

[Job Search documentation](https://cloud.google.com/talent-solution/job-search/) (https://cloud.google.com/talent-solution/job-search/)

[Documentation](#)

Search overview

Cloud Talent Solution allows users to search against jobs that you store in the Job Search index. Job Search is designed to be customizable and can be altered to meet your business needs. This page is an overview of the search functions available in CTS Job Search.

Making search requests

To make a Job Search request, send a POST request to the `jobs:search` (https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.jobs/search) endpoint, passing the search data in the body.

Commonly used fields (optional)

- `JobQuery`: The `JobQuery` object consists of two components, a `query` string and the filter(s) applied to that string. This feature allows your users to search on a query string (such as "communications assistant") using one or more filters (location, commute time, compensation, and so on). See [Search Filters](https://cloud.google.com/talent-solution/job-search/docs/search-filters) (https://cloud.google.com/talent-solution/job-search/docs/search-filters) for more information and a list of available filter options.
- `searchMode`: Specifies the type of search to apply. Options are:
 - `JOB_SEARCH` (recommended): This is the default search mode, which searches against all Job objects. The most relevant jobs are returned to the user in the order of relevance to the search query.

- **FEATURED_JOB_SEARCH**: Searches only for Job objects with a `promotionValue` greater than 0. Results are ordered by the highest `promotionValue` first; in the case of a tie, relevance is used to decide position. See [Featured Jobs](#) (<https://cloud.google.com/talent-solution/job-search/docs/featured-jobs>) for more information.
- **jobView**: Defines the number of fields that are returned for each matching Job object. Can be `JOB_VIEW_SMALL` (default), `JOB_VIEW_MINIMAL`, or `JOB_VIEW_FULL`. Descriptions for each option can be found in the [JobView](#) (<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/JobView>) reference documentation.
- **orderBy**: The sort order for returned Jobs. See the [reference](#) (https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.jobs/search#body.request_body.FIELDS.order_by) documentation for allowed values. The default is to sort by relevance, descending (`RELEVANCE_DESC`).

Required fields

The following field is required:

- **requestMetadata**: Contains user and session information for this search request.

RequestMetadata

Every search and histogram request must include a **RequestMetadata**

(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/RequestMetadata>) object. This object contains information about the user initiating the search. It is important for two main reasons:

- **User consistency**: Cloud Talent Solution runs several versions of the underlying models simultaneously to improve search relevance. CTS ensures that a job seeker (identified by [userId](#) (https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/RequestMetadata#FIELDS.user_id), [sessionId](#) (https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/RequestMetadata#FIELDS.session_id), and [domain](#)

(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/RequestMetadata#FIELDS.domain>) has a consistent experience. These fields should be obfuscated.

- **Search improvements:** The tuning algorithms are based on the information provided in the `RequestMetadata` fields along with the job seekers' data you provide. This improves search result relevance.

Using search operators

The search operators in the table below may be used alone or in combination.

Operator Description	
()	Groups one or more keywords together.
OR	Searches for results relevant to the keywords or groups of keywords on either side of the operator. Example: "software OR java" returns jobs with <i>either</i> "software" or "java" in their metadata.
AND	Searches for results relevant to the keywords or groups of keywords on both sides of the operator. Example: "(software OR java) AND (marketing)" returns jobs that are <i>either</i> relevant to both "software" <i>and</i> "marketing", or jobs that are relevant to both "java" <i>and</i> "marketing".
NOT	Excludes results relevant to the keyword or group of keywords immediately following the operator. The keyword or group of keywords must be in parentheses.
-	Excludes results relevant to the keyword or group of keywords after to the operator. Example: "software -java" returns jobs relevant to "software" but <i>excludes</i> jobs that are relevant to "java".
req or requisition:	Returns jobs with the specified requisition ID. Example, req:1234 returns jobs with requisition ID 1234.

Job seeker queries

Search terms can also be detected as actual Job-related entities. For example, if a user queries for "intern", Cloud Talent Solution treats this as the employment type `INTERN` and filters on Jobs with the designation `EMPLOYMENT_TYPE - INTERN`. Therefore, even a Job object with "intern" in its description would not be returned if the `EMPLOYMENT_TYPE` field for that Job was set to something else.

Note: It is very important to specify the employment type for each Job object. Not doing so negatively impacts the job seeker's experience.

Troubleshooting

Each search request generates a unique search `requestID` in the Job Search system. This value is used to track the parameters of that search as well as the response. This is crucial in troubleshooting issues related to specific searches or CRUD requests, and should be appropriately logged. The `requestID` is returned inside the `ResponseMetadata`

(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/ResponseMetadata>) field in the search response body.

(<https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.jobs/search#response-body>)

or the associated CRUD call response object.

For support issues that relate to relevance, search inquiries, indexing inquiries, and so on, please file a ticket (<https://cloud.google.com/talent-solution/docs/support>) with our support team and include relevant `requestIDs` or the requests themselves.

Result counts

The result count is used to communicate the number of search results to the job seeker. This doesn't always correspond to the histogram

(<https://cloud.google.com/talent-solution/job-search/docs/histogram>) results or the actual number of jobs returned. Job Search returns a maximum of 5000 jobs.

Pagination (optional)

By default, Job Search returns 100 results in a single call (or fewer, if the total number of results is less than 100). To view more than 100 results, you can implement pagination. If the total number of results exceeds the maximum of 5000, all pages after the page containing the 5000th result will be blank. The fields below control pagination:

- `pageSize`: The number of results returned.

- **nextPageToken**: A unique token, returned with the results of each page of search results, that specifies the starting position for the next page of results. This field is left empty if there are no more results.
- **offset**: Specifies the offset between the relevance ranking of the job returned at the top of a given page and 1 (the most relevant Job returned). As an example, a user performs a search and **pageSize** is set to 10. The first page returns the #1 job at the top, therefore the **offset** value for this page is 0. The next page returns job #11 at the top; **offset** for this page is 10.

There are two main ways to implement pagination. You can use **pageSize** and **nextPageToken** to directly point to the next page of results (each page has a new **nextPageToken** value that points to the subsequent page). If you use this method you will always start at the top of the results and page through in sequential order. Alternatively, you can use **pageSize** with **offset** to directly specify which numbered result appears at the top of each page. Using **offset** instead of **nextPageToken** allows you to jump to any point in the result set.

We recommend that you set the page size to 20 results or less to avoid latency issues. As a best practice, issue a separate search call per page rendered to the job seeker. The alternative, a single search call to return a larger number of jobs and paginating on the client side, impedes the ability of the machine learning model to learn accurately through the Jobs Analytics Framework.

Spell checking

Spell check is an optional feature of a CTS search request. It accounts for potentially misspelled search terms, corrects them and returns results based on the corrected query. For example, when spell check is enabled, if a job seeker searches for "manaer" instead of "manager", CTS returns jobs for "manager". Spell check can be enabled/disabled by setting the **disableSpellCheck** field of the search request. If spell checking is enabled, the corrected query can be found in the search response, in the **spellResult** field.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (https://www.apache.org/licenses/LICENSE-2.0). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (https://developers.google.com/terms/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 8, 2019.