flickr.photos.search

Return a list of photos matching some criteria. Only photos visible to the calling user will be returned. To return private or semi-private photos, the caller must be authenticated with 'read' permissions, and have permission to view the photos. Unauthenticated calls will only return public photos.

Authentifizierung

Für diese Methode ist keine Authentifizierung erforderlich.

Argumente

`api_key` (Erforderlich)
> Your API application key. [See here](#) for more details.

`user_id` (Optional)
> The NSID of the user who's photo to search. If this parameter isn't passed then everybody's public photos will be searched. A value of "me" will search against the calling user's photos for authenticated calls.

`tags` (Optional)
> A comma-delimited list of tags. Photos with one or more of the tags listed will be returned. You can exclude results that match a term by prepending it with a - character.

`tag_mode` (Optional)
> Either 'any' for an OR combination of tags, or 'all' for an AND combination. Defaults to 'any' if not specified.

`text` (Optional)
> A free text search. Photos who's title, description or tags contain the text will be returned. You can exclude results that match a term by prepending it with a - character.

`min_upload_date` (Optional)
> Minimum upload date. Photos with an upload date greater than or equal to this value will be returned. The date can be in the form of a unix timestamp or mysql datetime.

`max_upload_date` (Optional)
> Maximum upload date. Photos with an upload date less than or equal to this value will be returned. The date can be in the form of a unix timestamp or mysql datetime.

`min_taken_date` (Optional)
> Minimum taken date. Photos with an taken date greater than or equal to this value will be returned. The date can be in the form of a mysql datetime or unix timestamp.

`max_taken_date` (Optional)
> Maximum taken date. Photos with an taken date less than or equal to this value will be returned. The date can be in the form of a mysql datetime or unix timestamp.

`license` (Optional)
> The license id for photos (for possible values see the flickr.photos.licenses.getInfo method). Multiple licenses may be comma-separated.

`sort` (Optional)
> The order in which to sort returned photos. Deafults to date-posted-desc (unless you are doing a radial geo query, in which case the default sorting is by ascending distance from the point specified). The possible values are: date-posted-asc, date-posted-desc, date-taken-asc, date-taken-desc, interestingness-desc, interestingness-asc, and relevance.

`privacy_filter` (Optional)
> Return photos only matching a certain privacy level. This only applies when making an authenticated call to view photos you own. Valid values are:
>
> - 1 public photos
> - 2 private photos visible to friends
> - 3 private photos visible to family
> - 4 private photos visible to friends & family

- 5 completely private photos

bbox (Optional)

A comma-delimited list of 4 values defining the Bounding Box of the area that will be searched.

The 4 values represent the bottom-left corner of the box and the top-right corner, minimum_longitude, minimum_latitude, maximum_longitude, maximum_latitude.

Longitude has a range of -180 to 180 , latitude of -90 to 90. Defaults to -180, -90, 180, 90 if not specified.

Unlike standard photo queries, geo (or bounding box) queries will only return 250 results per page.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters — If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

accuracy (Optional)

Recorded accuracy level of the location information. Current range is 1-16 :

- World level is 1
- Country is ~3
- Region is ~6
- City is ~11
- Street is ~16

Defaults to maximum value if not specified.

safe_search (Optional)

Safe search setting:

- 1 for safe.
- 2 for moderate.
- 3 for restricted.

(Please note: Un-authed calls can only see Safe content.)

content_type (Optional)

Content Type setting:

- 1 for photos only.
- 2 for screenshots only.
- 3 for 'other' only.
- 4 for photos and screenshots.
- 5 for screenshots and 'other'.
- 6 for photos and 'other'.
- 7 for photos, screenshots, and 'other' (all).

machine_tags (Optional)

Aside from passing in a fully formed machine tag, there is a special syntax for searching on specific properties :

- Find photos using the 'dc' namespace : `"machine_tags" => "dc:"`
- Find photos with a title in the 'dc' namespace : `"machine_tags" => "dc:title="`
- Find photos titled "mr. camera" in the 'dc' namespace : `"machine_tags" => "dc:title=\"mr. camera\""`

- Find photos whose value is "mr. camera" : `"machine_tags" => "*:*=\"mr. camera\""`
- Find photos that have a title, in any namespace : `"machine_tags" => "*:title="`
- Find photos that have a title, in any namespace, whose value is "mr. camera" : `"machine_tags" => "*:title=\"mr. camera\""`
- Find photos, in the 'dc' namespace whose value is "mr. camera" : `"machine_tags" => "dc:*=\"mr. camera\""`

Multiple machine tags may be queried by passing a comma-separated list. The number of machine tags you can pass in a single query depends on the tag mode (AND or OR) that you are querying with. "AND" queries are limited to (16) machine tags. "OR" queries are limited to (8).

`machine_tag_mode` (Optional)

Either 'any' for an OR combination of tags, or 'all' for an AND combination. Defaults to 'any' if not specified.

`group_id` (Optional)

The id of a group who's pool to search. If specified, only matching photos posted to the group's pool will be returned.

`contacts` (Optional)

Search your contacts. Either 'all' or 'ff' for just friends and family. (Experimental)

`woe_id` (Optional)

A 32-bit identifier that uniquely represents spatial entities. (not used if bbox argument is present).

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

`place_id` (Optional)

A Flickr place id. (not used if bbox argument is present).

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

`media` (Optional)

Filter results by media type. Possible values are `all` (default), `photos` or `videos`

`has_geo` (Optional)

Any photo that has been geotagged, or if the value is "0" any photo that has *not* been geotagged.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

`geo_context` (Optional)

Geo context is a numeric value representing the photo's geotagginess beyond latitude and longitude. For example, you may wish to search for photos that were taken "indoors" or "outdoors".

The current list of context IDs is :

- **0**, not defined.
- **1**, indoors.
- **2**, outdoors.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

lat (Optional)

A valid latitude, in decimal format, for doing radial geo queries.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

lon (Optional)

A valid longitude, in decimal format, for doing radial geo queries.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters &emdash; If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

radius (Optional)

A valid radius used for geo queries, greater than zero and less than 20 miles (or 32 kilometers), for use with point-based geo queries. The default value is 5 (km).

radius_units (Optional)

The unit of measure when doing radial geo queries. Valid options are "mi" (miles) and "km" (kilometers). The default is "km".

is_commons (Optional)

Limit the scope of the search to only photos that are part of the [Flickr Commons project](). Default is false.

in_gallery (Optional)

Limit the scope of the search to only photos that are in a [gallery](). Default is false, search all photos.

is_getty (Optional)

Limit the scope of the search to only photos that are for sale on Getty. Default is false.

extras (Optional)

A comma-delimited list of extra information to fetch for each returned record. Currently supported fields are: `description`, `license`, `date_upload`, `date_taken`, `owner_name`, `icon_server`, `original_format`, `last_update`, `geo`, `tags`, `machine_tags`, `o_dims`, `views`, `media`, `path_alias`, `url_sq`, `url_t`, `url_s`, `url_m`, `url_z`, `url_l`, `url_o`

per_page (Optional)

Number of photos to return per page. If this argument is omitted, it defaults to 100. The maximum allowed value is 500.

page (Optional)

The page of results to return. If this argument is omitted, it defaults to 1.

Please note that Flickr will return at most the first 4,000 results for any given search query. If this is an issue, we recommend trying a more specific query.

Fehlercodes

1: Too many tags in ALL query
> When performing an 'all tags' search, you may not specify more than 20 tags to join together.

2: Unknown user
> A user_id was passed which did not match a valid flickr user.

3: Parameterless searches have been disabled
> To perform a search with no parameters (to get the latest public photos, please use flickr.photos.getRecent instead).

4: You don't have permission to view this pool
> The logged in user (if any) does not have permission to view the pool for this group.

10: Sorry, the Flickr search API is not currently available.
> The Flickr API search databases are temporarily unavailable

11: No valid machine tags
> The query styntax for the machine_tags argument did not validate.

12: Exceeded maximum allowable machine tags
> The maximum number of machine tags in a single query was exceeded.

100: Invalid API Key
> The API key passed was not valid or has expired.

105: Service currently unavailable
> The requested service is temporarily unavailable.

111: Format "xxx" not found
> The requested response format was not found.

112: Method "xxx" not found
> The requested method was not found.

114: Invalid SOAP envelope
> The SOAP envelope send in the request could not be parsed.

115: Invalid XML-RPC Method Call
> The XML-RPC request document could not be parsed.

116: Bad URL found
> One or more arguments contained a URL that has been used for abuse on Flickr.