Date :  01-03-2024

reverse geocoding and geolocation service

version: 1.0.0

# **Overview**

Find nearest or largest city information or time zone for any given point location (latitude/longitude). The reverse geocodes API back “reverse” codes any GPS location point into nearest or largest readable places with county, city, distance, and population. Provides distance information between two locations. Provides additional “Is-On-Water” check for any location.

Returns nearest or largest city and/or time zone info for any latitude/longitude geo coordinate within a defined range: City Name, Population, Distance, bearing (Degrees), Compass Direction (16-wind compass rose), Country, Time zone, GMT Offset. In addition, it will provide distance information in meters, kilometers and miles for any two given locations with bearing, compass and country. A check if a location is “on-water” or not is also available as check against sea and lake geo information.

# **Getting Started**

## **Prerequisites**

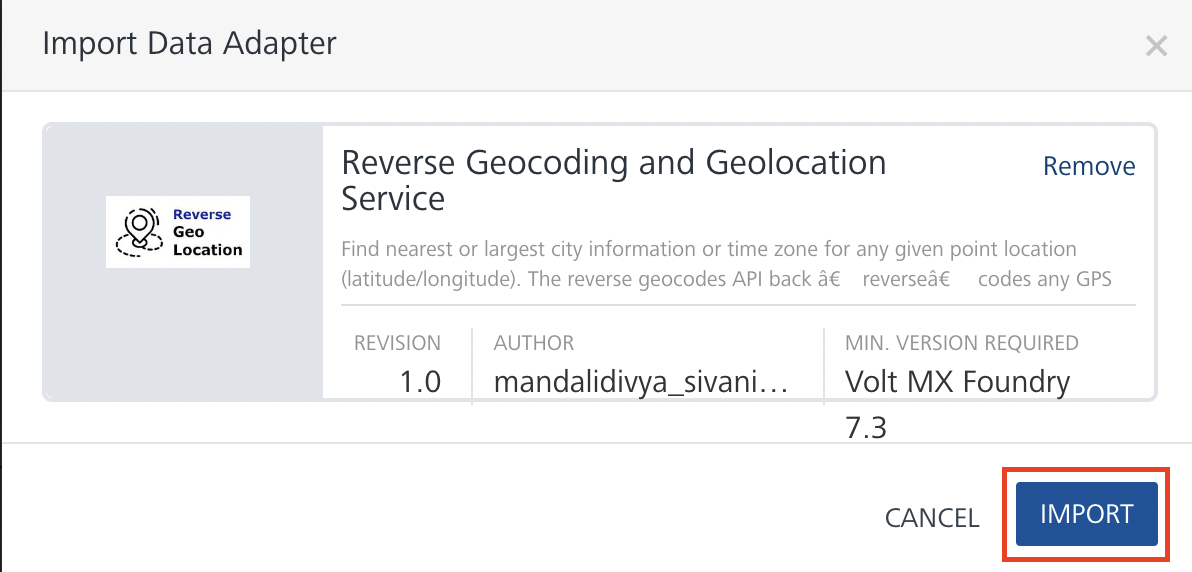
* Volt Foundry
* [**Rapid API Account**](https://rapidapi.com/Noggle/api/reverse-geocoding-and-geolocation-service)
* X-RapidAPI-Key, X-RapidAPI-Host.

## **Importing the adapter**

**To import the Data Adapter to Volt Foundry, do the following:**

1. Sign in to the  [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Custom Data Adapters**.  
   
4. Click **IMPORT** to import a custom data adapter.  
   
5. On the Import Data Adapter dialog box, click browser to import.  
   
6. Select **Reverse Geocoding and Geolocation Service** **zip** file and click **IMPORT**.

After you import the data adapter, Volt Foundry opens a window that shows the metadata of the data adapter.



After you import the data adapter, you can view it on the Custom Data Adapters page and use it to create services on Volt Foundry.

A screenshot of a computer

Description automatically generated

## **Creating an Integration service**

After you import the data adapter into Volt Foundry, you can use it to create an Integration Service.

Follow the given steps to create an Integration service using the **Reverse Geocoding and Geolocation Service** Adapter.

1. Sign in to the [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Integration**.

A screenshot of a computer

Description automatically generated

1. To create a new service, click the **+** button or the **CONFIGURE NEW** button.

A screenshot of a computer

Description automatically generated

1. On the Service Definition tab, select the service type as**Reverse Geocoding and Geolocation Service,** and click **SAVE**.

A screenshot of a computer

Description automatically generated

Alternatively, you can also create a Foundry app and create an Integration service inside it.

## **Creating and Executing operations**

After you create an integration service, you can create and execute operations using the service.

#### **Creating an Operation**

* In **API Management/Foundry app you created**, in the **Integration** section, select the service that you created.
* After you select the service, navigate to the **Operation List** tab.  
  
* From the drop-down list, select an operation that you want to execute, and click **ADD OPERATION**.

A screenshot of a computer

Description automatically generated

#### **Executing an Operation**

* From the **Operations List** tab, in the **Configured Operations** section, select the operation you want to execute.

A screenshot of a computer

Description automatically generated

**Note:**You can follow the same steps to create operations for other methods but ensure that you select the required API.

You can create multiple operations with required APIs and save them in the **Configured Operations** section. You can then execute the operations whenever required.

* On the Operation Page, in the **Request Input** tab, in the **Body** section, for the following parameters, specify the values in the respective boxes under the **Test Value** column.

**Mandatory Parameters**

* + **X-RapidAPI-Key, X-RapidAPI-Host [in header]:**  API Key associated with your RapidAPI account, required for authentication.

To get keys – Login in to [**Rapid API Account**](https://rapidapi.com/Noggle/api/reverse-geocoding-and-geolocation-service)-> You can view your keys.

* + **latitude:** latitude in decimal degrees (wgs84).
  + **longitude:** longitude in decimal degrees (wgs84).
  + **range:** max radial range for lookup in meter (0=no range), (max 100.000).

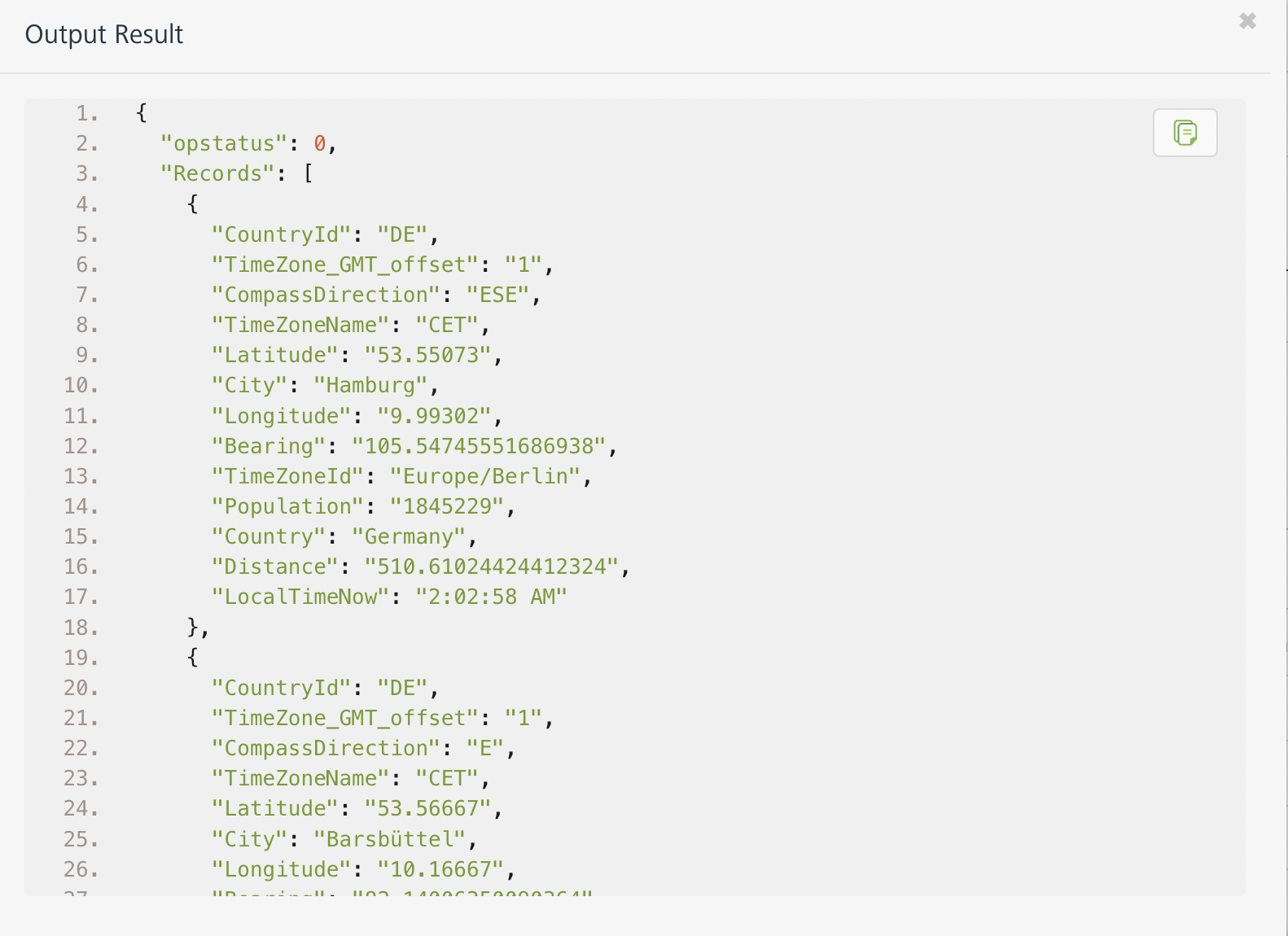
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Select a run-time environment and click **Save and Fetch Response** to get a response based on your inputs.



## **Publishing your application**

If you want to use the services in client applications, you need to publish an app to a run-time environment. You can create the service (as described above) in an application or import the service into an application and publish the application.

# **References**

## Endpoint Documentation

## **Requests Input Parameters for GetNearestCities**

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| X-RapidAPI-Key(string) | **[Required]** **[Header]**API Key associated with your account. |
| X-RapidAPI-Host (string) | **[Required]** **[Header]** The host parameter specifies the domain name or IP address of the RapidAPI endpoint. In the context of the Reverse Geocoding and Geolocation Service, the default host is set to 'geocodeapi.p.rapidapi.com'. This parameter is necessary for routing the request to the appropriate API endpoint. |
| latitude (Integer) | **[Required]** latitude in decimal degrees (wgs84). |
| longitude (Integer) | **[Required]** longitude in decimal degrees (wgs84). |
| range (Integer) | **[Required]** max radial range for lookup in meter (0=no range), (max 100.000) |

### **Response Output Parameters for GetNearestCities**

## 

|  |  |  |
| --- | --- | --- |
| **Object Property** | **Data Type** | **Description** |
| Records | Array | An array containing records of city information. |
| Records.CountryId | String | The country code for the city. |
| Records.TimeZone\_GMT\_offset | String | The GMT offset of the city's time zone. |
| Records.CompassDirection | String | The compass direction of the city. |
| Records.TimeZoneName | String | The name of the city's time zone. |
| Records.Latitude | String | The latitude coordinate of the city. |
| Records.City | String | The name of the city. |
| Records.Longitude | String | The longitude coordinate of the city. |
| Records.Bearing | String | The bearing of the city. |
| Records.TimeZoneId | String | The time zone ID of the city. |
| Records.Population | String | The population of the city. |
| Records.Country | String | The country of the city. |
| Records.Distance | String | The distance of the city from a reference point. |
| Records.LocalTimeNow | String | The local time of the city. |

**Reference Document:** [Reverse Geocoding and Geolocation Service API Documentation (Noggle) | RapidAPI](https://rapidapi.com/Noggle/api/reverse-geocoding-and-geolocation-service)

# **Revision History**

Adapter version 1.0.0:

## **Known Issues**

* No Known Issues

## **Limitations**

* No limitations