements in the list that come after the specified cursor.
before ( <u>String</u> )	Returns the elements in the list that come before the specified cursor.

<pre>dependencyEcosystem ( SecurityAdvisoryEcosystem )</pre>	Optional filter for which dependencies should be checked for sponsorable owners. Only sponsorable owners of dependencies in this ecosystem will be included. Used when onlyDependencies = true.  Upcoming Change on 2022-07-01 UTC  Description: dependencyEcosystem will be removed. Use the ecosystem argument instead.  Reason: The type is switching from SecurityAdvisoryEcosystem to DependencyGraphEcosystem.
ecosystem ( <a href="DependencyGraphEcosystem">DependencyGraphEcosystem</a> )	Optional filter for which dependencies should be checked for sponsorable owners. Only sponsorable owners of dependencies in this ecosystem will be included. Used when onlyDependencies = true.
first (Int)	Returns the first $n$ elements from the list.
last ( <u>Int</u> )	Returns the last $n$ elements from the list.
onlyDependencies ( Boolean )	Whether only sponsorables who own the viewer's dependencies will be returned. Must be authenticated to use. Can check an organization instead for their dependencies owned by sponsorables by passing orgLoginForDependencies.  The default value is false.
orderBy ( <u>SponsorableOrder</u> )	Ordering options for users and organizations returned from the connection.
orgLoginForDependencies ( <u>String</u> )	Optional organization username for whose dependencies should be checked. Used when onlyDependencies = true. Omit to check your own dependencies. If you are not an administrator of the organization, only dependencies from its public repositories will be considered.

## topic 🔗

Look up a topic by name.

Type: Topic

## Arguments for topic

Name	Description
name ( <u>String!</u> )	The topic's name.

### user 🔗

Lookup a user by login.

Type: <u>User</u>

Arguments for user

Name	Description
login ( <u>String!</u> )	The user's login.
viewer ₽	

The currently authenticated user.

Type: User!

## Legal

© 2023 GitHub, Inc. <u>Terms</u> <u>Privacy</u> <u>Status</u> <u>Pricing</u> <u>Expert services</u> <u>Blog</u>