

# Get Started with Oracle Address Verification Cloud Service

Address Verification Cloud Service lets you verify and standardize addresses in your applications. For example, the service can correct city or street names that are misspelled, complete full postal codes, and standardize abbreviations like St and Pkwy. The service takes your data and returns matches—one match or multiple matches—and you can select which version you want to use as the standard.

## Topics:

- [About Address Verification Cloud Service](#)
- [How to Begin with Address Verification](#)
- [Use Address Verification REST APIs](#)

See Oracle Cloud Terminology in *Getting Started with Oracle Cloud* for definitions of terms found in this and other documents in the Oracle Cloud library.

## About Address Verification Cloud Service

Address Verification Cloud Service can be used in conjunction with DaaS (Social Data and Insight), or it can be used as a standalone service with other applications. It provides a REST-based endpoint that supports searching and verifying addresses worldwide.

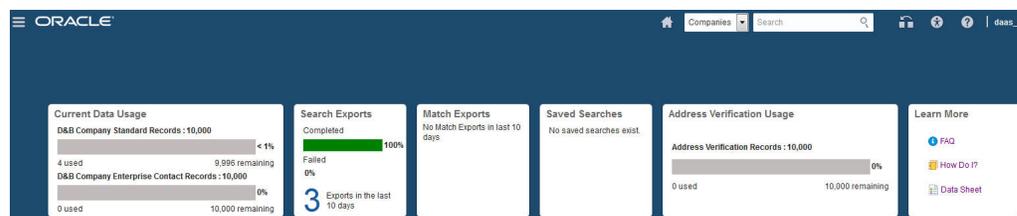


Address Verification is available with one of the following subscriptions:

- Unlimited annual subscription
- Per record annual subscription (for example, purchase a pool of 200,000 address verification transactions)
- Free monthly trial subscription with 1,000 address verification transactions

There is no charge for the search capability. Search as much as you want, and when you see a record you want to verify, then call the verify API with this record, and the usage count increases accordingly. The record usage count only increases with the VERIFY record and when the "verified" field in the response body is "Y".

There is no user interface provided, but you have access to your usage report on the DaaS interface. The following screen shot shows 0 records used (0%) on a 10,000 record subscription. For an unlimited subscription, it shows only the number of records used (and not a percentage). If you haven't purchased a subscription for DaaS, then this interface shows only the **Address Verification Usage** and the **Learn More** boxes.



## How to Begin with Address Verification

Here's a summary of the key steps to get started with Address Verification subscriptions:

1. Request a trial or purchase a subscription. See Requesting a Trial Subscription or Buying a Nonmetered Subscription to an Oracle Cloud Service in *Getting Started with Oracle Cloud*.

### Note:

Address Verification and DaaS (also known as Social Data and Insight Cloud Service) offer a combined *trial* subscription; that is, when you order a trial for one service, you also get the other. Each service must be *purchased* separately, but they can be part of the same subscription order.

2. Activate the service. See Activating Oracle Cloud Services in *Getting Started with Oracle Cloud*.

3. Log on to the My Services application to get the Address Verification API URL.

The following default account roles are created during Address Verification provisioning:

- Data Service Administrator (dataService\_administrator)
  - Data Service Client APPID (dataservice\_client\_api\_appid)
  - Data Service User (dataService\_user)
4. Create accounts for your users and assign them appropriate privileges and roles. See Managing Your Oracle Cloud Service in *Getting Started with Oracle Cloud*.

5. Optionally, enable Oracle Engagement Cloud to integrate with Address Verification for real-time address verification within Engagement Cloud. This also lets you schedule large-scale batch verifications. Integration with Engagement Cloud requires the Fusion Data Quality add-on. Assuming you have an active Engagement Cloud subscription with Fusion Data Quality, and then added a DaaS subscription (completing all steps above), just follow these four steps:



Video

- a. In the My Services application, create a new DaaS user. Expand the **Advanced Roles** section and add the **Data Service Client AppID** role for this user.
- b. In the My Services application, reset the password for this DaaS user.
- c. In Engagement Cloud, configure the **Manage Sales Cloud to Oracle Social Data and Insight Cloud Service Integration** task. To do this, from Setup and Maintenance, select the **Tasks** tab and search for **Manage Sales Cloud to Oracle Social Data and Insight Cloud Integration**. Then click the **Go to Task** button. For the URL, remove `/data/ui` from the end of the instance address listed in your Welcome email and on the My Services application; for example, `https://mydataservice-myidentitydomain.data.us2.oraclecloud.com`. Select **oracle/wss\_username\_token\_over\_ssl\_client\_policy** for **Security Policy** to connect to DaaS, and enter the user and password for the user you created in the first step.
- d. In Engagement Cloud, similarly, configure the **Manage Administrator Profile Values** task. On the Manage Administrator Profile Values page, search for the **DAAS\_PRODUCTION\_MODE** profile option code. Make sure the **Profile Level Site** value stays at No. This is the only value supported today.

See [Configure Oracle Social Data and Insight Cloud Service for Data Enrichment and Address Verification](#) in *Oracle Engagement Cloud Implementing Customer Data Management* for information on using the services together.

**Note:**

To associate an Engagement Cloud instance with both DaaS and Address Verification, both instances must be part of the same service instance; that is, both services must be part of the same subscription order. For example, if you order DaaS first and later add Address Verification, then you must upgrade your DaaS subscription to include Address Verification, or vice-versa. DaaS and Address Verification cannot be on separate service instances when associating them with the same instance of Engagement Cloud.

 **Note:**

If you have a preproduction environment for Engagement Cloud, test the service association between your Engagement Cloud preproduction environment and your DaaS trial subscription (limit 500 records to test). When you go to production, do the service association between your Engagement Cloud production environment and your DaaS production environment. The service association steps are the same, but the trial/production service URLs are different.

## Use Address Verification REST APIs

The Address Verification Cloud Service lets you search and verify addresses in your applications. Access the REST APIs in the following format: `https://serviceName-identityDomain.data.us2.oraclecloud.com/av/api`. For example: `https://datatrial1234-usoracletrial1234.data.us2.oraclecloud.com/av/api/v3/address/find`.

## Topics

- [Search Addresses](#)
- [Verify Addresses](#)
- [Use a REST Client to Make Calls to Address Verification Cloud Service](#)

## Search Addresses

The Address Verification service supports Capture+ (by GBG/Loqate) for searching valid mailing addresses worldwide.

 **Note:**

Previously, you could enter your known information into the API (such as `city` and `address1`) and specify search mode (`s`). In the response, the API returned a list of records (or a single record with `verified="None"` if nothing was found for that input). The search mode (available in v2, based on Oracle EDQ) and the auto complete method (available in v3, based on Loqate) now are in maintenance mode only.

Capture+ is available through the following RESTful endpoints:

```
/api/v2/address/find  
/api/v2/address/preparefind  
/api/v2/address/retrieve
```

```
/api/v3/address/find  
/api/v3/address/preparefind  
/api/v3/address/retrieve
```

Capture+ functions the same in v2 and in v3. The only difference is that `/address/retrieve` shows EDQ-style fields in v2, and it shows Loqate-style fields in v3.

### Search Simple Use Case

This simple use case applies whether you search via `/preparefind` or `/find`: the functionality is identical. When you enter the first three digits (such as “123”) of an address in the United States, as in the following request and response, the first `find` request is issued.

#### Request 1

```
http://adc00qrw.us.oracle.com:7767/av/api/v3/address/find?Text=123
```

#### Response 1

```
[  
  {  
    "Id": "US|US|ENG|94107-CA-SAN_FRANCISCO--ST-TOWNSEND--123",  
    "Type": "BuildingNumber",  
    "Text": "123 Townsend St",  
    "Highlight": "0-3",  
    "Description": "San Francisco CA 94107 - 320 Addresses"  
  },  
  {  
    "Id": "US|US|ENG|94105-CA-SAN_FRANCISCO--ST-MISSION--123",  
    "Type": "BuildingNumber",  
    "Text": "123 Mission St",  
    "Highlight": "0-3",  
    "Description": "San Francisco CA 94105 - 48 Addresses"  
  },  
  {  
    "Id": "US|US|A|Z222445177|123",  
    "Type": "Address",  
    "Text": "123 2nd St",  
    "Highlight": "0-3",  
    "Description": "San Francisco CA 94105"  
  },  
  {  
    "Id": "US|US|A|Z222445178|1",  
    "Type": "Address",  
    "Text": "123 2nd St Ste 1",
```

```

    "Highlight": "0-3",
    "Description": "San Francisco CA 94105"
  },
  {
    "Id": "US|US|A|Z223933823|123",
    "Type": "Address",
    "Text": "123 Mission St",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94105"
  },
  {
    "Id": "US|US|ENG|94103-CA-SAN_FRANCISCO--ST-GUERRERO--123",
    "Type": "BuildingNumber",
    "Text": "123 Guerrero St Apt",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94103 - 3 Addresses"
  },
  {
    "Id": "US|US|A|Z222437371|123",
    "Type": "Address",
    "Text": "123 Guerrero St",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94103"
  },
  {
    "Id": "US|US|A|Z222457206|123",
    "Type": "Address",
    "Text": "123 Waverly Pl",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94108"
  },
  {
    "Id": "US|US|A|Z222454357|123",
    "Type": "Address",
    "Text": "123 Joice St",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94108"
  },
  {
    "Id": "US|US|ENG|94108-CA-SAN_FRANCISCO--ST-JOICE--123",
    "Type": "BuildingNumber",
    "Text": "123 Joice St Apt",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94108 - 4 Addresses"
  }
]

```

The service assumes that you're looking for mailing addresses. These can be found in the response: all JSON objects which have the `Type` field set to the value `Address`. In this simple use case, the UI could suppress all other types (such as `BuildingNumber`, `Street`, and so on) and display only address objects.

If you see the address you want, such as “123 Guerrero St”, click that. This causes the client application to issue a request to the `/address/retrieve` endpoint, passing an `Id` parameter with the value `US|US|A|Z222437371|123`, which then retrieves the full address record. Otherwise, continue to enter more of the address, and the UI continually issues equivalent `find` requests, until the desired address is found.

### Retrieve Addresses

After an ID has been found, get the full and componentized information for this address by invoking the `/address/retrieve` endpoint with the `Id` parameter. The simple `find` example then produces the following results:

### Version 2 Request

```
http://adc00qrw.us.oracle.com:7767/av/api/v2/address/retrieve?Id=US|US|A|Z222437371|123
```

### Version 2 Response

```
[
  {
    "addressid": "US|US|A|Z222437371|123",
    "address1": "123 Guerrero St",
    "address2": "",
    "address3": "",
    "address4": "",
    "dependentlocality": "",
    "doubledependentlocality": "",
    "city": "San Francisco",
    "subadminarea": "San Francisco",
    "adminarea": "CA",
    "superadminarea": "",
    "postalcode": "94103-1072",
    "postalcodeprimary": "94103",
    "postalcodesecondary": "1072",
    "country": "United States",
    "fulladdress": "123 Guerrero St|SAN FRANCISCO CA 94103-1072",
    "countrycode": "US",
    "verificationcode": null,
    "verified": null,
    "verificationcodedescription": null,
    "latitude": null,
    "longitude": null,
    "geoaccuracycode": null,
    "geoaccuracycodedescription": null,
    "geodistance": null
  }
]
```

### Version 3 Request

```
http://adc00qrw.us.oracle.com:7767/av/api/v3/address/retrieve?Id=US|US|A|Z222437371|123
```

## Version 3 Response

```
[
  {
    "Id": "US|US|A|Z222437371|123",
    "Premise": "123",
    "Thoroughfare": "Guerrero St",
    "Locality": "San Francisco",
    "DeliveryAddress1": "123 Guerrero St",
    "SubAdministrativeArea": "San Francisco",
    "AdministrativeArea": "CA",
    "AdministrativeAreaName": "California",
    "PostalCode": "94103-1072",
    "CountryName": "United States",
    "ISO3166-2": "US",
    "ISO3166-3": "USA",
    "ISO3166-N": "840",
    "Address": "123 Guerrero St|SAN FRANCISCO CA 94103-1072",
    "DeliveryAddress": "123 Guerrero St",
    "Address1": "123 Guerrero St",
    "Address2": "SAN FRANCISCO CA 94103-1072",
    "PostalCodePrimary": "94103",
    "PostalCodeSecondary": "1072"
  }
]
```

## Search Advanced Drilldown Use Case

Suppose that you want to search for the address “941 Dolores St, unit #3”. You enter 941. The following Request 1 and Response 1 show the corresponding REST transaction initiated by the UI.

In this advanced drilldown search case, the UI takes all types of records into account, not only `Type=Address`. From the results in Response 1, you might not continue typing additional characters, but instead select the result “Dolores St Apt”, where `Type` is `BuildingNumber`.

The UI understands that this is not a final selection of an address, but rather a drilldown into one of the results, which contains four more addresses. This is accomplished by using the result’s `Id` and passing it as a `Container` parameter into a new request, as shown in Request 2. In Response 2, the previous result is not visible anymore, but all four apartments are presented. After seeing the desired #3 unit, you get the record by calling the `/address/retrieve` endpoint using the `Id` value of `US|US|A|Z222467771|3`.

### Request 1

```
http://adc00qrw.us.oracle.com:7767/av/api/v3/address/find?Text=941
```

### Response 1

```
[
  {
```

```

    "Id": "US|US|A|Z222467773|941",
    "Type": "Address",
    "Text": "941 Dolores St",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94110"
  },
  {
    "Id": "US|US|ENG|94110-CA-SAN_FRANCISCO--ST-DOLORES--941",
    "Type": "BuildingNumber",
    "Text": "941 Dolores St Apt",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94110 - 4 Addresses"
  },
  {
    "Id": "US|US|ENG|94117-CA-SAN_FRANCISCO--ST-PAGE--941",
    "Type": "BuildingNumber",
    "Text": "941 Page St Apt",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94117 - 5 Addresses"
  },
  {
    "Id": "US|US|A|Z222490758|941",
    "Type": "Address",
    "Text": "941 Page St",
    "Highlight": "0-3",
    "Description": "San Francisco CA 94117"
  },
  {
    "Id": "US|US|A|Z222412877|941",
    "Type": "Address",
    "Text": "941 Ridgeview Ct Unit",
    "Highlight": "0-3",
    "Description": "South San Francisco CA 94080"
  },
  {
    "Id": "US|US|A|Z222412876|941",
    "Type": "Address",
    "Text": "941 Ridgeview Ct",
    "Highlight": "0-3",
    "Description": "South San Francisco CA 94080"
  },
  {
    "Id": "US|US|A|Z213529633|941",
    "Type": "Address",
    "Text": "941 Shorepoint Ct",
    "Highlight": "0-3",
    "Description": "Alameda CA 94501"
  },
  {
    "Id": "US|US|ENG|94501-CA-ALAMEDA--CT-SHOREPOINT--941",
    "Type": "BuildingNumber",
    "Text": "941 Shorepoint Ct Apt",

```

```

    "Highlight": "0-3",
    "Description": "Alameda CA 94501 - 146 Addresses"
  },
  {
    "Id": "US|US|ENG|94608-CA-EMERYVILLE--ST-37TH--941",
    "Type": "BuildingNumber",
    "Text": "941 37th St Apt",
    "Highlight": "0-3",
    "Description": "Emeryville CA 94608 - 5 Addresses"
  },
  {
    "Id": "US|US|A|Z224475658|941",
    "Type": "Address",
    "Text": "941 Aileen St",
    "Highlight": "0-3",
    "Description": "Emeryville CA 94608"
  }
]

```

## Request 2

```

http://adc00qrw.us.oracle.com:7767/av/api/v3/address/find?
Text=941&Container=US|US|ENG|94110-CA-SAN_FRANCISCO--ST-DOLORES--941

```

## Response 2

```

[[
  {
    "Id": "US|US|A|Z222467771|1",
    "Type": "Address",
    "Text": "941 Dolores St Apt 1",
    "Highlight": "",
    "Description": "San Francisco CA 94110"
  },
  {
    "Id": "US|US|A|Z222467771|2",
    "Type": "Address",
    "Text": "941 Dolores St Apt 2",
    "Highlight": "",
    "Description": "San Francisco CA 94110"
  },
  {
    "Id": "US|US|A|Z222467771|3",
    "Type": "Address",
    "Text": "941 Dolores St Apt 3",
    "Highlight": "",
    "Description": "San Francisco CA 94110"
  },
  {
    "Id": "US|US|A|Z222467771|4",
    "Type": "Address",
    "Text": "941 Dolores St Apt 4",

```

```

    "Highlight": "",
    "Description": "San Francisco CA 94110"
  },
  {
    "Id": "US|US|A|Z222467772|A",
    "Type": "Address",
    "Text": "941 Dolores St Apt A",
    "Highlight": "",
    "Description": "San Francisco CA 94110"
  }
]

```

For detailed information, see *REST API for Oracle Address Verification Cloud Service*.

## Verify Addresses

Use Address Verification in verify mode to do the following tasks:

- Check if an address is valid (`addressclean`).
- Correct and enhance an address. For example, you can correct misspelled city or street names; add missing elements, like full postal code or state; and get geographic location, such as latitude, longitude.
- Get accurate representation of address from postal files. For example, change “street” to “St” or “parkw” to “Pkwy”.
- Transliterate an address; for example, you can input addresses in Arabic and get the verified output in English. This is supported for a native language to Latin (English) outputscript or from Latin language to a native outputscript. The following values are currently supported for the `OutputScript` option:
  - Latn – Latin (English transliteration wherever possible)
  - Cyrl – Cyrillic (Russia)
  - Grek – Greek (Greece)
  - Hebr – Hebrew (Israel)
  - Hani – Kanji (Japan)
  - Hans – Simplified Chinese (China)
  - Arab – Arabic (United Arab Emirates)
  - Thai – Thai (Thailand)
  - Hang – Hangul (South Korea)
  - Native – Output in the native script wherever possible

If the result is verified or changed according to the configuration specified, then `verified=Y`. By default, if no option is specified, then the output script will match the input wherever possible.

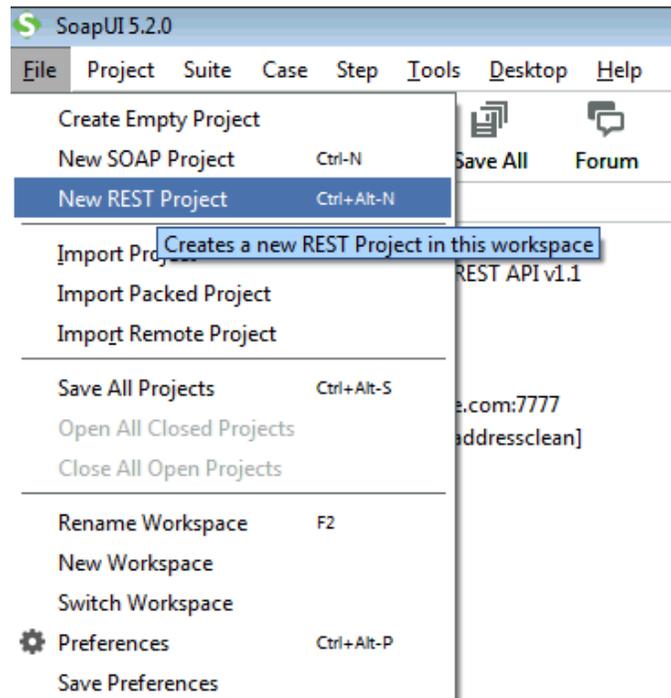
 **Note:**

Do not set `minimumverificationmatchscore` to 100 unless you know you have a valid address and you just want standardize it. In general, use a country's default `minimumverificationlevel` (for example, US=4). The exception is if you need to verify a zip code.

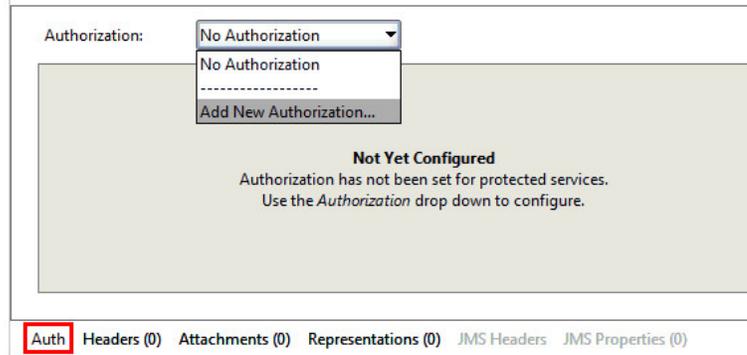
## Use a REST Client to Make Calls to Address Verification Cloud Service

This section shows the steps to verify an address using SoapUI. The process is the same for a single address verification request and for multiple (batch) requests.

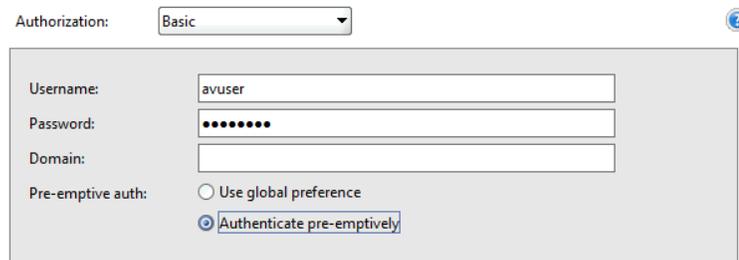
1. In SoapUI, select **File** — **New REST Project**.



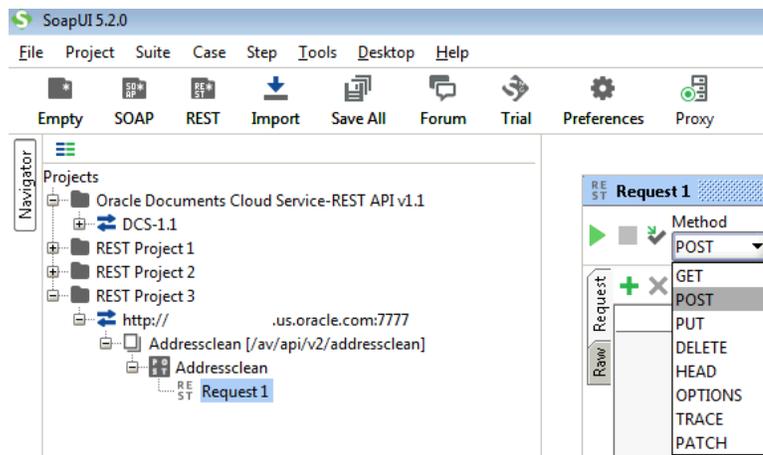
2. Enter the Address Verification Cloud Service URL as `http://host:port/av/api/v2/addressclean`.
3. In the new REST project, click **Auth** to add new authorization.



4. Select **Basic** for the authorization type (or create one if it is not present).
5. Enter user credentials, and select **Authenticate pre-emptively**.



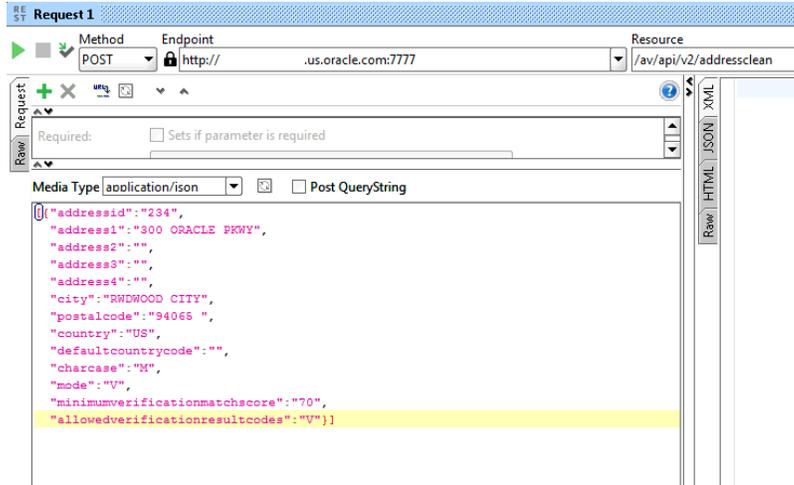
6. From the Method list, select **POST**.



7. Enter the request in the **Media Type** (application/json) window. For example:

```
[{"addressid": "234",
"address1": "300 ORACLE PKWY",
"address2": "",
"address3": "",
"address4": "",
"city": "REDWOOD CITY",
"postalcode": "94065 ",
"country": "US",
```

```
"defaultcountrycode": "",
"charcase": "M",
"mode": "V",
"minimumverificationmatchscore": "70",
"allowedverificationresultcodes": "V"}]
```



8. Submit the request. On success, it sends any output records back.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

## Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

---

Oracle® Cloud Using Oracle Address Verification Cloud Service,  
E59772-12

Copyright © 2016, 2019, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental

regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.