On This Page

Targeting Search

Target Ad sets on a number of criteria you provide in targeting specs. Most targets are predefined values, such as country "Japan" or city "Tokyo".

Find valid values with Marketing API, Targeting Search: `https://graph.facebook.com/{API_VERSION}/search`. You must provide your query string in `UTF8` format.

Targeting Option Status

To verify current and/or planned status of targeting objects, use the `targeting_option_list` parameter:

```
curl -G \
  -d 'targeting_option_list=[<TARGETING_OPTION_ ID>,<TARGETING_OPTION_ID>]'
  -d 'type=targetingoptionstatus'
  https://graph.facebook.com/<API_VERSION>/search
```

The response:

```
{"data":[{"id":"<TARGETING_OPTION_ ID>","current_status":"NON-DELIVERABLE"},{"id":"<TARGETING_OPTION_ID>","current_status":"NON-DELIVERABLE","future_plan":[{"key":"2018-05-10T00:00:00+0000","value":"DEPRECATING"}]}]}
```

Return Fields

Field	Value
`current_status`	• `NORMAL` • `NON-DELIVERABLE` - Does not deliver, though the ad set may continue to deliver according to predetermined rules. • `DEPRECATING` - Ad sets targeted at this objective continue to deliver, but the object can't be used to create new or update ad sets. When updated, ad sets with with this term will be rejected, unless the term is removed. • `NON-DELIVERABLE-IN-EXCLUSION` - The object can't be used in targeting exclusions. • `UNKNOWN`
`future_plan`	Map of timestamp to status. Returns a map of dates and planned statuses, which are the same values as available under `current_status`.

Geographic

Search targeting by country, country group, city, state, zip code, and other geographic areas at `type=adgeolocation`. You can specify optional parameters with `type=adgeolocation`. To find the United States' country code:

```
curl -G \
  -d 'location_types=["country"]' \
  -d 'type=adgeolocation' \
  -d 'q=un' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "key": "GB",
      "name": "United Kingdom",
      "type": "country",
      "supports_city": false,
      "supports_region": false
    },
    {
      "key": "AE",
      "name": "United Arab Emirates",
      "type": "country",
      "supports_city": false,
      "supports_region": false
    },
    {
      "key": "UM",
      "name": "United States Minor Outlying Islands",
      "type": "country",
      "supports_city": false,
      "supports_region": false
    }
  ]
}
```

`key` is a fixed number unique in per category, such as countries or country groups. Other fields, including `name`, are subject to change. Use `key` to define Targeting Specs.

In the response:

- if `supports_region` is `true`, this country has region codes
- if `supports_city` is `true`, this country has city codes

Name	Description
`location_types` type: array	`country`, `country_group`, `region`, `city`, `zip`, `geo_market`, or `electoral_district`, latter only in US. `location_types` is preferred over `type=adcountry`, etc.
`region_id` type: int	Region to search from
`country_code` type: string	Country to search from: `country_code=US`

Countries

Every country you can target has a country code. Optional parameters for `type=adgeolocation&location_types=['country']`:

Name	Description
`q` type: string	The string to autocomplete values. To list all countries with `location_types=['country']`, leave this blank `q=`, and set limit to a large number `limit=1000`
`match_country_code` type: boolean	Defaults to `false`. Find country by country code. Match Country by `country_code` versus `name`

Country Group

All country groups have a code to search and get a list of countries. For all country groups named `mercosur`:

```
curl -G \
  -d 'location_types=["country_group"]' \
  -d 'type=adgeolocation' \
  -d 'q=mercosur' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "key": "mercosur",
      "name": "Mercosur",
      "type": "country_group",
      "country_codes": [
        "BR",
        "AR",
        "UY",
        "PY",
        "VE"
      ],
      "is_worldwide": false,
      "supports_region": true,
      "supports_city": true
    }
  ]
}
```

If `is_worldwide` is `true`, this is a worldwide country group. If `supports_region` is `true`, the country group has region codes. If `supports_city` is `true`, the group has city codes.

Regions

To search for all regions starting the code `al`:

```
curl -G \
  -d 'location_types=["region"]' \
  -d 'type=adgeolocation' \
  -d 'q=al' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "key": "3843",
      "name": "Alabama",
      "type": "region",
      "country_code": "US",
      "country_name": "United States",
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "3844",
      "name": "Alaska",
      "type": "region",
      "country_code": "US",
      "country_name": "United States",
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "527",
      "name": "Alberta",
      "type": "region",
      "country_code": "CA",
      "country_name": "Canada",
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "1089",
      "name": "Alsace",
      "type": "region",
      "country_code": "FR",
      "country_name": "France",
      "supports_region": true,
      "supports_city": true
    }
  ]
}
```

Options for `type=adgeolocation&location_types=['region']`:

Name	Description
`q` type: string	String to autocomplete values. To get all countries with `location_types=['region']`: provide no parameters, `q=`, and set the limit to a large number, `limit=1000`

If `supports_region` is `true`, you can target this region. If `supports_city` is `true`, the region has city codes.

Cities

Since March 2019, we have reclassified several cities to other roles, but you can continue to use `city`. The search will return results that were formerly cities.

To search codes for all cities starting with `Manhattan`:

```
curl -G \
     -d 'location_types=["city"]' \
     -d 'type=adgeolocation' \
     -d 'q=Manhattan' \
     -d 'access_token=ACCESS_TOKEN' \
     https://graph.facebook.com/VERSION/search
```

The response:

```
{
  "data": [
    {
      "key": "2447439",
      "name": "Manhattan",
      "type": "city",
      "country_code": "US",
      "country_name": "United States",
      "region": "Kansas",
      "region_id": 3859,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "2439596",
      "name": "Manhattan",
      "type": "city",
      "country_code": "US",
      "country_name": "United States",
      "region": "Illinois",
      "region_id": 3856,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "2479541",
      "name": "Manhattan",
      "type": "city",
      "country_code": "US",
      "country_name": "United States",
      "region": "Montana",
      "region_id": 3869,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "2428908",
      "name": "Manhattan",
      "type": "city",
      "country_code": "US",
      "country_name": "United States",
      "region": "Florida",
      "region_id": 3852,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "2703980",
      "name": "Manhattan",
      "type": "subcity",
      "country_code": "US",
      "country_name": "United States",
      "region": "New York",
      "region_id": 3875,
      "supports_region": true,
      "supports_city": true,
      "geo_hierarchy_level": "SUBCITY",
      "geo_hierarchy_name": "BOROUGH"
    },
   ...
```

If `supports_region` is true, the region for this city is available for targeting. If `supports_city` is set `true`, this city is available for targeting.

Geographic Areas

We have other geographical areas you can use to target. Some of these areas are not yet defined, as noted below.

Area	Description
`LARGE_GEO_AREA`	Known commonly as a district or governate covering hundreds of square kilometers or more. Example: `Akkar` in `Lebanon`.
`MEDIUM_GEO_AREA`	Known commonly as a county, covering more than one city. Example: `Henrico` county in the state of `Virginia` in `United States`
`SMALL_GEO_AREA`	Known commonly as a residential area near a city or town. Example: `El Rosario` near Marbella in `Spain`.
`SUBCITY`	Such as a borough. Example: `Brooklyn` in `New York`.
`NEIGHBORHOOD`	Area within a city. Example: `Barton Estates, Irving` in `Texas`.
`SUBNEIGHBORHOOD`	Not yet available.
`METRO_AREA`	Densely populated area surround a larger city. Not yet available.

The hierarchy of geographical areas is as follows, from largest to smallest:

- `REGION`
- `LARGE_GEO_AREA`
- `MEDIUM_GEO_AREA`
- `SMALL_GEO_AREA`
- `METRO_AREA`
- `CITY`
- `SUBCITY`
- `NEIGHBORHOOD`
- `SUBNEIGHBORHOOD`

Zip Code

You can also search zip codes to target on Facebook. For zip code search, we recommend `adgeolocation` with `location_types=['zip']`. Visit the Meta Helpcenter to view a list of countries with supported Zip codes.

Search zip codes starting with `9`:

```
curl -G \
  -d 'location_types=["zip"]' \
  -d 'type=adgeolocation' \
  -d 'q=9' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/LATEST-API-VERSION/search
```

The response:

```
{
  "data": [
    {
      "key": "US:90028",
      "name": "90028",
      "type": "zip",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "primary_city": "Los Angeles",
      "primary_city_id": 2420379,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "US:94110",
      "name": "94110",
      "type": "zip",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "primary_city": "San Francisco",
      "primary_city_id": 2421836,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "US:94501",
      "name": "94501",
      "type": "zip",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "primary_city": "Alameda",
      "primary_city_id": 2417628,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "US:95190",
      "name": "95190",
      "type": "zip",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "primary_city": "San Jose",
      "primary_city_id": 2421846,
      "supports_region": true,
      "supports_city": true
    }
  ]
}
```

Locales

Targetable locales by locale codes. To search for all locales starting with `en`:

```
curl -G \
  -d 'type=adlocale' \
  -d 'q=en' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
{
      "data": [
        {
          "key": 51,
          "name": "English (Upside Down)"
        },
        {
          "key": 6,
          "name": "English (US)"
        },
        {
          "key": 24,
          "name": "English (UK)"
        }
      ]
    }
}
```

Name	Description
`q` type: string	String to autocomplete values. To get all locales, leave this blank, `q=`, and set the limit to a large number `limit=1000`

DMA Codes

To get these, specify `type=adgeolocation` and `location_types=['geo_market']` in your query. To search for DMA codes that start with "New":

```
curl -G \
  -d 'location_types=["geo_market"]' \
  -d 'type=adgeolocation' \
  -d 'q=New' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The result:

```
{
  "data": [
    {
      "key": "DMA:622",
      "name": "New Orleans",
      "type": "geo_market",
      "country_code": "US",
      "country_name": "United States",
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "DMA:501",
      "name": "New York",
      "type": "geo_market",
      "country_code": "US",
      "country_name": "United States",
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "DMA:533",
      "name": "Hartford &amp; New Haven",
      "type": "geo_market",
      "country_code": "US",
      "country_name": "United States",
      "supports_region": true,
      "supports_city": true
    },
    {
....
    }
  ]
}
```

Electoral Districts

To search for Electoral Districts to target, specify `type=adgeolocation` and `location_types=['electoral_district']`. To search for Electoral Districts in California:

```
curl -G \
  -d 'location_types=["electoral_district"]' \
  -d 'type=adgeolocation' \
  -d 'q=California' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "key": "US:CA14",
      "name": "California's 14th District",
      "type": "electoral_district",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "supports_region": true,
      "supports_city": true
    },
    {
      "key": "US:CA02",
      "name": "California's 2nd District",
      "type": "electoral_district",
      "country_code": "US",
      "country_name": "United States",
      "region": "California",
      "region_id": 3847,
      "supports_region": true,
      "supports_city": true
    },
 ...
}
```

Geo Locations Metadata

You can use additional optional parameters with `type=adgeolocationmeta`:

```
curl -G \
  -d 'cities=[2418779]' \
  -d 'zips=["US:90210"]' \
  -d 'countries=["US","JP"]' \
  -d 'regions=[10]' \
  -d 'type=adgeolocationmeta' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response is a JSON object with metadata for geo locations specified:

```
{
  "data": {
    "countries": {
      "US": {
        "key": "US",
        "type": "country",
        "name": "United States",
        "supports_city": true,
        "supports_region": true
      },
      "JP": {
        "key": "JP",
        "type": "country",
        "name": "Japan",
        "supports_city": true,
        "supports_region": true
      }
    },
    "regions": {
      "10": {
        "key": "10",
        "type": "region",
        "name": "Dubai",
        "country_code": "AE",
        "supports_city": true,
        "supports_region": false
      }
    },
    "cities": {
      "2418779": {
        "key": "2418779",
        "type": "city",
        "name": "Danville",
        "region_id": 3847,
        "region": "California",
        "country_code": "US",
        "supports_city": true,
        "supports_region": true
      }
    },
    "zips": {
      "US:90210": {
        "key": "US:90210",
        "type": "zip",
        "name": "90210",
        "primary_city": "Beverly Hills",
        "region_id": 3847,
        "region": "California",
        "country_code": "US",
        "supports_city": true,
        "supports_region": true
      }
    }
  }
}
```

Options:

Name	Description
`countries` type: string	Array of country codes
`regions` type: integer	Array of region codes
`country_groups` type: string	Array of country group codes
`cities` type: integer	Array of city keys
`zips` type: string	Array of full zip codes. For example `US:92103`

Radius Suggestions

To target around a specific location, get a suggested radius reach enough people with `suggested_radius`:

```
curl -G \
  -d 'latitude=37.449478' \
  -d 'longitude=-122.173016' \
  -d 'type=adradiussuggestion' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response is JSON object with `suggested_radius` and `distance_unit`.

```
{
  "data": [
    {
      "suggested_radius": 10,
      "distance_unit": "mile"
    }
  ]
}
```

Example fetching `suggested_radius` with a `distance_unit` specified:

```
curl -G \
  -d 'latitude=37.449478' \
  -d 'longitude=-122.173016' \
  -d 'type=adradiussuggestion' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

```
{
  "data": [
    {
      "suggested_radius": 16,
      "distance_unit": "kilometer"
    }
  ]
}
```

Use these parameters:

Name	Description
`latitude` type: float	**Required.** Latitude of the location
`longitude` type: float	**Required.** Longitude of the location
`distance_unit` type: string	**Optional.** Unit of measurement, `mile` or `kilometer`

See also Local Awareness Ads to use with suggestions.

Interests

Send a `GET` request to the `/search` endpoint and set `type` to `adinterest` and `q` to the specific interest to search:

```
curl -G \
  -d 'type=adinterest' \
  -d 'q=baseball' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response will return the following fields:

Name	Description
`id` *integer*	Facebook ID of interest targeting
`locale` *string*	If available, retrieve content in language of a particular locale in the format `language_TERRITORY`. Default `en_US`

Name	Description
`name` *string*	Name of interest
`path` *array of strings*	Includes category and any parent categories for targeting

Interest Suggestions

Send a `GET` request to the `/search` endpoint and set `type` to `adinterestsuggestion` to get a list of suggested interests related to your interest.

Sample Query

```
curl -G \
  -d 'interest_list=["Basketball"]' \
  -d 'type=adinterestsuggestion' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/<API_VERSION>/search
```

Sample Response

```
{
  "data": [
    {
      "id": "6003598240487",
      "name": "la biblia",
      "audience_size": 7419780,
      "path": [
  ],
      "description": null
   },
   {
      "id": "6003022269556",
      "name": "Rugby football",
      "audience_size": 13214830,
      "path": [
  ],
      "description": null
   },
   {
      "id": "6003146664949",
      "name": "Netball",
      "audience_size": 4333770,
      "path": [
  ],
"description": null
   },
   {
      "id": "6003013291881",
      "name": "Kaizer Chiefs F.C.",
      "audience_size": 1812850,
      "path": [
  ],
      "description": null
  },
  ....
  {
      "id": "6003400886535",
      "name": "espn sportscenter",
      "audience_size": 222960,
      "path": [
  ],
     "description": null
  },
  {
     "id": "6002925969459",
     "name": "watching movies",
     "audience_size": 4630950,
     "path": [
  ],
     "description": null
  },
  {
     "id": "6003214125247",
     "name": "lakers",
     "audience_size": 340360,
     "path": [
  ],
     "description": null
  }
```

Options include:

Name	Description
`interest_list` type: array of strings	**Required.** List of terms you want suggestions for. Case sensitive.

Limitations

- Not all available interests will be returned in a search.
- Interests may be renamed at any time, and validating by name may fail when this happens. Therefore, we recommend validating interests by `interest_fbid_list` rather than by name. Check if terms are valid, by quering with `type=adinterestvalid` and the interest to validate:

```
curl -G \
  -d 'interest_list=["Japan","nonexistantkeyword"]' \
  -d 'type=adinterestvalid' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "name": "Japan",
      "valid": true,
      "id": 6003700426513,
      "audience_size": 68310258
    },
    {
      "name": "nonexistantkeyword",
      "valid": false
    }
  ]
}
```

Options:

Name	Description
`interest_list` type: array of strings	**Required, if there is no `interest_fbid_list`.** List of terms to validate. Case sensitive.

Name	Description
interest_fbid_list type: array of IDs	**Required, if there is no interest_list.** List of IDs to validated.

Interests

To browse possible interests to target, send a GET request to the /search endpoint with type set to adTargetingCategory and class to interests.

```
curl -G \
  -d 'type=adTargetingCategory' \
  -d 'class=interests' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

Limitations

- Not all available interests will be returned in a search.

Behaviors

Target based on a user's actions or past purchase behavior. Retrieve all possible behavior targeting options with type=adTargetingCategory&class=behaviors.

```
curl -G \
  -d 'type=adTargetingCategory' \
  -d 'class=behaviors' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response contains the following:

Name	Description
name type: string	Name of behavior targeting
id type: integer	Facebook ID of behavior targeting
audience_size_lower_bound *integer*	Estimated lower bound target audience size
audience_size_upper_bound *integer*	Estimated upper bound target audience size
path type: array of strings	Category and any parent categories for this targeting

Name	Description
`description` type: string	Describes target audience
`type` type: string	Targeting category class

Demographics

This includes workplace, education, job title types and relationship status types. You can also target based on recency of a life event: 3 months, 6 months, and 1 year. You can reference schools to target by an id and name.

To search for all schools starting with `ha`:

```
curl -G \
  -d 'type=adeducationschool' \
  -d 'q=ha' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
 {
  "data": [
    {
      "name": "Harvard University",
      "id": 105930651606,
      "coverage": 8395398,
      "subtext": "Cambridge, Massachusetts"
    },
    {
      "name": "Hajvery University",
      "id": 148971135122588,
      "coverage": 124162
    },
    {
      "name": "Harvard-Westlake School",
      "id": 107799365910274,
      "coverage": 14106
    }
  ]
}
```

Education Majors

Target majors by an id and name. To search for all majors that start with `ph`:

```
curl -G \
  -d 'type=adeducationmajor' \
  -d 'q=ph' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "name": "Photography",
      "id": 108170975877442,
      "coverage": 613618
    },
    {
      "name": "Physics",
      "id": 109279729089828,
      "coverage": 942491
    },
    {
      "name": "Philosophy",
      "id": 108026662559095,
      "coverage": 701271
    }
  ]
}
```

Work Employer

Reference targetable employers by id and name. To search for all work employer starting with `mic`:

```
curl -G \
  -d 'type=adworkemployer' \
  -d 'q=mic' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
  "data": [
    {
      "name": "Western Michigan University",
      "id": 10022826163,
      "coverage": 24366
    },
    {
      "name": "University of Michigan",
      "id": 21105780752,
      "coverage": 17357
    },
    {
      "name": "Michigan State University - SPARTANS",
      "id": 8891783019,
      "coverage": 65853
    }
  ]
}
```

Job Title

Every self-declared, targetable job title has an id and name. To get all job titles that include `Business Analyst`:

```
curl -G \
  -d 'type=adworkposition' \
  -d 'q=Business Analyst' \
  -d 'access_token=<ACCESS_TOKEN>' \
  https://graph.facebook.com/v<API_VERSION>/search
```

The response:

```
{
   "data": [
    {
      "name": "Business Analyst",
      "id": 105763692790962,
      "coverage": 282124
    },
    {
      "name": "Financial Analyst",
      "id": 112930925387573,
      "coverage": 212889
    }
  ]
}
```

The response has these fields:

Name	Description
`name` type: string	Name of demographic targeting
`id` type: integer	Facebook ID of demographic targeting
`coverage` type: int	Estimated target audience size
`subtext` type: string	Description for target audience

The following are common parameters for this API. For type-specific input parameters, see the details below.

Parameter Name	Description
`q`	**Required for most search types.** String to autocomplete values.
`type`	**Required.** Type of autocomplete data to retrieve. See below
`list`	**Optional.** Retrieve preferred Facebook global ID's instead of FIPS codes. Supported for `adzipcode` When used, value must equal `GLOBAL`
`limit`	**Optional.** Maximum results to return, defaults 8

Based on the category of autocomplete data, provide the appropriate `type`. To retrieve locales, specify `type=adlocale`. Valid categories are:

Value for the `type` parameter	Description
adeducationschool	Autocomplete college targeting
adeducationmajor	Autocomplete college major targeting
adgeolocation	Autocomplete combined for country, city, state & zip
adgeolocation.adcountry	Autocomplete for country
adgeolocation.adzipcode	Autocomplete for zip code
adgeolocation.adgeolocationmeta	Additional metadata for geolocations
adgeolocation.adradiussuggestion	Returns recommended radius around location
adinterest	Autocomplete locale targeting
adinterest.adinterestsuggestion	Suggestions based on interest targeting
adinterest.adinterestvalid	Validates string as valid interest targeting option
adlocale	Autocomplete locale targeting
adTargetingCategory	Parameter `q` ignored. See all possible targeting options for class with parameter `class`. Possible values of `class`: `interests`, `behaviors`, `demographics`, `life_events`, `industries`, `income`, `family_statuses`, `user_device`, `user_os`
adworkemployer	Autocomplete values for work employer
adworkposition	Autocomplete values for job title

Demographic Browse

Retrieve all possible demographic targeting options with `type=adTargetingCategory` and a `class`.

Name	Description
`class` type: string	Specify one: `life_events`, `industries`, `income`, `family_statuses`, `user_device`. Specifying `demographics` retrieves all. **Demographic targeting options are not available in all countries.** Facebook may return different results, including empty results, depending on the home country setting of the user whose access token is being used to make this API call.

The response contains these fields:

Name	Description
`name` type: string	Name of the demographic targeting

Name	Description
`id` type: integer	Facebook ID of the demographic targeting
`audience_size_lower_bound` *integer*	Estimated lower bound target audience size
`audience_size_upper_bound` *integer*	Estimated upper bound target audience size
`description` type: string	Description of the target audience
`type` type: string	Type of demographic. Useful if you retrieve all demographics.
`path` type: array of strings	Includes the category and any parent categories the targeting falls into

Subscribe to Changes

Marketing API

- Overview
- Get Started
- Ad Creative
- Bidding
- Ad Rules Engine
- **Audiences**
  - Overview
  - Customer File Custom Audiences
  - Audience Rules
  - Lookalike Audiences
  - Value-Based Lookalikes
  - Engagement Custom Audiences
  - Mobile App Custom Audiences
  - Website Custom Audiences
  - Offline Custom Audiences
  - Dynamic Audiences
  - Audience Network Ads
  - Reach Estimate API
  - Special Ad Category
  - **Reference**
    - Basic Targeting
    - Advanced Targeting
    - Placement Targeting

- Targeting Search
- Detailed Targeting
- Targeting Description
- Advantage Targeting
- Flexible Targeting
- Deprecated Targeting Terms
- Targeting Restrictions
- Estimated Daily Results
- Custom Audience Terms Of Service

Insights API

Brand Safety and Suitability

Best Practices

Troubleshooting

API Reference

Changelog