Geocoder Autocomplete API

Developer's Guide

Version 6.2.173

Important Information
Notices

Topics:

This section contains document notices.

Legal Notices

© 2019 HERE Global B.V. and its Affiliate(s). All rights reserved.

This material, including documentation and any related computer programs, is protected by copyright controlled by HERE. All rights are reserved. Copying, including reproducing, storing, adapting or translating, any or all of this material requires the prior written consent of HERE. This material also contains confidential information, which may not be disclosed to others without the prior written consent of HERE.

Trademark Acknowledgements

HERE is trademark or registered trademark of HERE Global B.V.

Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

Disclaimer

This content is provided "as-is" and without warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, satisfactory quality and non-infringement. HERE does not warrant that the content is error free and HERE does not warrant or make any representations regarding the quality, correctness, accuracy, or reliability of the content. You should therefore verify any information contained in the content before acting on it.

To the furthest extent permitted by law, under no circumstances, including without limitation the negligence of HERE, shall HERE be liable for any damages, including, without limitation, direct, special, indirect, punitive, consequential, exemplary and/ or incidental damages that result from the use or application of this content, even if HERE or an authorized representative has been advised of the possibility of such damages.

Document Information

Product	
Name:	Geocoder Autocomplete API
Version:	Version 6.2.173
Document	
Name:	Geocoder Autocomplete API Developer's Guide
ID:	1172d5b1-e018-4775-9af6-9d4b6b1fbb6f
Status:	FINAL
Date:	2019-Feb-04, 9:06 (GMT)

Contents

Chapter 1
Introduction

Topics:

This document introduces the Geocoder Autocomplete API and:

- explains key concepts
- provides examples
- documents resources and query parameters
- documents response structures and data types

What Is the Geocoder Autocomplete API?

The HERE Geocoder Autocomplete API is a REST API that you can integrate into web applications to help users obtain better results for address searches with fewer keystrokes. Spatial and region filters can be used to return suggestions with greater relevance to end users, such as results that are within a specified country or in the proximity of the current location.

The Geocoder Autocomplete API retrieves a complete address and an ID. You can subsequently use the Geocoder API to geocode the address based on the ID and thus obtain the geographic coordinates of the address.

Why Use the Geocoder Autocomplete API?

The table below lists the high level use cases for the Geocoder Autocomplete API.

Table 1: Main features of the HERE Geocoder Autocomplete API

Feature	Description
Reduce the number of keystrokes	The end user starts typing and after the first character, typically, the system provides a list of suggestions. The end user sees the address he or she is looking for in this list and selects one to retrieve its location.
Assist the end user with the spelling of addresses	The end user receives immediate feedback relative to their input in the form of suggestions that he or she can use to complete or correct their input quickly.
Assist the end user with the context of addresses	The end user receives assistance in completing an address request based on the context. • For example, "520 Hayes" can be completed to "520 Hayes St". The 520 suggests an address context. But "Hayes" alone could also mean the area "Hayes Valley". • "Praca Falcao" suggests a complete address with all its details and can be completed to "Praça General Gentil Falcão, Sao Paulo".

Chapter 2
Quick Start

Topics:

This section demonstrates how to start using the Geocoder Autocomplete API by showing how to get a list of suggestions for an incomplete search text.

Requesting Autocomplete Suggestions

To request a list of address suggestions for the search text `Pariser 1 Berl`, send the following GET request:

```
http://autocomplete.geocoder.api.here.com/6.2/suggest.json
    ?app_id={YOUR_APP_ID}
    &app_code={YOUR_APP_CODE}
    &query=Pariser+1+Berl
    &beginHighlight=<b>
    &endHighlight=</b>
```

The response to the above request includes the following high level elements:

- List of suggested matches to the search text with the label and individual address elements containing highlights showing matches for the search text.

```
{
    "suggestions": [
        {
            "label": "Deutschland, Berlin, Berlin, 10117, Berlin, Pariser Platz 1",
            "language": "de",
            "countryCode": "DEU",
            "locationId": "NT_5mGkj3z90Fbj4abzMbUE4C_xA",
            "address": {
                "country": "Deutschland",
                "state": "Berlin",
                "county": "Berlin",
                "city": "Berlin",
                "district": "Mitte",
                "street": "Pariser Platz",
                "houseNumber": "1",
                "postalCode": "10117"
            },
            "matchLevel": "houseNumber"
        },
        {
            "label": "Deutschland, Berlin, Berlin, 12623, Berlin, Pariser Straße",
            …
        },
        …
    ]
}
```

Related Information

- *Constructing a Request* on page 11
- *Examples* on page 15
- *Request Parameters* on page 19
- *Response Structure* on page 21

Chapter 3
Guide

Topics:

The articles in this section provide a guide to using the Geocoder Autocomplete API. The guide describes common use cases for the API and illustrates them with practical examples.

Key Concepts

This section provides detailed information on the features of theGeocoder Autocomplete API.

The Geocoder Autocomplete API allows you to request autocomplete suggestions based on submitted search terms and supports the following features:

- no requirement for a minimum number of characters – suggestions are available based on only a single letter
- world-wide indexing of the following elements for quick searches:
  - street names, house numbers
  - areas (any combination of district, city, county, state, country name and country code)
  - postal codes
- response language selection based on relevance to query (for example, addresses in francophone countries)
- street types matched to their abbreviated or spelled out forms
- recognition of special characters like German umlauts, French acute accent or acute grave
- spatial and region filters to return suggestions with greater relevance to end users
- simple JSON responses
- formatted labels for localized address formatting (for example, house no. first or street name first) and structured address fields in customizable order
- support for highlighting a match

Acquiring Credentials

All users of HERE APIs must obtain authentication and authorization credentials and provide them as values for the parameters `app_id` and `app_code`. The credentials are assigned per application.

This document uses the placeholder text `{YOUR_APP_ID}` and `{YOUR_APP_CODE}` as placeholders for access and authorization credentials. Replace these placeholders with your own unique application-specific credentials to access the API resources.

To obtain the credentials for an application, please visit *http://developer.here.com/plans* to register with HERE.

If you wish to explore the API, use the API Explorer at *https://developer.here.com/api-explorer*.

Constructing a Request

A typical request to the Geocoder Autocomplete API includes the basic elements shown in the following table and, in addition, it may contain resource-specific parameters or data.

Table 2: Basic request elements

Element	Value/Example	Description
Base URL	`http://autocomplete.geocoder.api.here.com`	Production environment
Path	`/6.2/`	
Resource	*suggest*	GET only, specify request details via query parameters
Application Code	`&app_code={YOUR_APP_CODE}`	Substitute your own unique `app_code`
Application Id	`&app_id={YOUR_APP_ID}`	Substitute your own unique `app_id`

This example search request uses HTTP GET to request autocomplete suggestions for the search term `Berli`:

```
http://autocomplete.geocoder.api.here.com/6.2/suggest.json
?query=Berli
&app_id={YOUR_APP_ID}
&app_code={YOUR_APP_CODE}
```

HERE Server Environments

HERE provides two server environments for handling your requests: a Production environment and a Customer Integration Testing (CIT) environment.

The table below summarizes when to use these environments.

Environment	Functional/Integration Testing	Load/Performance Testing	General Production Use
Production	yes	no	yes
Customer Integration Testing (CIT)	yes	no	no

Production

Use the Production environment for general production use.

Customer Integration Testing (CIT)

Use the CIT Environment for low-volume integration and functional testing. To access CIT for REST APIs, amend the base URL to include an additional `cit` segment. For example, the CIT URL for this API is `http://autocomplete.geocoder.cit.api.here.com`.

Both the Production and CIT environments are not designed to support sudden, unannounced high loads. If you need to do performance testing, *contact HERE* to discuss options.

Using Autocomplete

The figures below show various search texts entered by the end user and the list of suggestions in the Geocoder Autocomplete API response. The highlights mark the parts of the suggestions that match the search text. The highlighting shown here is based on the mark-up provided with each request for suggestions (in the

`beginHighlight` and `endHighlight` request parameters). Client applications can implement their own highlighting, modify the mark-up, or simply ignore it.

Requesting Suggestions without Spatial Filtering

Figure 1: With a Partial Name

San Francisco Hay
United States, CA, Hayward, San Francisco Bay United States, CA, San Francisco, San Francisco, Hayes Valley United States, CA, San Francisco Bay Area, North Hayward United States, CA, San Francisco, San Francisco, Hayes St

San Francisco Hayes
United States, CA, San Francisco, San Francisco, Hayes Valley United States, CA, San Francisco, San Francisco, Civic Center, Hayes St United States, CA, San Francisco, San Francisco, Hayes Valley, Hayes St United States, CA, San Francisco, San Francisco, Northern Park, Hayes St United States, CA, San Francisco, San Francisco, Panhandle, Hayes St

Figure 2: With a Street Name and a Partial District Name

Invalidenstraße M
Deutschland, Berlin, Mitte, Invalidenstraße Deutschland, Berlin, Moabit, Invalidenstraße Deutschland, Mecklenburg-Vorpommern, Parchim, Invalidenstraße Deutschland, Schmalkalden-Meiningen, Walldorf, Invalidenstraße

Figure 3: With a Preposition Omitted in the Street Name

Calle Doctor Juan Pad
España, Las Palmas de Gran Canaria, Calle Doctor del Juan de Padilla España, Igualada, Calle del Doctor Juan Padró

Figure 4: With Special Characters: è vs e ...

Bagnere
France, Bagnères-de-Luchon France, Bagnères-de-Bigorre France, Barie, Bagnères France, Saint-Côme, Bagnères France, Gabarret, Bagnères

Figure 5: With Special Characters: ß vs ss ...

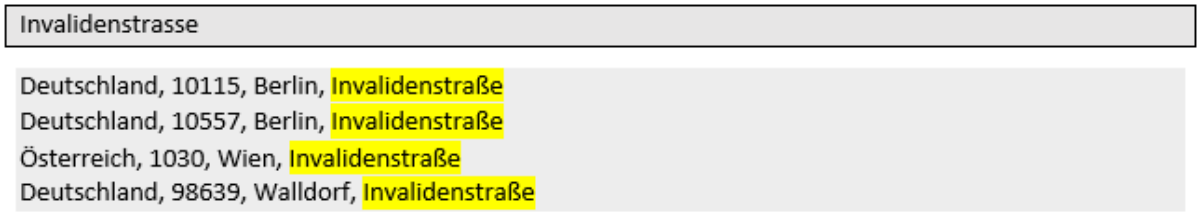
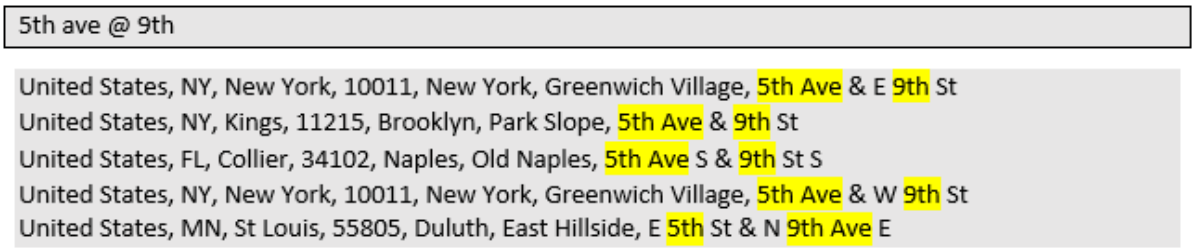
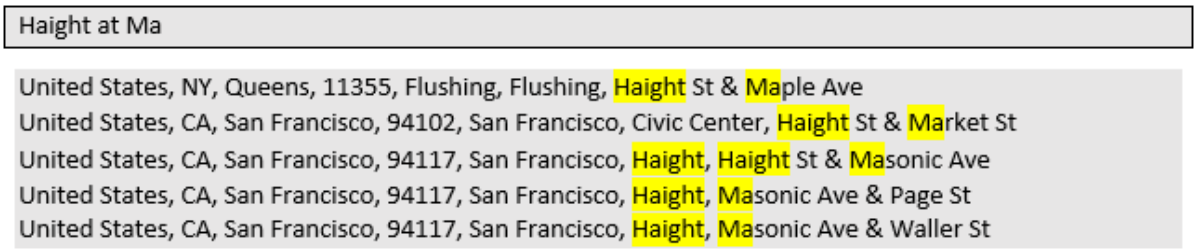


Figure 6: With an Intersection Query: (`query=5th ave @ 9th`)



The service returns intersections for queries containing a street name followed by `and`, `at`, `&` or `@`. Currently, intersection queries are supported for queries on USA only.

Figure 7: With an Intersection Query Using "at" as a Separator: (`query=Haight at Ma`)



Requesting Suggestions with Spatial Filtering

You can use various types of spatial filtering to specify a context for the suggestions.

The figure below illustrates a request made without the country filter.

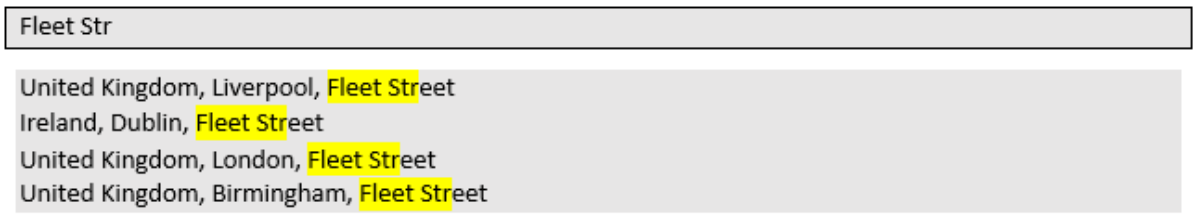
Figure 8: Without a Country Filter

The figure below illustrates the same request, but this time with the request parameter `country` set to France (`country=FRA`).

Figure 9: With a Country Filter

The figure below illustrates a request made without the mapview filter.

Figure 10: Without a Mapview Filter



The figure below illustrates the same request, but this time with the mapview parameter set to Montréal, Québec(`mapview=46.0346628,-74.3384855;45.2173875,-73.043472`).

Figure 11: With a Mapview Filter

Related Information

- *Constructing a Request*
- *Examples*

Examples

This section walks through a complete usage scenario starting from issuing a query with the Geocoder Autocomplete API and ending with the retrieval of the geocoded location information from the Geocoder API.

Retrieve Location Details for a Suggestion

After the user types a few characters, request a list of suggestions from the Geocoder Autocomplete API. Show the suggestions to the user. After the user selects a suggestion, retrieve the detailed location information corresponding to the suggestion using the Geocoder API.

Request

Request a list of address suggestions for the search text `Pariser 1 Berl`:

```
http://autocomplete.geocoder.api.here.com/6.2/suggest.json
    ?app_id={YOUR_APP_ID}
    &app_code={YOUR_APP_CODE}
    &query=Pariser+1+Berl
    &beginHighlight=<b>
    &endHighlight=</b>
```

Response

The response from the Geocoder Autocomplete API contains a number of suggestions. Each of them represents a location defined through a set of attributes, but these do not provide detailed location information such as geographic coordinates.

```
{
   "suggestions": [
      {
        "label": "Deutschland, Berlin, Berlin, 10117, Berlin, Pariser Platz 1",
        "language": "de",
        "countryCode": "DEU",
        "locationId": "NT_5mGkj3z90Fbj4abzMbUE4C_xA",
        "address": { ... },
        "matchLevel": "houseNumber"},
      ... ]
}
```

Geocoder Request to retrieve the Location Details

After the user has selected a suggestion, you can retrieve the location details such as the geographic coordinates via the HERE Geocoder API, using look-up by `locationId`.

Here is the request to the Geocoder API. For more information about the HERE Geocoder API, see the *Geocoder API Developer's Guide.*

```
http://geocoder.api.here.com/6.2/geocode.json
?locationid=NT_5mGkj3z90Fbj4abzMbUE4C_xA
&jsonattributes=1
&gen=9
&app_id={YOUR_APP_ID}
&app_code={YOUR_APP_CODE}
```

Geocoder Response

```
{
  "response":
```

```
...
      "view": [{
         "result": [{
            "matchLevel": "houseNumber",
            "matchType": "pointAddress",
            "location": {
               "locationId": "NT_5mGkj3z90Fbj4abzMbUE4C_xA",
               "locationType": "point",
               "displayPosition": {
                  "latitude": 52.51588,
                  "longitude": 13.37804 },
               "navigationPosition": [{
                  "latitude": 52.51591,
                  "longitude": 13.37833 }],
            "mapView": {
               "topLeft": {
                  "latitude": 52.5248732,
                  "longitude": 13.3632617 },
               "bottomRight": {
                  "latitude": 52.5068868,
                  "longitude": 13.3928183 }},
            "address": {
               "label": "Pariser Platz 1, 10117 Berlin, Deutschland",
               "country": "DEU",
               "state": "Berlin",
               "county": "Berlin",
               "city": "Berlin",
               "district": "Mitte",
               "street": "Pariser Platz",
               "houseNumber": "1",
               "postalCode": "10117",
               "additionalData": [{
                  "value": "Deutschland",
                  "key": "CountryName" },
                  {
                  "value": "Berlin",
                  "key": "StateName" },
                  {
                  "value": "Berlin",
                  "key": "CountyName" }]}}],
         "viewId": 0 }]
   }
}
```


Related Information

- *Geocoder Developer Guide*
- *Request Parameters* on page 19
- *Response Structure* on page 21

Service Support

If you need assistance with this or other HERE products, contact your HERE representative or Technical Customer Support.

Chapter 4
API Reference

Topics:

This part of the guide provides a comprehensive reference to Geocoder Autocomplete API.

Request Parameters

The Geocoder Autocomplete API is very simple. It supports a small number of request parameters. Among them, only the authorization parameters (`app_id`, `app_code`) and `query` are mandatory.

Requests must conform to this general pattern:

```
.../6.2/suggest.json?<parameter>=<value>...
```

The table below documents the supported request parameters.

Table 3: Request Parameters

Parameter	Type	Description	Examples
query	`string`	The search text which is the basis of the query.	`query=Pariser Pl`
maxresults	`integer`	The upper limit the for number of suggestions to be included in the response. Default is set to 5. Valid range: 1 to 20.	`query=Pariser Pl` `&maxresults=10`

country	A comma separated list of strings (3-letter ISO country codes)	A type of Spatial Filter. The spatial filter limits the search for any other attributes in the request. The country parameter limits suggestions to a country or set of countries.
		country=`ISO3 country code`
		Can be combined with the `mapview` or `prox` spatial filters.
			`query=Rue du Froma`
			`&country=FRA`
			`query=Rue du Froma`
			`&country=FRA,BEL`

mapview	two lat/lon pairs	A type of Spatial Filter. Sets a focus on a geographic area represented by the top-left and the bottom-right corners of a bounding box so the results within this area are more important than results outside of this area.
		`mapview=`
		`TopLeft.Latitude,`
		`TopLeft.Longitude;`
		`BottomRight.Latitude,`
		`BottomRight.Longitude`
		Can be combined with the `country` spatial filter.
			`mapview=41.90852,-87.67629;`
			`41.86827,-87.60419`

prox	lat/lon pair and (optional) radius	A type of Spatial Filter. Sets a focus on a geographic area represented by a single geo-coordinate pair and optionally a radius (in meters) so the results within this area are more important than results outside of this area.
		`prox=Latitude,Longitude`
		`prox=Latitude,Longitude,Radius`
		Can be combined with the country spatial filter.
			`prox=37.86946,-122.26811,10000`

beginHighlight	`string`	Mark the beginning of the match in a token. This can be any character sequence. Common usage is a HTML tag such as <b> for bold. But it can also be a square bracket, "[".
		Default: no marker
			`query=Barcelo`
			`&beginHighlight=<b>`
			`&endHighlight=</b>`
			Result:
			`"label": "Espanya, Catalunya,`
			`<b>Barcelo</b>na"`

Parameter	Type	Description	Examples
endHighlight	`string`	Mark the end of the match in a token.	See *beginHighlight*

language	`string`	The preferred language of address elements in the result.
		The language parameter must be provided as 2-letter ISO language code. The plural form of the parameter (languages) is not supported and ignored. Only one language can be provided. The language setting changes the language of result elements where available in the map data. The language setting has no impact on matching or ranking and it does not express any regional preference.
		The default response language is the language that is most relevant to the query.
			`language=it&query=champs elys`
			Result:
			`"label": "Francia, Pariggi, Avenue des Champs-Élysées"`

resultType	areas	The `resultType=areas` mode will filter results to return areas. Results with matchLevels of `houseNumber`, `postalCode`, `street` and `intersection` will not be returned.	`resultType=areas&query=High`

Response Structure

The Geocoder Autocomplete API response is in JSON format.

The table below documents the elements of the response.

Table 4: Response Elements

Element	Description	Example
suggestions	Top-most element holding list of suggestions. The element is absent in case of an error.	`suggestions: [ .... ]`
label	Formatted response label including support for match highlights.	`label: "Deutschland, [Berli]n, [Berli]n, 10117, [Berli]n, [Pariser] Platz [1]"`
language	Language of the match. 2-letter ISO language code.	`language: "de"`
countryCode	Country of the location. 3-letter ISO country code.	`countryCode: "DEU"`
locationId	Client applications use the `locationId` for location lookup.	`locationId: "NT_5mGkj3z90Fbj4abzMbUE4C_xA"`

address	Structured address block. The address elements support match highlighting.	`address: { country: "Deutschland", state: "Berlin", county: "Berlin", city: "Berlin", district: "Mitte", street: "Pariser Platz", houseNumber: "1", postalCode: "10117" }`
		For addresses from Australia, New Zealand and Canada a unit number can be included in the result if it was part of the query.
		`address: { country: "New Zealand", state: "Canterbury", county: "Christchurch City", city: "Christchurch", district: "Edgeware", street: "Allard St", houseNumber: "9", unit: "3", postalCode: "8013" }`

distance	Distance in meters from the search center. Only available for queries with `mapview` or `prox` parameter.	`suggestions [{ label: "Deutschland, Berlin, Pariser Platz", ... address: { country: "Deutschland", state: "Berlin", county: "Berlin", city: "Berlin", district: "Mitte", street: "Pariser Platz", houseNumber: "1", postalCode: "10117" }, distance: 666, matchLevel: "street" }`
matchLevel	Match precision. One of: `houseNumber`, `intersection`, `street`, `postalCode`, `district`, `city`, `county`, `state`, `country`	`matchLevel: "houseNumber"`

Errors

In case of an error, Geocoder Autocomplete API returns an empty response. Client applications can check if the `suggestions` element exists, to identify a successful request.

Authentication errors are indicated using HTTP status codes:

HTTP Status Code	Status	Description
400	Bad Request	Missing `app_id` or `app_code` parameter
401	Unauthorized	Validation of `app_id`/`app_code` pair failed. Either `app_code` does not match the `app_id` or the `app_id` is not entitled to use the Geocoder Autocomplete API.