Bridge Documentation

[API Dashboard](https://bridgedataoutput.com/login)

Bridge is a data distribution platform that allows MLSs and other data providers to manage licensing, billing and data access using APIs built on modern industry standards.

For software developers, the platform standardizes how you access MLS datasets and other real estate data in the United States and Canada. MLS data is normalized to RESO Data Dictionary standards and made available through a variety of transports, including [RESO Web API](https://bridgedataoutput.com/docs/platform/#reso-web-api) and [RETS](https://bridgedataoutput.com/docs/platform/#rets).

Support and Updates

Recent updates to the platform and functionality can be found here: <https://www.bridgeinteractive.com/bridge-platform-updates/>

If you have any questions about using the APIs available on the platform, contact support at [api@bridgeinteractive.com](mailto:api@bridgeinteractive.com).

For questions related to MLS licensing agreements and listing data, it’s usually best to talk to the MLS directly, as all access is at their discretion.

At this time support is only offered in English.

Getting started with the Bridge Platform

The Bridge platform is comprised of two components:

* The platform dashboard: for managing data access, API tokens and documentation.
* The data transports: [RESO Web API](https://bridgedataoutput.com/docs/platform/#reso-web-api), [RETS](https://bridgedataoutput.com/docs/platform/#rets) and [Bridge Web API](https://bridgedataoutput.com/docs/platform/#bridge).

Access to the Bridge platform is typically initiated by the MLS. If you are looking to access MLS data, first contact the relevant MLS to request an invite to the platform.

Step 1: Registering on the Bridge platform

If you are invited directly by a data provider to the Bridge platform, click the link in the email invitation and follow the prompts to complete the registration process.

Once registered, you will be able to log into the dashboard to manage various aspects of your applications, including getting API tokens and data. If the data provider has included data licenses in your invitation, you'll be able to execute the agreements and manage your billing if applicable.

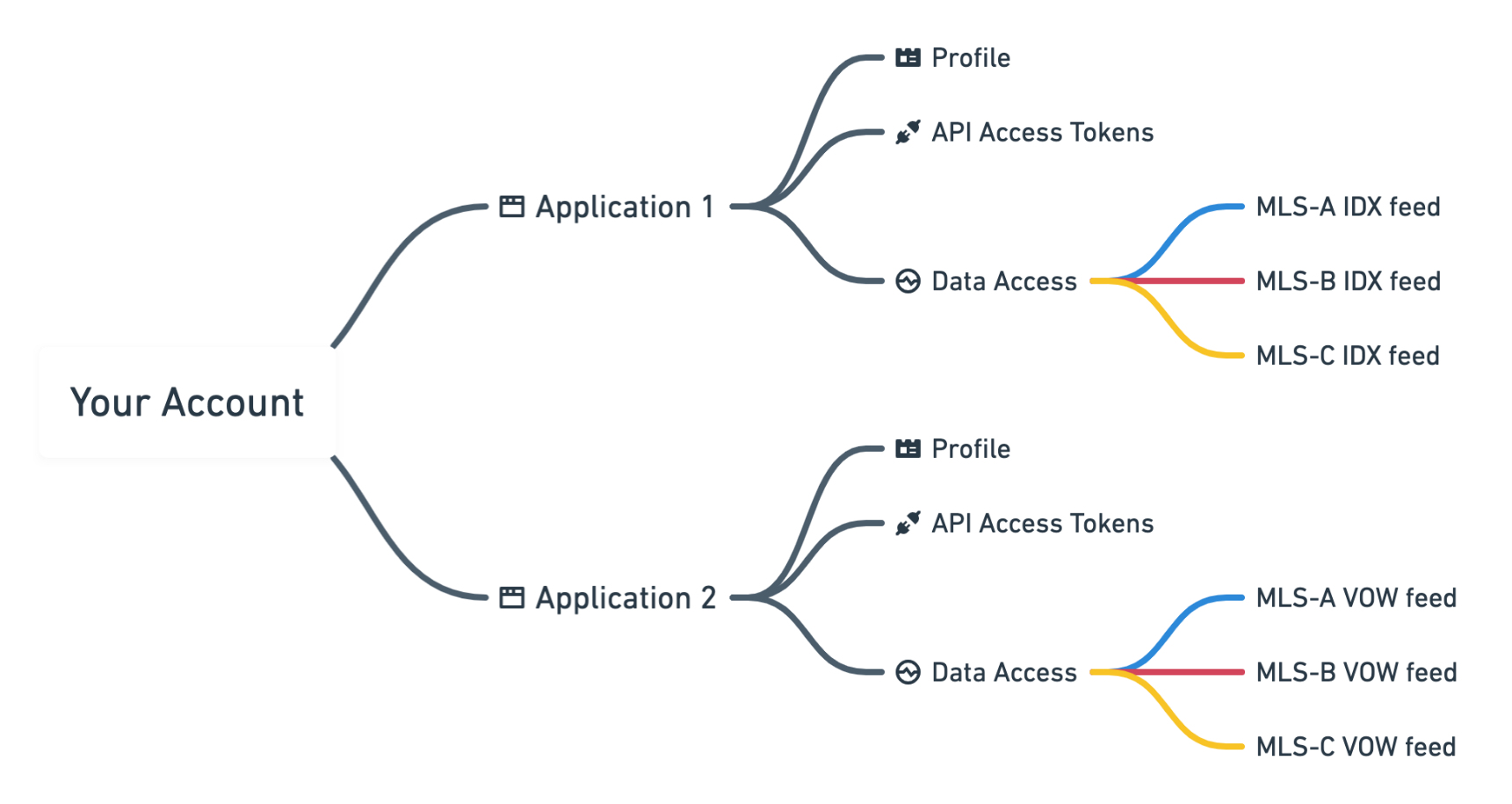
What is an application?

An application represents a product that consumes data using the API and requires a distinct API token. Elements linked to an application include:

* The application profile
* API tokens
* Data access approvals for various datasets
* API traffic logs

When registering your account, your first application is automatically generated. While most users will only need a single application, you may create as many as you need. Typically you’ll only need more than one if you are receiving multiple feeds from a single MLS.

For example, if you are receiving an IDX feed and a VOW feed from a single MLS, then you will need to create an additional application for the second feed. This can then be re-used for other MLSs that also want to give multiple feeds. Note that you will now have two sets of API access tokens - one for each type of feed.



Your Primary application

By default, your first application is set as the Primary application. Any data access that is shared with you (either from an MLS invite or by a Broker) is provisioned against your Primary application. You can specify which application is Primary under Application Settings on the dashboard.

Application Profile

Your application profile is shown to data providers during the approval process, and allows you to add details that will help them decide whether or not to grant access and what type of data feed you should get.

If you are a vendor working with a specific brokerage, be sure to include the brokerage name and contact details.

Including details about what data you need, as well as how and where the data will be used, will speed up the approval process. Consider adding a link to your application or website, as well as screenshots to help provide context for your request.

Note that each application will have its own profile, and they can be updated at any time.

Merging Accounts

If you have more than one account on the platform, you may use the tool in your account settings to merge them together. By doing so, you will combine all the applications, users, agreements and payment profiles into a single account. You will require the platform credentials for the account you wish to merge.

Since all the users will still be active, you are able to sign into the platform with any of the credentials. You may de-activate a user by adjusting your account settings.

Accounts can only be merged if they have the same role (broker, agent or vendor). If you require the account role to be updated, please reach out to support.

Account merges cannot be undone.

If you previously had separate accounts for Bridge Agreement Management and Bridge API, you will be able to merge these, allowing you to see your data access and agreements in a single dashboard.

If you sign in using your MLS's SSO, you may still merge with your Bridge API account, and continue to sign in using either method. You are not able to merge two SSO accounts together.

If your Bridge Agreement Management and Bridge API used the same email and password, you'll be signed in to your Bridge API account by default. To see your agreements you'll need to merge in your Agreement Management account.

Step 2: Data Access

The Bridge platform allows you to request data access from any of our customer data providers. All datasets available through the platform require approval from the data provider. Whether they approve your request, or what is required to get that approval, is at their discretion. In some cases, data providers may require you to execute data license agreements, pay fees or do compliance testing.

A list of available datasets you can apply for can be found in your dashboard. Not all available MLSs are listed here; some have additional requirements you must fulfil in order to apply for the dataset.

If you are a member of an MLS that you want to work with, and you do not see that MLS on the dashboard, please reach out to support to see if we are able to facilitate access.

If you have been invited onto the platform by an MLS, your Primary application will automatically have data access to that MLS in a pending state. Once the MLS makes final approval, that status will change to “Approved”, and you will be able to retrieve data via the API.

Data Access Statuses

* Pending: Your application has been submitted to the data provider and is pending approval.
* Approved: Your application has been approved by the data provider, and the data is available through the Bridge.
* Denied: Your application has been denied for a data license. A reason will be supplied when you are notified of this status change. You may apply again if desired.
* Suspended: Your application has been temporarily suspended.
* Revoked: The data provider has revoked your access to their data. The data feed for this resource is no longer available. You will need to reapply to regain access.

Data Approval

MLSs will approve your application for specific feed types. These typically represent whether access is provisioned for an IDX, VOW, BBO, or other data license.

Each feed type has its own set of rules that govern the subset of data you are granted. These rules include which fields and which records are available, as well as any conditional field suppression that might apply.

Indication of what data you have been granted access to is available using the metadata endpoint for each dataset.

The MLS is also able to specify whether you are allowed to replicate datasets (as opposed to only making on-demand API queries), as well as to specify the ability for brokers to share data access with 3rd parties.

You will be notified by email when an MLS has approved your data access request.

Test datasets

Every application is automatically approved for access a Test dataset. This is machine-generated and useful when trying an initial integration with the platform using generic, static data. Note that test data may not accurately match real data that will be provisioned, which will vary between MLSs.

This dataset should not be used for building production integration tests, as it is subject to change.

Combined Feed Types

MLSs may approve a single application for multiple feed types. This is done to create efficiencies for data consumers, who may have multiple data licenses with an MLS.

By combining multiple feed types within a single approval, instead of maintaining two or more feeds with similar data, all the data available to you from the MLS is accessible in one.

The API response will include a superset of all available listings, while still respecting the underlying rules for each type.

Managing Combined Feed Types

Each record in an API response contains the field “FeedTypes”. This automatically-generated field indicates which feed type(s) the record is sourced from - and therefore what license rules would be applicable.

Splitting combined feeds

While the FeedTypes field itself not queryable, you are able to split a combined feed into its individual components by adding the feed type metadata tag into the API path:

Return only IDX properties from a combined feedhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/idx/Properties

Field level rules

In addition to knowing which license type governs a listing, you are also able to use the API to determine the licenses type for each field. This is important for the scenario where a listing is sourced from multiple feed types that have different field payloads.

Since the payloads are combined, this functionality makes it easier to determine which fields are allowed to be used for each feed type. This information is available using the “FieldRules” resource. This resource allows you to answer the following two questions:

* “What data licenses back this field?”
* “What fields can be used as part of this data license”

The values returned in this resource are automatically derived from the field payloads available as part of the underlying feed types that make up your combined feeds.

Note that fields in this resource are not queryable. If you'd like to see only the relevant fields for a specific subset of a combined feed you may include the feed type metadata tag into the URL path:

Return only IDX fieldshttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/idx/Field

Shared MLS data access

When approving your application, an MLS may also allow you to share your data access with addtional 3rd party vendors, at your discretion. This ability is only available for broker and agent accounts.

To share data access, go to Share Data on your dashboard, and enter the details of the vendor you want to give access. If this vendor is not on the platform they will be sent an invite, otherwise data access will be provisioned on their Primary application.

If you are a vendor, any data access that has been shared with you is visible under Approved Datasets on your dashboard.

Virtual MLS datasets

Virtual Datasets are designed to allow you to make on-demand queries for data across your approved MLS datasets using a single endpoint. You are able to choose which datasets you'd like to include in each virtual dataset, and you may create as many as you need.

* Virtual Datasets may only be comprised of your approved MLS datasets
* The Metadata and Replication endpoints are not available for Virtual Datasets.
* Virtual Datasets are only available using the RESO Web API and Bridge Web API.
* The response schema for each record is identical to the schema returned by the parent dataset. Schemas are not normalized between child datasets.

Step 3: Using the APIs

The platform offers various transport mechanisms to choose from when accessing MLS data:

* [RESO Web API](https://bridgedataoutput.com/docs/platform/#reso-web-api)
* [RETS](https://bridgedataoutput.com/docs/platform/#rets)
* Our proprietary [Bridge Web API](https://bridgedataoutput.com/docs/platform/#bridge)

All APIs make the same data available (ie whatever the specific data provider has allowed you access to). For MLS data, we recommend using the RESO Web API, as it conforms to the latest industry standards and will provide the most interoperability with other MLS data distribution platforms.

MLS data is normalized to RESO Data Dictionary standards. Any fields that are not able to be normalized to the RESO Data Dictionary are name-spaced with a prefix corresponding to the dataset.

Each API request should contain the dataset code of the MLS, the resource you are querying, your access token, and any parameters you’d like to use to constrain the result set. HTTPS is required, as the Bridge only respond to encrypted traffic.

Program defensively! MLS data is prone to change as data providers have the ability to update restrictions and field payloads at their discretion. To avoid disruption, we recommend that you program with this expectation in mind.

Dataset Codes

All API requests need to include a dataset code. This indicates which dataset you wish to query and either represents a specific data provider or a virtual dataset consisting of multiple MLS datasets.

Resources

Records are organized into standardized resources: typically Properties, Members, and Offices. Most MLSs will also have Open Houses available and possibly other resources like Rooms and UnitTypes.

API Access Tokens

All data access for an application is authorized and authenticated using API tokens. Your Client ID, Client Secret, Browser and Server tokens can be found on your dashboard.

For Web API we recommend using the Server token to implement server-to-server API requests. Including your token in user-accessible client code may allow unauthorized access to your data feeds.

|  |  |
| --- | --- |
| Token | Description |
| Client ID | Use as your username for the RETS API |
| Client Secret | Use as your password for the RETS API |
| Server Token | Use as the bearer token for API requests |
| Browser Token | Used for websites that may query the API directly from the browser; be sure to set the Referrer Domain if you use this approach |

IP and Domain whitelisting

Authorization tokens will only work on the HTTP referrer domains and IP addresses you have specified for your application settings in the dashboard. You may specify multiple domains, but no wildcards. If no domains or IPs have been set, this validation is skipped.

Regenerating your Client Secret

You're able to regenerate your Client Secret at any time. It's important to note that the Client Secret is used in the generation of both Server and Browser tokens, and when regenerating you are effectively expiring all tokens that were generated off that Client Secret. It may be useful to regenerate your Client Secret if your Client Secret or any of the subsequent tokens have been compromised.

Rate limits

There is no usage cap on the total number of API requests or concurrent requests you can make against a token. However, there is a default rate limit of 5,000 requests per hour. If your application has high usage, please reach out to discuss increasing the limit.

There are three rate limit headers that are included in every response from the Bridge.

|  |  |
| --- | --- |
| Rate Limit Header | Description |
| Application-RateLimit-Limit | The maximum number of API requests that can be made in a given period of time (determined by the ratelimit-reset header) |
| Application-RateLimit-Remaining | The number of API requests that can still be made. This number will be set to the value of ratelimit-limit when the rate limit resets |
| Application-RateLimit-Reset | The UTC time at which the rate limit resets |

Error Codes

The Web API may return the following HTTP status codes. For non-200 statuses, we also return an error object in the response body, with a message containing additional specifics about the error.

|  |  |
| --- | --- |
| HTTP Status Code | Description |
| 200 | OK |
| 400 | Bad request |
| 401 | Unauthorized |
| 403 | Forbidden |
| 404 | Not found |
| 408 | Request timeout |
| 415 | Unsupported MIME type |
| 429 | Too many requests |
| 500 | Internal server error |
| 503 | Service unavailable |

Using the API Explorer

Our API Explorer is the simplest way to query the API. It allows you to build and make API requests in the browser.

By default, the API Explorer is set to use a public API token that only has access to a test dataset. Alternatively, you can specify your application's Server token and any dataset you have been approved for.

You can use the UI to build simple queries using various parameters and filters using Data Dictionary fields. To handcraft and test more complex queries, we recommend using an API client such as [Postman](https://www.postman.com/).

RESO Web API

The Bridge platform allows you to query MLS data using the RESO Web API specification, which is based on OData. For more information about the specification, please visit the [RESO website](https://www.reso.org/reso-web-api/). The API has been RESO Platinum Certified to version 1.0.2.

Authorization

Your application should send an Authorization header with every HTTP request to the API:

Authorization: Bearer {token}

If you can't set headers, you can send the token in the access\_token parameter in the query string of your request:

GET https://api.bridgedataoutput.com/api/v2/OData/{dataset\_id}/{resource}?access\_token={server\_token}

API Response

By default, the RESO Web API will return 10 listings, regardless of the number of total records available. It is recommended that you use the maxpagesize preference header instead of $top parameter to specify your request to return up to 200 listings at a time.

If there are more than 200 records available, you will need to paginate through the results.

If you wish to paginate through more than 10,000 listings you will need to use the dedicated replication endpoint.

All API responses besides metadata are returned in JSON format.

Paginating through results

If the total number of records that are returned by a query is greater than 200 (the maximum limit of a result set), then you will need to paginate through the results. This is done by incrementing the number of skipped records.

Return the next 200 records, ordered by ListPricecurl -H "prefer: 'maxpagesize=200'" https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/{resource}?access\_token=access\_token&$skip=200&$orderby=ListPrice desc

Using Expand

Using the $expand operator allow you to include associated data from additional resources. For example, you are able to bring in more detail about the relevant office or member into the response payload for a property, without having to make a second or third API query to the other resources.

* You are not able to use all query parameters on data you’ve expanded into your response.
* If you are using the `$select` parameter in your query to limit the fields in the response payload, be sure to include the expanded field as well (eg, add the ListOffice field if you're expanding ListOffice)

Expand ListOffice in a Property querycurl https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$expand=ListOffice

Dataset Replication

While we encourage the use of the API to query the listing data as needed, there are use-cases where replication of the full dataset may be preferred.

To help with this, you are able to request data with the /replication endpoint. Whereas on-demand API requests can have a maximum of 200 results returned at once, with this endpoint the maximum maxpagesize is 2,000 results.

The header of the response will contain a 'next' link. Results are returned ordered from oldest to newest, so by using the next link you are able to pull down all the available records to seed your data, and then continue using the next link at regular intervals to keep up to date.

Use the replication endpointcurl -H "prefer: 'maxpagesize=2000'" https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property/replication?access\_token=access\_token

* If you are trying to replicate anything more than 10,000 records you will need to use the replication endpoint.
* Because of the potential payload size, this is only suitable for server to server requests.
* This functionality is only available at the discretion of the MLS that has authorized data access.
* Certain parameters like ‘skip’ and ‘orderby’ are not available with this endpoint.
* To reduce payload size and improve performance we recommend using the `select` parameter to request only the fields you need.
* BridgeModificationTimestamp is the best field to use for incremental updates as it represents the last modification in the Bridge system; ModificationTimestamp is ingested directly from the MLS system and is not a consistent proxy for modifications to the listing in the Bridge database.

Media

Rather than keeping it in a separate resource, Media is returned as an object directly on the Property record. Typically, it is the highest resolution media available from the MLS and is stored on our CDN. You may link directly to the CDN.

Metadata

You can request metadata that will return the fields and lookup values that have been made available to by the data provider. Metadata is returned in XML, according to the RESO spec.

Request metadatacurl https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/$metadata?access\_token=access\_token

Operators

You can constrain the result set of a resource by passing additional operators with your request. Valid operators include:

|  |  |
| --- | --- |
| Operator | Description |
| eq | Equal |
| ne | Not equal |
| gt | Greater than |
| lt | Less than |
| ge | Greater than or equal |
| le | Less than or equal |
| and | Logical and |
| or | Logical or |
| not | Logical not |

Parameters

The following query parameters may be passed to narrow down your results.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| access\_token | string | Token to identify the application. This is always required.Example request: return all datasets approved for an applicationhttps://api.bridgedataoutput.com/api/v2/OData/DataSystem?access\_token=access\_token |
| ListingKey | string | The listing key, available on the /Properties resource.return data relating to a listing where the ListingKey “12345”https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties(‘12345’)?access\_token=access\_token |
| MemberKey | string | The member key, available on the /Members resource.Return data relating to a member where the memberKey is “12345https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Members('12345')?access\_token=access\_token |
| OfficeKey | string | The office key, available on the /Offices resource.Return data relating to an office where the officeKey is “12345”https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Offices('12345')?access\_token=access\_token |
| $skip | number | Skips this number of resultsSkip the first 10 records of a datasethttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$skip=10 |
| $select | string | Select the fields to be returnedOnly return the LivingArea fieldhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$select=LivingArea |
| $unselect | string | Select the fields to be exludedDo not return the Media objecthttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$unselect=Media |
| $filter | string | Filter the results to be returnedOnly return the listings where the ListPrice is greater than $100000https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=ListPrice gt 100000 |
| $top | number | Limits the size of the result set. Default is 10, maximum is 200.Limit results from the Test dataset to only 2https://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$top=2 |
| $orderby | string | Response field to sort query by (either “desc” or “asc”)Sort order by descending pricehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$orderby=ListPrice desc |
| $expand | string | Include query specified entities inline with responseExpand the relevant listing agent for a given propertyhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$expand=ListAgent |

Query Functions

|  |  |
| --- | --- |
| Function | Description |
| any | Search fields where any element of an array is satisfied by a conditionReturn listings where there is an option of electric heatinghttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=Heating/any(a: a eq 'Electric') |
| all | Search fields where all elements of an array is satisfied by a conditionReturn listings where all of the flooring is hardwoodhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=Flooring/all(a: a eq 'Hardwood') |
| geo.distance | Search by coordinatesReturn listings that are near specific co-ordinates, to a radius of 0.5 mileshttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=geo.distance(Coordinates, POINT(-118.62 34.22)) lt 0.5 |
| geo.intersects | If you know the extents of a polygonal region, you can provide the each point as co-ordinatesReturn listings within a shapehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=geo.intersects(Coordinates, POLYGON((-127.02 45.08,-127.02 45.38,-127.32 45.38,-127.32 45.08,-127.02 45.08))) |
| tolower | Search fields with lowercase queriesReturn listings using a lowercase queryhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=tolower(StandardStatus) eq 'active' |
| startswith | Search fields by a string prefixReturn listings using a city that starts with a specific stringhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=startswith(City, 'Spring') |
| endswith | Search fields by string endingReturn listings using a city that ends with a specific stringhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=endswith(City, 'field') |
| contains | Search field by string inclusionReturn listings using a city that contains a specific stringhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=contains(City, 'nge') |
| date | Search fields by dateReturn listings with a specific datehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=date(ModificationTimestamp) eq 2017-08-29 |
| time | Search fields by timeReturn listings with a specific timehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=time(ModificationTimestamp) eq 17:03:04 |
| year | Search fields by yearReturn listings with a specific yearhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=year(ModificationTimestamp) eq 2017 |
| month | Search fields by monthReturn listings with a specific monthhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=month(ModificationTimestamp) eq 12 |
| day | Search fields by dayReturn listings with a specific dayhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=day(ModificationTimestamp) eq 23 |
| hour | Search fields by hourReturn listings with a specific hourhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=hour(ModificationTimestamp) eq 17 |
| now() | Search fields by current timestampReturn listings within the current timestamphttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Properties?access\_token=access\_token&$filter=ModificationTimestamp eq now() |

Examples

Search for a specific property by ListingIdhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=ListingId eq ‘123456789’Search for a property by addresshttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=UnparsedAddress eq ‘123 Main’Search for a property by address, using ‘tolower’ to work around API case-sensitivityhttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=tolower(UnparsedAddress) eq ‘123 main’Search for all residential properties that are on salehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=PropertyType eq ‘Residential’ and StandardStatus eq ‘Active’Search for all residential properties that are for renthttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=PropertyType eq ‘Residential Income’ and StandardStatus eq ‘Active’Search for all residential properties that are for sale and in a specific zip codehttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=PropertyType eq ‘Residential Lease’ and PostalCode eq ‘90210’ and StandardStatus eq ‘Active’Search for all properties that are in one of two zip codeshttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=PostalCode eq ‘12345’ or PostalCode eq ‘54321’Search using complex nested querieshttps://api.bridgedataoutput.com/api/v2/OData/dataset\_id/Property?access\_token=access\_token&$filter=((InternetEntireListingDisplayYN ne false) and ((StandardStatus eq ‘Closed’) and (((YearBuilt eq null) or ((YearBuilt le 1986) and (YearBuilt ge 1976))) and (((LivingArea eq null) or ((LivingArea le 3264) and (LivingArea ge 2412))) and ((CloseDate ge 2019-09-01) and (((BedroomsTotal eq null) or ((BedroomsTotal le 5) and (BedroomsTotal ge 3))) and (((BathroomsTotalDecimal eq null) or ((BathroomsTotalDecimal le 3.5) and (BathroomsTotalDecimal ge 1.5))) and ((SpecialListingConditions ne ‘Auction’) and ((SpecialListingConditions ne ‘Probate’) and ((SpecialListingConditions ne ‘Short Sale’) and ((SpecialListingConditions ne ‘REO’) and (((ClosePrice eq null) or ((ClosePrice le 472666) and (ClosePrice ge 315110))) and (geo.distance(Coordinates,POINT(-115.10998 36.091513)) lt 0.5)))))))))))))

RETS

Using the virtual RETS API (vRETS) you are able to get listing data just as you previously had with a traditional RETS server. This makes the Bridge platform backwards-compatible with existing data transport standards that are still used by many data consumers in the real estate space.

This API was built to allow data consumers to continue to use RETS as a transport layer if an MLS's RETS server is no longer available.

Any data consumer that has been approved for access (with replication) on the Bridge platform by an MLS is able to use RETS API.

The virtual RETS API is built according to the RESO specifications for RETS v1.9. It may not support functionality that extended the spec to deal with edge-cases specific to an MLS’s requirements.

It's important to note that the virtual RETS API will likely not be a drop-in replacement for an existing MLS RETS feed, as there will be a difference in the data being served. All data on the Bridge platform has been normalized to RESO Data Dictionary standards, unlike most MLS RETS feeds, and some fields may be generated differently from the native MLS server (e.g., short/system names).

Using your Bridge application Client ID and Secret as your RETS username and password, you can create a RETS session and make DMQL queries for data. Responses can be formatted in either XML or CSV. Data served by the RETS API is identical to what has been made available to you with the RESO Web API: normalized to RESO Data Dictionary standards and respecting any data restrictions that the MLS has put in place.

BridgeModificationTimestamp is the best field to use for incremental updates as it represents the last modification in the Bridge system; ModificationTimestamp is ingested directly from the MLS system and is not a consistent proxy for modifications to the listing in the Bridge database.

Complete documentation for using the RETS specification is available here: [RESO RETS Specifications](https://www.reso.org/downloads/).

Media

Media should be retrieved using GetObject. One key difference from previous RETS standards is that the default GetObject location behavior changed from Location=0 to Location=1. We only support Location=1 (returning the URL for the object rather than the object itself), which is in line with the RETS 1.9 specification.

Establishing a session

Establish a session with the virtual RETS API login URL, either through your preferred RETS client or directly. For Username, use your Client ID token. For Password, use your Client Secret. Both tokens can be found in your application profile on your dashboard under API Access.

Establish a sessionhttps://api.bridgedataoutput.com/api/v2/rets/dataset\_id/login

Querying for metadata

Complete documentation for requesting metadata using RETS is available here: [RESO RETS Specifications](https://www.reso.org/downloads/).

Query for metadatahttps://api.bridgedataoutput.com/api/v2/rets/dataset\_id/getMetadata?Type=METADATA-RESOURCE&ID=0

Querying for listing data

Complete documentation for requesting data using RETS is available here: [RESO RETS Specifications](https://www.reso.org/downloads/).

Query for Active listingshttps://api.bridgedataoutput.com/api/v2/rets/dataset\_id/search?SearchType=Property&Query=(StandardStatus=Active)

Tips

When searching for a value that has a space in the string, enclose it in double quotes.

Example query to search on PropertyType of Residential Leasehttps://api.bridgedataoutput.com/api/v2/rets/dataset\_id/search?SearchType=Property&Query=(PropertyType="Residential Lease")

Bridge Web API

While we generally recommend using the RESO Web API for querying MLS data, you are also able to use our simple native syntax as well.

This is also the only API that can be used for the non-MLS datasets available on our platform, like [Zillow Group Data](https://bridgedataoutput.com/docs/platform/#zg-data).

Authorization

Your application should send an Authorization header with every HTTP request to the API:

Authorization: Bearer {token}

If you can't set headers, you can send the token in the access\_token parameter in the query string of your request:

GET https://api.bridgedataoutput.com/api/v2/{dataset\_id}/{resource}?access\_token={server\_token}

API Response

By default, the Bridge Web API will return 10 listings, regardless of the number of total records available. You may use the limit parameter to specify your request to return up to 200 listings at a time.

If there are more than 200 records available, you will need to paginate through the results.

If you wish to paginate through more than 10,000 listings you will need to use the dedicated replication endpoint, which is only available when using the RESO Web API.

All API responses besides metadata are returned in JSON format.

Paginating through results

If the total number of records returned by a query is greater than 200 (the maximum limit of a result set), then you will need to paginate through the results. This is done by incrementing the number of skipped records.

When paginating through records, it's necessary to sort the results as there is no guarantee on the order, and multiple pages may include the same record more than once.

Return the next 200 records, sorted by ListPricehttps://api.bridgedataoutput.com/api/v2/dataset\_id/{resource}?access\_token=access\_token&limit=200&offset=200&sortBy=ListPrice&order=asc

Operators

You can constrain the result set of a resource by passing additional operators with your request.

|  |  |
| --- | --- |
| Operator | Description |
| eq | equal |
| ne | not equal |
| gt | greater than |
| lt | less than |
| gte | greater than or equal |
| lte | less than or equal |
| in | comma-separated list, equal only |
| nin | not in comma-separated list |

If the operator is omitted, the eq operator is assumed.

Parameters

The following query parameters may be passed to narrow down your results. Note that geographic coordinates should be ordered {Lng},{Lat} and contain a decimal.

Path Parameters

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| access\_token | string | Token to identify the application. This is always required.Example request: return all datasets approved for an applicationhttps://api.bridgedataoutput.com/api/v2/datasets?access\_token=access\_token |
| datasetID | string | Specifies the dataset to get the data from.Return all listings from a datasethttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token |
| listingId | string | The listing ID, available on the /listings resource.return data relating to listing with the key 12345https://api.bridgedataoutput.com/api/v2/dataset\_id/listings/12345?access\_token=access\_token |
| agentId | string | The Agent ID, available on the /agents resource.Return data relating to an agent with the key 12345https://api.bridgedataoutput.com/api/v2/dataset\_id/agents/12345?access\_token=access\_token |
| officeId | string | The office ID, available on the /offices resource.Return data relating to an office with the key 12345https://api.bridgedataoutput.com/api/v2/dataset\_id/offices/12345?access\_token=access\_token |

Query parameters

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| offset | number | Skips this number of results.Skip the first 10 listingshttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&offset=10 |
| limit | number | Limits the size of the result set. Default is 10, maximum is 200Limit results from a dataset to 200https://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&limit=200 |
| sortBy | string | Response field to sort query by.Sort order by pricehttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&sortBy=ListPrice |
| order | string | Order of responses: "asc/desc".Sort order by descending pricehttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&sortBy=ListPrice&order=asc |
| near | string | Coord or location e.g. near={Lng},{Lat} or near=San DiegoReturn listings that are near specific co-ordinateshttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&limit=200&near=-122.395743,37.793662 |
| radius | string | Radius to search within. Can be suffixed with units mi(miles) or km(kilometer). Miles are default.Return listings that near to specific co-ordinants, constrained 5 mileshttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&limit=2&near=-122.395743,37.793662&radius=5mi |
| box | string | If you know the extents of a box-shaped region, you can provide the lower-left and upper-right co-ordinates "{lng1},{lat1},{lng2},{lat2}" as a string or you can provide them as an array [[{lng1},{lat1}],[{lng2},{lat2}]] box parameter.Return listings within a boxhttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&box=-122.435676,37.781368,-122.415764,37.799580 |
| poly | string | If you know the extents of a polygonal region, you can provide the each point as co-ordinates "{lng-1},{lat-1},{lng-2},{lat-2}...{lng-n},{lat-n}" as a string or you can provide them as an array [[{lgn-1},{lat-2}],[{lng-2},{lat-2}]]....,[{lng-n},{lat-n}]] poly parameter.Return listings within a box-shapehttps://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&poly=-122.435676,37.781368,-122.415764,37.799580,-122.447092,37.796291 |

Examples

Get a list of all datasets approved for an applicationhttps://api.bridgedataoutput.com/api/v2/datasets?access\_token=access\_tokenGet listings in the dataset where price is greater than $500,000https://api.bridgedataoutput.com/api/v2/dataset\_id/listings?access\_token=access\_token&ListPrice.gt=500000Get a set of agents in the dataset whose name is not 'Ursula'https://api.bridgedataoutput.com/api/v2/dataset\_id/agents?access\_token=access\_token&firstName.ne=UrsulaGet a set of offices in the dataset that are located in Vermonthttps://api.bridgedataoutput.com/api/v2/dataset\_id/offices?access\_token=access\_token&state=VT

Zillow Group Data

As well as MLS data, the Bridge API platform also offers access to various datasets from Zillow Group:

* [Public Records](https://bridgedataoutput.com/docs/platform/#Public-Records-data)
* [Zestimates](https://bridgedataoutput.com/docs/platform/#Zestimates)
* [Zillow Group Economic Data](https://bridgedataoutput.com/docs/platform/#Econ)
* [Zillow Agent Reviews](https://bridgedataoutput.com/docs/platform/#agent-reviews)

Access to these datasets requires approval from Zillow Group and can be applied for in the same way as MLS datasets.

Before applying for Zillow Group data, please ensure your use-case is allowable by the [Zillow Terms of Use.](https://bridgedataoutput.com/zillowterms)

We do receive a high volume of requests for this data so please allow at least 10+ business days for us to determine your eligibility.

Please include the following information in your application profile before applying:

* Are you requesting this for Personal or Business use?
* Who is the audience?
* How will the data be used?
* Is this a paid service?
* Will you be storing data?
* Will you be displaying Zillow data as an as-is metric or combining it with other data to create a derivative work/calculation?
* Where are you and your company located?
* You are able to query all these datasets using the [Bridge Web API](https://bridgedataoutput.com/docs/platform/#bridge), as well as the [RESO Web API](https://bridgedataoutput.com/docs/platform/#reso-web-api) for Zestimates
* Replication and metadata endpoints are not available.
* Access requires adherence to the Zillow Data Terms of Use.

Public Records Data

This includes property records, tax assessments, and transaction records for approximately 148,000,000 properties across 3,200 counties throughout the United States. By default, you are limited to 1,000 API requests per day.

Coverage for any given data point, as well as update frequency, may differ between counties depending on how their data is made available.

Parcels

The Parcels resource includes a document for each property in our dataset, whether it's a detached home, a strata lot, a business, a bare land property, or any other kind of property. It’s typically a subset of data available from the Assessments and Transactions resources. Each record will contain links to the Assessments and Transactions relevant to that parcel.

Assessments

The Assessments resource includes present and historical property tax assessments. This includes estimated market and taxable values, tax amounts, and information about the property itself.

Transactions

The Transactions resource includes present and historical property transactions. Any type of change to a property's ownership is reflected in the transactions data set, including sale of new and existing properties, title transfers within families, and finance changes (such as a home loan refinance).

Public Records Field Enumerations

The following is a list of possible values for any fields with closed enumerations for the parcels, transactions and assessments resources: [Public Records Enumerations](https://bridgedataoutput.com/assets/public_records.xls)

Examples

Query parcel by with the APN "12345"https://api.bridgedataoutput.com/api/v2/api/v2/zestimates\_v2/zestimates?access\_token=access\_token&apn=12345Get all transactions for a specific parcel with the id "12345"https://api.bridgedataoutput.com/api/v2/pub/parcels/12345/transactions?access\_token=access\_tokenQuery for all assessments for "123 Main Street", sorted by yearhttps://api.bridgedataoutput.com/api/v2/pub/assessments?access\_token=access\_token&zpid=55023086&address.full="123 Main Street"&sortBy=year

Zestimates

This API will return all current Property and Rental Zestimates available on Zillow.com. Note that not all properties will have associated Zestimates. For more information about Zestimates, visit [Zillow.com](https://www.zillow.com/zestimate/).

Examples

Query Zestimate by addresshttps://api.bridgedataoutput.com/api/v2/zestimates\_v2/zestimates?access\_token=access\_token&address="123 Main Street"Query by list of ZPIDshttps://api.bridgedataoutput.com/api/v2/zestimates\_v2/zestimates?access\_token=access\_token&zpid.in=123456,54321,15243

Zillow Group Economic Data

This API provides housing market metrics otherwise made available as CSV downloads on [www.zillow.com/data](https://www.zillow.com/data). This API is intended for organizations and individuals who want programmatic access to those metrics for their own tools and apps.

Most metrics are published monthly. You can access the most recent publication date by accessing the releaseDate field in the API.

Zillow's Econ team has created some supplemental docs with some examples using this API: [Additional Documentation](https://documenter.getpostman.com/view/9197254/UVsFz93V)

Note that the Create Date is the date the metric was calculated, while Report Date is the date that the metric represents. For example, if you want to know what the Zillow Home Value Index was on 4/31, the Report Date would be 4/31, while the Create Date might say 5/10 if it was calculated 10 days later.

Examples

Get market report for State FIPS "48"https://api.bridgedataoutput.com/api/v2/zgecon/marketreport?access\_token=access\_token&stateCodeFIPS=48Use the Region endpoint to get metadata for State FIPs "48"https://api.bridgedataoutput.com/api/v2/zgecon/region?access\_token=access\_token&stateCodeFIPS=48Use the Type endpoint to get metadata about the metricTypeKey "uloc" in a market reporthttps://api.bridgedataoutput.com/api/v2/zgecon/type?access\_token=access\_token&key="uloc"

Zillow Agent Reviews

Consumers increasingly make decisions on what product or service to use after reviewing online reviews, and real estate agents are no different. The Zillow Agent Reviews API is designed to assist the consumer in choosing the best agent for their particular set of needs.

At Zillow, we have found one of the most important factors in an agent's success on Zillow is the number of reviews they make available to consumers.

Hundreds of thousands of consumers have posted reviews on Zillow. Now we are making those available on partner sites via the Zillow Agent Reviews API.

Both the RESO Web API and the Bridge native syntax can be used. There are two resources available: Reviewees (all agents who have reviews), and Reviews (all the available reviews).

The Reviews resource returns 10 most recent reviews for each respective reviewee. To return reviews for a team profile, start by locating the TeamLeadAccountId in the Reviews resource.

The Replication endpoint is not available for the Agent Reviews API.

Examples

Return all agentshttps://api.bridgedataoutput.com/api/v2/OData/reviews/RevieweesReturn all reviewshttps://api.bridgedataoutput.com/api/v2/OData/reviews/ReviewsReturn all reviews for a specific agent using their primary keyhttps://api.bridgedataoutput.com/api/v2/OData/reviews/Reviews?$filter=RevieweeKey eq 'f0406f76c3c46b10494c29dc'Locate data for a specific agent by email, and include all available reviewshttps://api.bridgedataoutput.com/api/v2/OData/reviews/Reviewees?$filter=RevieweeEmail eq 'agent@domain.com'&$expand=Reviews

Glossary of common terminology

|  |  |
| --- | --- |
| Term | Description |
| Application | A representation of a single product that consumes data via the API. Each application has a profile, a set of access tokens and a set of data approvals. |
| API Key | Also called an API Token, it’s an authorization code sent with an API request that identifies the user, and also what data they are allowed to see. |
| API Platform | The system that an MLS will leverage to manage and monitor data distribution, and provide software developers with various APIs to get data. |
| Authentication | Identifying the user of the API. Common techniques include API Keys and OAuth Tokens. |
| Authorization | The data an authenticated user of the API is allowed to access and query. |
| CDN | Content Delivery Network - a network of servers used to store data to provide high availability and performance. |
| Data License | A legal agreement between a data provider and a data consumer to allow access to content. |
| Dataset | A representation of either a single MLS on the platform, or an aggregation of multiple MLSs available through a single endpoint. |
| Endpoint | A web address that specifies the data or service the user is requesting. For example, going to the “/Offices” address in the Web API will return office data. |
| JSON | Similar to XML or CSV, JSON is a simple way to represent data, but it’s easy for both humans and machines to interpret. By default, the RESO Web API returns data in JSON. |
| Latency | The amount of time it takes to get data back. This may include how long it took the server to find the right data and how long data took to travel back to the recipient. |
| OAuth | A specification for token-based authentication and authorization. The RESO Web API can use this to determine who is allowed to get data and what data they are allowed to see. |
| OData | A standard specification for APIs that defines how the data should be organized and how it can be queried. The RESO Web API is based on this specification. |
| Metadata | Information about the data the user has access to, usually accessible from a dedicated API endpoint. May include things like the available fields and their possible values. |
| MLS | Multiple Listing Service. An MLS is a network for agents and brokers that grants them access to exchange property listing information and share compensation for a sale. |
| Payload | A bundle of data that is returned by an API. |

|  |  |
| --- | --- |
| Query | The request a user makes to an API endpoint, which usually contains their API key, what data they want, and any additional parameters, like filters. |
| Rate Limit | Limits on how many requests to an API can be made within a specific time frame. There may be multiple rate limits in place specific to the user or the data they are requesting. |
| Replication | The process of recreating and storing your own version of the original database and continuously updating to keep the two copies in sync. |
| Resource | A collection of data within a dataset, e.g. Properties, Members, Offices, or Open Houses. |

Bridge API Explorer

[API Dashboard](https://bridgedataoutput.com/login)

RESO Web API

/Property

GET

/{dataset}/Property

ReferenceAPI Explorer

Description

Retrieves a set of properties.

Request

Required fields

Name

Type

Description

access\_token

string

Token to identify the user or application.

dataset

string

Dataset to get the listings from.

$skip

integer

Skip the first n items

$select

string

Select properties that are explicitly specified e.g. OfficeType

$unselect

string

Filter record properties that are explicitly specified e.g. OfficeFax

$top

integer

Show only the first n items

$orderby

string

Order items by specified property value, e.g. ModificationTimestamp asc

$filter

string

Filter items by certain values

$expand

array

Expand related entities

Response

Name

Type

Description

@odata.context

string

Link to the metadata context.

@odata.id

string

Link to the OData endpoint for the specific record.

AboveGradeFinishedArea

number

Finished area within the structure that is at or above the surface of the ground.

AboveGradeFinishedAreaSource

string

The source of the measurements. This is a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

AboveGradeFinishedAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

AccessCode

string

If the property is located behind an unmanned security gate such as in a Gated Community, what is the code to gain access through the secured gate.

AccessibilityFeatures

[string]

A list or description of the accessibility features included in the sale/lease.

AdditionalParcelsDescription

string

If additional parcels are included in the sale, a list of those parcel's IDs separated by commas. Do not include the first or primary parcel number, that should be located in the Parcel Number field.

AdditionalParcelsYN

boolean

Are there more than one parcel or lot included in the sale?

AnchorsCoTenants

string

The main or most notable tenants as well as other tenants of the shopping center or mall in which the commercial property is located.

Appliances

[string]

A list of the appliances that will be included in the sale/lease of the property.

ApprovalStatus

string

When an MLS has the ability to set a listing to Draft and/or has facility to allow an agent to input, but their manager to approve the listings before publishing, this field is used for such control.

ArchitecturalStyle

[string]

A list describing the style of the structure. For example, Victorian, Ranch, Craftsman, etc.

AssociationAmenities

[string]

Amenities provided by the Home Owners Association, Mobile Park or Complex. For example Pool, Clubhouse, etc.

AssociationFee

number

A fee paid by the homeowner to the Home Owners Association which is used for the upkeep of the common area, neighborhood or other association related benefits.

AssociationFee2

number

A fee paid by the homeowner to the second of two Home Owners Associations, which is used for the upkeep of the common area, neighborhood or other association related benefits.

AssociationFee2Frequency

string

The frequency the association fee is paid. For example, Weekly, Monthly, Annually, Bi-Monthly, One Time, etc.

AssociationFeeFrequency

string

The frequency the association fee is paid. For example, Weekly, Monthly, Annually, Bi-Monthly, One Time, etc.

AssociationFeeIncludes

[string]

Services included with the association fee. For example Landscaping, Trash, Water, etc.

AssociationName

string

The name of the Home Owners Association.

AssociationName2

string

The name of the second of two Home Owners Association.

AssociationPhone

string

The phone number of the Home Owners Association. North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

AssociationPhone2

string

The phone number of the second of two Home Owners Association. North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

AssociationYN

boolean

Is there a Home Owners Association. A separate Y/N field is needed because not all associations have dues.

AttachedGarageYN

boolean

A flag indicating that the garage attached to the dwelling.

AvailabilityDate

string

The date the property will be available for possession/occupation.

Basement

[string]

A list of information and features about the basement. i.e. None/Slab, Finished, Partially Finished, Crawl Space, Dirt, Outside Entrance, Radon Mitigation

BathroomsFull

number

A room containing all 4 of the 4 elements constituting a bath, which are; Toilet, Sink, Bathtub or Shower Head. A Full Bath will typically contain four elements; Sink, Toilet, Tub and Shower Head (in tub or stall). However, some may considered a Sink, Toilet and Tub (without a shower) a Full Bath, others consider this to be a Three Quarter Bath. In the event that BathroomsThreeQuarter is not in use, this field may represent the sum of all Full and Three Quarter bathrooms.

BathroomsHalf

number

A room containing 2 of the 4 elements constituting a bath, which are; Toilet, Sink, Bathtub or Shower Head. A Half Bath will typically contain a Sink and Toilet.

BathroomsOneQuarter

number

A room containing 1 of the 4 elements constituting a bath which are; Toilet, Sink, Bathtub or Shower Head. Examples are a vanity with a sink or a WC (Water Closet, which is a room with only a toilet).

BathroomsPartial

number

The number of partial bathrooms in the property being sold/leased. When used in combination with the BathroomsFull field, this replaces (or is the sum of) all Half and One Quarter bathrooms; and in the event BathroomsThreeQuarter is not used, BathroomsFull replaces (or is the sum of) all Full and Three Quarter baths. This field should not be used in combination with the BathroomsOneQuarter or the BathroomsHalf.

BathroomsThreeQuarter

number

A room containing 3 of the 4 elements constituting a bath, which are; Toilet, Sink, Bathtub or Shower Head. A typical Three Quarter Bath will contain Sink, Toilet and Shower. Some may considered a Sink, Toilet and Tub (without a shower) a Three Quarter Bath, others consider this to be a Full Bath.

BathroomsTotalDecimal

number

A decimal representation of the total number of bathrooms.

BathroomsTotalInteger

number

The simple sum of the number of bathrooms. For example for a property with two Full Bathrooms and one Half Bathroom, the Bathrooms Total Integer will be 3. To express this example as 2.5, use the BathroomsTotalDecimal field. To express this example as 2.1, use the BathroomsTotalNotational.

BedroomsPossible

number

The sum of BedroomsTotal plus other rooms that may be used as a bedroom but are not defined as bedroom per local policy.

BedroomsTotal

number

The total number of bedrooms in the dwelling.

BelowGradeFinishedArea

number

Finished area within the structure that is below ground.

BelowGradeFinishedAreaSource

string

The source of the measurements. This is a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

BelowGradeFinishedAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

BodyType

[string]

Type of mobile home.

BridgeModificationTimestamp

string

A timestamp representing when last this record was modified in the Bridge system.

BuilderModel

string

The builders model name or number for the property.

BuilderName

string

Name of the builder of the property or builder's tract.

BuildingAreaSource

string

The source of the measurements. This is a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

BuildingAreaTotal

number

Total area of the structure. Includes both finished and unfinished areas.

BuildingAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

BuildingFeatures

[string]

Features or amenities of the building or business park.

BuildingName

string

Name of the building or business park.

BusinessName

string

Name of the business being sold.

BusinessType

[string]

The type of business being sold. Retail, Wholesale, Grocery, Food & Bev, etc.

BuyerAgencyCompensation

string

The total commission to be paid for this sale, expressed as either a percentage or a constant currency amount.

BuyerAgencyCompensationType

string

A list of types to clarify the value entered in the BuyerAgencyCompensation field. For example $, % or some other clarification of the BuyerAgencyCompensation.

BuyerAgentAOR

string

The Buyer's Agent's Board or Association of REALTORS.

BuyerAgentCellPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentDesignation

[string]

Designations and certifications acknowledging experience and expertise in various real estate sectors are awarded by NAR and each affiliated group upon completion of required courses.

BuyerAgentDirectPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentEmail

string

The email address of the Buyer's Agent.

BuyerAgentFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentFirstName

string

The first name of the buyer's agent.

BuyerAgentFullName

string

The full name of the buyer's agent. (First Middle Last)

BuyerAgentHomePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey.

BuyerAgentKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey. This is the numeric only key and used as an alternative to the BuyerAgentKey field.

BuyerAgentLastName

string

The last name of the buyer's agent.

BuyerAgentMiddleName

string

The middle name of the buyer's agent.

BuyerAgentMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

BuyerAgentMobilePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentNamePrefix

string

Prefix to the name (e.g. Dr. Mr. Ms. etc.)

BuyerAgentNameSuffix

string

Suffix to the BuyerAgentLastName (e.g. Esq., Jr., III etc.)

BuyerAgentOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentOfficePhoneExt

string

The extension of the given phone number (if applicable).

BuyerAgentPager

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentPreferredPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentPreferredPhoneExt

string

The extension of the given phone number (if applicable).

BuyerAgentStateLicense

string

The license of the buyers agent. Separate multiple licenses with a comma and space.

BuyerAgentTollFreePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentURL

string

The website URI of the buyers agent.

BuyerAgentVoiceMail

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerAgentVoiceMailExt

string

The extension of the given phone number (if applicable).

BuyerFinancing

[string]

A list of options that describe the type of financing used. This field is used when setting a listing to Closed. i.e. cash, FHA loan, etc.

BuyerOfficeAOR

string

The Buyer's Office's Board or Association of REALTORS.

BuyerOfficeEmail

string

The email address of the Buyer's Office.

BuyerOfficeFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerOfficeKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey.

BuyerOfficeKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey. This is the numeric only key and used as an alternative to the BuyerOfficeKey field.

BuyerOfficeMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

BuyerOfficeName

string

The legal name of the brokerage representing the buyer.

BuyerOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

BuyerOfficePhoneExt

string

The extension of the given phone number (if applicable).

BuyerOfficeURL

string

The website URI for the buyers office.

BuyerTeamKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Teams resource's TeamKey.

BuyerTeamKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Teams resource's TeamKey. This is the numeric only key and used as an alternative to the BuyerTeamKey field.

BuyerTeamName

string

The name of the team representing the buyer.

CableTvExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

CancelationDate

string

Date the listing contract between the seller and listing agent was cancelled. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

CapRate

number

Cap Rate is equivalent to the return on investment you would receive if you pay cash for a property. The ratio between the net operating income produced by an asset and its capital cost (the original price paid to buy the asset) or alternatively its current market value.

CarportSpaces

number

The number of carport spaces included in the sale.

CarportYN

boolean

A flag indicating that the listing has a car port. This flag may be T/F, Y/N or other true, false or unknown indicator. As with all flags, the field may be null.

CarrierRoute

string

The group of addresses to which the USPS assigns the same code to aid in mail delivery. For the USPS, these codes are 9 digits: 5 numbers for the ZIP Code, one letter for the carrier route type, and 3 numbers for the carrier route number.

City

string

The city in listing address.

CityRegion

string

A sub-section or area of a defined city. Examples would be SOHO in New York, NY, Ironbound in Newark, NJ or Inside the Beltway.

CloseDate

string

With for-sale listings, the date the purchase agreement was fulfilled. With lease listings, the date the requirements were fulfilled, such as contract and/or deposit. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

ClosePrice

number

The amount of money paid by the purchaser to the seller for the property under the agreement.

CoBuyerAgentAOR

string

The Co Buyer's Agent's Board or Association of REALTORS.

CoBuyerAgentCellPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentDesignation

[string]

Designations and certifications acknowledging experience and expertise in various real estate sectors are awarded by NAR and each affiliated group upon completion of required courses.

CoBuyerAgentDirectPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentEmail

string

The email address of the Buyer's Co Agent.

CoBuyerAgentFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentFirstName

string

The first name of the buyer's co-agent.

CoBuyerAgentFullName

string

The full name of the buyer's co-agent. (First Middle Last)

CoBuyerAgentHomePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey.

CoBuyerAgentKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey. This is the numeric only key and used as an alternative to the CoBuyerAgentKey field.

CoBuyerAgentLastName

string

The last name of the buyer's co-agent.

CoBuyerAgentMiddleName

string

The middle name of the buyer's co-agent.

CoBuyerAgentMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

CoBuyerAgentMobilePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentNamePrefix

string

Prefix to the name (e.g. Dr. Mr. Ms. etc.)

CoBuyerAgentNameSuffix

string

Suffix to the CoBuyerAgentLastName (e.g. Esq., Jr., III etc.)

CoBuyerAgentOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentOfficePhoneExt

string

The extension of the given phone number (if applicable).

CoBuyerAgentPager

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentPreferredPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentPreferredPhoneExt

string

The extension of the given phone number (if applicable).

CoBuyerAgentStateLicense

string

The license of the co-buyers agent. Separate multiple licenses with a comma and space.

CoBuyerAgentTollFreePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentURL

string

The website URI of the co-buyers agent.

CoBuyerAgentVoiceMail

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerAgentVoiceMailExt

string

The extension of the given phone number (if applicable).

CoBuyerOfficeAOR

string

The Co Buyer's Office's Board or Association of REALTORS.

CoBuyerOfficeEmail

string

The email address of the Buyer's Co Office.

CoBuyerOfficeFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerOfficeKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey.

CoBuyerOfficeKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey. This is the numeric only key and used as an alternative to the CoBuyerOfficeKey field.

CoBuyerOfficeMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

CoBuyerOfficeName

string

The legal name of the brokerage co-representing the buyer.

CoBuyerOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoBuyerOfficePhoneExt

string

The extension of the given phone number (if applicable).

CoBuyerOfficeURL

string

The website URI for the co-buyers office.

CoListAgentAOR

string

The Co Listing Agent's Board or Association of REALTORS.

CoListAgentCellPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentDesignation

[string]

Designations and certifications acknowledging experience and expertise in various real estate sectors are awarded by NAR and each affiliated group upon completion of required courses.

CoListAgentDirectPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentEmail

string

The email address of the Co Listing Agent.

CoListAgentFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentFirstName

string

The first name of the co-listing agent.

CoListAgentFullName

string

The full name of the co-listing agent. (First Middle Last)

CoListAgentHomePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey.

CoListAgentKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Member resource's MemberKey. This is the numeric only key and used as an alternative to the CoListAgentKey field.

CoListAgentLastName

string

The last name of the co-listing agent.

CoListAgentMiddleName

string

The middle name of the co-listing agent.

CoListAgentMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

CoListAgentMobilePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentNamePrefix

string

Prefix to the name (e.g. Dr. Mr. Ms. etc.)

CoListAgentNameSuffix

string

Suffix to the CoListAgentLastName (e.g. Esq., Jr., III etc.)

CoListAgentOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentOfficePhoneExt

string

The extension of the given phone number (if applicable).

CoListAgentPager

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentPreferredPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentPreferredPhoneExt

string

The extension of the given phone number (if applicable).

CoListAgentStateLicense

string

The license of the co-listing agent. Separate multiple licenses with a comma and space.

CoListAgentTollFreePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentURL

string

The website URI of the co-listing agent.

CoListAgentVoiceMail

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListAgentVoiceMailExt

string

The extension of the given phone number (if applicable).

CoListOfficeAOR

string

The Co Listing Office's Board or Association of REALTORS.

CoListOfficeEmail

string

The email address of the Co Listing Office.

CoListOfficeFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListOfficeKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey.

CoListOfficeKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey. This is the numeric only key and used as an alternative to the CoListOfficeKey field.

CoListOfficeMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

CoListOfficeName

string

The legal name of the brokerage co-representing the seller.

CoListOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

CoListOfficePhoneExt

string

The extension of the given phone number (if applicable).

CoListOfficeURL

string

The website URI for the co-listing office.

CommonInterest

string

Common Interest is a type of ownership in a property that is composed of an individual lot or unit and a share of the ownership or use of common areas. A Common Interest Development (CID) is usually governed by a recorded set of Covenants, Conditions & Restrictions (CC&Rs).

CommonWalls

[string]

A multi select list with options like 1 Common Wall, 2 Common Walls, No Common Walls, No One Above, No One Below. Implementation should include rules preventing illogical selection combinations and to ensure consistency with the Property Attached Y/N field.

CommunityFeatures

[string]

A list of features related to, or available within, the community.

Concessions

string

Are there concessions included in the sales agreement? Yes, No or Call Listing Agent

ConcessionsAmount

number

The dollar amount of the concessions. If the concessions are made by the seller, some may subtract this value from the sales price as a means of calculating their own true price. If concessions are made by the buyer, some may add this amount to the sale price to create their own true price. Concessions made by both buyer and seller should be subtracted from each other providing a net value. Details of this calculation should be added to the Concessions Comments field.

ConcessionsComments

string

Comments describing the concessions made by the buyer or the seller.

ConstructionMaterials

[string]

A list of the materials that were used in the construction of the property.

ContinentRegion

string

A sub-section or area of a continent. Examples would be Southern Europe or Scandinavia.

Contingency

string

A list of contingencies that must be satisfied in order to complete the transaction.

ContingentDate

string

The date an offer was made with a contingency. The Listing remains On Market. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

ContractStatusChangeDate

string

The date of the listings contractual status change. This is not necessarily the time the agent made the change in the MLS system, but rather the date of the contractual change.

Cooling

[string]

A list describing the cooling or air conditioning features of the property.

CoolingYN

boolean

The property has cooling or Air Conditioning.

Coordinates

[array]

Geographical coordinates (longitude, latitude)

CopyrightNotice

string

Notice of the legal rights of the owner of the information or data.

Country

string

The country abbreviation in a postal address.

CountryRegion

string

A sub-section or area of a defined country. Examples would be Napa Valley in the US, or the Amalfi Coast in Italy.

CountyOrParish

string

The County, Parish or other regional authority

CoveredSpaces

number

The total number of garage and carport spaces.

CropsIncludedYN

boolean

Are crops included in the sale of the property.

CrossStreet

string

Nearest cross streets to the property. This field is in addition to, and independent of, the driving directions field.

CultivatedArea

number

Measurement or percentage of the property that has been cultivated.

CumulativeDaysOnMarket

number

The number of days the property is on market, as defined by the MLS business rules.

CurrentFinancing

[string]

A list of options that describe the type of financing that the seller currently has in place for the property being sold. i.e. cash, assumable, FHA loan, etc.

CurrentUse

[string]

A list of the type(s) of current use of the property. The current use of the property is an important factor in understanding the overall condition of the land and determining it's appropriateness for intended use.

DaysOnMarket

number

The number of days the listing is on market, as defined by the MLS business rules.

DevelopmentStatus

[string]

A list of the Development Status of the property. The developmental status of land is an important factor in selling, purchasing and developing of land properties.

DirectionFaces

string

The compass direction that the main entrance to the building faces. For example, North, South, East, West, South-West, etc. It may also be known as the building exposure.

Directions

string

Driving directions to the property.

Disclaimer

string

Text that serves as the negation or limitation of the rights under a warranty given by a seller to a buyer.

Disclosures

[string]

Legal or pertinent information that should be disclosed to potential buyer's agents.

DistanceToBusComments

string

A textual description of the distance to local bus stops.

DistanceToBusNumeric

number

Numeric distance from the property to the nearest bus stop.

DistanceToBusUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToElectricComments

string

If the property does not currently have electrical utility, is service available and if so, what is the distance.

DistanceToElectricNumeric

number

Numeric distance from the property to the electrical utility.

DistanceToElectricUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToFreewayComments

string

A textual description of the distance to freeways.

DistanceToFreewayNumeric

number

Numeric distance from the property to the nearest freeway.

DistanceToFreewayUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToGasComments

string

If the property does not currently have natural gas utility, is service available and if so, what is the distance.

DistanceToGasNumeric

number

Numeric distance from the property to the gas utility.

DistanceToGasUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToPhoneServiceComments

string

If the property does not currently have phone service, is service available and if so, what is the distance.

DistanceToPhoneServiceNumeric

number

Numeric distance from the property to the phone utility.

DistanceToPhoneServiceUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToPlaceofWorshipComments

string

A textual description of the distance to local places of worship.

DistanceToPlaceofWorshipNumeric

number

Numeric distance from the property to the nearest place of worship.

DistanceToPlaceofWorshipUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToSchoolBusComments

string

Distance from the property to the nearest school bus pickup point.

DistanceToSchoolBusNumeric

number

Numeric distance from the property to the nearest school bus pickup point.

DistanceToSchoolBusUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToSchoolsComments

string

A textual description of the distance to local schools.

DistanceToSchoolsNumeric

number

Numeric distance from the property to the nearest school.

DistanceToSchoolsUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToSewerComments

string

If the property does not currently have sewer or septic, is sewer service available and if so, what is the distance.

DistanceToSewerNumeric

number

Numeric distance from the property to the sewer utility.

DistanceToSewerUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToShoppingComments

string

A description of the distance to primary shopping sources such as groceries, gasoline, clothing or department stores.

DistanceToShoppingNumeric

number

Numeric distance from the property to the nearest shopping.

DistanceToShoppingUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToStreetComments

string

If the property does not have a maintained road or street adjacent to the lot, what are the conditions of access and distance to a maintained road.

DistanceToStreetNumeric

number

Numeric distance from the property to the street.

DistanceToStreetUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DistanceToWaterComments

string

If the property does not currently have water utility, is service available and if so, what is the distance.

DistanceToWaterNumeric

number

Numeric distance from the property to the water utility.

DistanceToWaterUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

DocumentsAvailable

[string]

A list of the Documents available for the property. Knowing what documents are available for the property is valuable information.

DocumentsChangeTimestamp

string

System generated timestamp of when the last update or change to the documents for this listing was made.

DocumentsCount

number

The total number of documents or supplements included with the listings.

DOH1

string

Department of Housing decal number for the mobile or manufactured home. For the first or only unit/section use DOH 1 over DOH 2 or 3.

DOH2

string

Department of Housing decal number for the mobile or manufactured home. For two units/sections use DOH 1 and 2 over DOH 3.

DOH3

string

Department of Housing decal number for the mobile or manufactured home. For two units/sections use DOH 1 and 2 over DOH 3.

DoorFeatures

[string]

A list of features or description of the doors included in the sale/lease.

DualVariableCompensationYN

boolean

A commission arrangement in which the seller agrees to pay a specified commission to the listing broker if the property is sold through the efforts of a cooperating broker, but the seller pays the Listing broker a different commission amount if the sale occurs if:1) there is no cooperating broker involved or 2) due to the efforts of the seller directly.

Electric

[string]

A list of electric-service related features of the property (e.g. 110 Volt, 3 Phase, 220 Volt, RV Hookup). Note: the previous "Electric" field was renamed to DistanceToElectricComments

ElectricExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

ElectricOnPropertyYN

boolean

Does the property currently have electrical utility available on the property.

ElementarySchool

string

The name of the primary school having a catchment area that includes the associated property.

ElementarySchoolDistrict

string

The name of the elementary school district having a catchment area that includes the associated property.

Elevation

number

The elevation of the property in relation to sea level. Use the Elevation Units field to communicate the unit of measurement. i.e. Feet or Meters.

ElevationUnits

string

A pick list of the unit of measurement used in the Elevation field. i.e. Feet, Meters.

EntryLevel

number

A numeric field that describes the level within the structure, SFR or a unit in a building, where the main entry to the dwelling is located. When a unit has one floor it is implicit that this is also the level of the unit itself.

EntryLocation

string

A description of the main entry way to the property. i.e. Elevator, Ground Level w/ Steps, Ground Level w/o Steps, Mid Level, Top Level, etc.

Exclusions

string

Elements of the property that will not be included in the sale. i.e. Chandeliers will be removed prior to close.

ExistingLeaseType

[string]

Information about the status of the existing lease on the property. i.e. Net, NNN, NN, Gross, Absolute Net, Escalation Clause, Ground Lease, etc.

ExpirationDate

string

The date when the listing agreement will expire. This is the date entered by the agent reflecting when the change occurred, or will occur, contractually, not a timestamp of when the change was made in the MLS. The expiration date of listings, prior to their expiration, cancellation, sale or lease, is confidential information and should be restricted to the agent and their managers, partners or broker.

ExteriorFeatures

[string]

A list of features or description of the exterior of the property included in the sale/lease.

FarmCreditServiceInclYN

boolean

Specifies whether or not Farm Credit Service shares are included in the price of the property.

FarmLandAreaSource

string

The source of the measurements. This may be a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc. This field applies to all farm area fields (Cultivated, Pasture, Range, Wooded)

FarmLandAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc. This field applies to all farm area fields (Cultivated, Pasture, Range, Wooded)

Fencing

[string]

A list of types of fencing found at the property being sold.

FinancialDataSource

[string]

The source of the Rental information. For example Accountant, Owner, etc.

FireplaceFeatures

[string]

A list of features or description of the fireplace(s) included in the sale/lease.

FireplacesTotal

number

The total number of fireplaces included in the property.

FireplaceYN

boolean

Does the property include a fireplace.

Flooring

[string]

A list of the type(s) of flooring found within the property.

FoundationArea

number

The area or dimensions of the footprint of the structure on the lot.

FoundationDetails

[string]

A list of the type(s) of foundation on which the property sits.

FrontageLength

string

Textual description of the length of the frontages selected in the Frontage Type field.

FrontageType

[string]

Pick list of types of frontage. i.e. Oceanfront, Lakefront, Golf course, etc. Information about roads or road frontage should be located in the Road Frontage Type and Road Surface Type fields.

FuelExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

Furnished

string

The property being leased is furnished, unfurnished or partially furnished.

FurnitureReplacementExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

GarageSpaces

number

The number of spaces in the garage(s).

GarageYN

boolean

A flag indicating that the listing has a garage. This flag may be T/F, Y/N or other true, false or unknown indicator. As with all flags, the field may be null.

GardenerExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

GardnerExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

Gas

[string]

A list of gas-service related features of the property (e.g. Natural Gas, Private LP Tank, None). Note: the previous "Gas" field was renamed to DistanceToGasComments

GrazingPermitsBlmYN

boolean

Specifies whether or not the property owner has grazing permits from the Bureau of Land Management.

GrazingPermitsForestServiceYN

boolean

Specifies whether or not the property owner has grazing permits from the Forestry Service.

GrazingPermitsPrivateYN

boolean

Specifies whether or not the property owner has private grazing permits.

GreenBuildingVerificationType

[string]

The name of the verification or certification awarded to a new or pre-existing residential or commercial structure. For example: LEED, Energy Star, ICC-700. In cases where more than one certification have been awarded, leverage multiple iterations of the green verification fields via the repeating element method.

GreenEnergyEfficient

[string]

Pick list of general green attributes such as energy efficient doors, or appliances without naming specific elements with ratings that may wane over time.

GreenEnergyGeneration

[string]

Methods of generating power that are included in the sale or lease.

GreenIndoorAirQuality

[string]

Pick list of indoor air quality measures without naming specific elements with ratings that may wane over time.

GreenLocation

[string]

Pick list describing efficiencies involved with the property's location such as walkability or transportation proximity without naming specific elements with ratings that may wane over time.

GreenSustainability

[string]

Pick list of sustainable elements used in the construction of the structure without naming specific elements with ratings that may wane over time.

GreenWaterConservation

[string]

Pick list of general water conserving attributes of the property such as landscaping or reclamation without naming specific elements with ratings that may wane over time.

GrossIncome

number

The actual current income from rent and all other revenue generating sources.

GrossScheduledIncome

number

The maximum amount of annual rent collected if the property were 100% occupied all year and all tenants paid their rent.

HabitableResidenceYN

boolean

Does the property include a structure that can be lived in.

Heating

[string]

A list describing the heating features of the property.

HeatingYN

boolean

The property has heating.

HighSchool

string

The name of the high school having a catchment area that includes the associated property.

HighSchoolDistrict

string

The name of the high school district having a catchment area that includes the associated property. When only one school district is used, this field should be used over the Junior or Elementary Districts.

HomeWarrantyYN

boolean

Is a home warranty included in the sale of the property? Single select.

HorseAmenities

[string]

A list of horse amenities on the lot or in the community.

HorseYN

boolean

The Property is allowed to raise horses.

HoursDaysOfOperation

[string]

A simplified enumerated list of the days and hours of operation of the business being sold. i.e. Open 24 Hours or Open 7 Days. For more detailed descriptions use the HoursDaysofOperationDescription field.

HoursDaysOfOperationDescription

string

A detailed description of the hours and days the business being sold is open for business. For a specific list of simplified times the business is open, use the HoursDaysofOperation enumerated field.

IDXParticipationYN

boolean

Should this listing participate in IDX.

Inclusions

string

Portable elements of the property that will be included in the sale.

IncomeIncludes

[string]

A list of income sources included in the GrossScheduledIncome and GrossIncome. i.e. Laundry, Parking, Recreation, Storage, etc.

InsuranceExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

InteriorFeatures

[string]

A list of features or description of the interior of the property included in the sale/lease.

InternetAddressDisplayYN

boolean

A yes/no field that states the seller has allowed the listing address to be displayed on Internet sites.

InternetAutomatedValuationDisplayYN

boolean

A yes/no field that states the seller allows the listing can be displayed with an AVM on Internet sites.

InternetConsumerCommentYN

boolean

A yes/no field that states the seller allows a comment or blog system to be attached to the listing on Internet sites.

InternetEntireListingDisplayYN

boolean

A yes/no field that states the seller has allowed the listing to be displayed on Internet sites.

IrrigationSource

[string]

The source which the property receives its water for irrigation.

IrrigationWaterRightsAcres

number

The number of acres allowed under the property's water rights.

IrrigationWaterRightsYN

boolean

Does the property include water rights for irrigation? A Boolean or Yes / No field.

LaborInformation

[string]

Information about labor laws that are applicable to the business being sold. i.e. Union, Non-Union, Employee License Required.

LandLeaseAmount

number

When the land is not included in the sale, but is leased, the amount of the lease. This is the Space Rent for Mobile homes in a Park.

LandLeaseAmountFrequency

string

When the land is not included in the sale, but is leased, the frequency the Land Lease Fee is paid.

LandLeaseExpirationDate

string

When the land is not included in the sale, but is leased, the expiration date of the Land Lease.

LandLeaseYN

boolean

The land is not included in the sale and a lease exists.

Latitude

number

The geographic latitude of some reference point on the property, specified in degrees and decimal parts. Positive numbers must not include the plus symbol.

LaundryFeatures

[string]

Add this pick list of features and locations where the laundry is located in the property being sold. i.e. Gas Dryer Hookup, In Kitchen, In Garage, etc. CRMLS sees over 50% utilization of this field which has a dozen enumerations making it too long to fold into other fields such as rooms or Interior Features.

LeasableArea

number

The area that may be leased within the commercial property.

LeasableAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

LeaseAmount

number

The amount of any lease the business pays for it's current location.

LeaseAmountFrequency

string

The frequency of the LeaseAmount is paid. Monthly, weekly, annual, etc.

LeaseAssignableYN

boolean

Can the lease at the business' current location be assigned to another party.

LeaseConsideredYN

boolean

Will the seller consider leasing the property instead of selling? Single select.

LeaseExpiration

string

The expiration date of the lease for the business' current location.

LeaseRenewalCompensation

[string]

A list of compensations other than the original Selling Office Compensation. i.e. Compensation Paid on Renewal, Compensation Paid on Tennant Purchase, No Renewal Commission, Call Listing Office, etc.

LeaseRenewalOptionYN

boolean

Is there an option to renew the lease at the business' current location.

LeaseTerm

string

A pick list of lengths that represent the length of the lease. i.e. Weekly, Month to Month, 6 Month Lease, 12 Month Lease, 24 Month Lease.

Levels

[string]

The number of levels in the property being sold. For example, One Level, Two Levels, Split Level, Three or More Levels, Multi Level, Loft. A discreet horizontal plane of interior living space (excluding basements).

License1

string

License number of the mobile or manufactured home. Also known as the Department of Housing label/insignia number. For the first or only unit/section use License 1 over License 2 or 3.

License2

string

License number of the mobile or manufactured home. Also known as the Department of Housing label/insignia number. For two units/sections use License 1 and 2 over License 3.

License3

string

License number of the mobile or manufactured home. Also known as the Department of Housing label/insignia number. For two units/sections use License 1 and 2 over License 3.

LicensesExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

ListAgentAOR

string

The Listing Agent's Board or Association of REALTORS.

ListAgentCellPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentDesignation

[string]

Designations and certifications acknowledging experience and expertise in various real estate sectors are awarded by NAR and each affiliated group upon completion of required courses.

ListAgentDirectPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentEmail

string

The email address of the Listing Agent.

ListAgentFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentFirstName

string

The first name of the listing agent.

ListAgentFullName

string

The full name of the listing agent. (First Middle Last)

ListAgentHomePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentKey

string

A system unique identifier. Specifically, in aggregation systems, the ListAgentKey is the system unique identifier from the system that the record was retrieved. This may be identical to the related xxxId. This is a foreign key relating to the Member resource's MemberKey.

ListAgentKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the ListAgentKey is the system unique identifier from the system that the record was retrieved. This may be identical to the related xxxId. This is a foreign key relating to the Member resource's MemberKey. This is the numeric only key and used as an alternative to the ListAgentKey field.

ListAgentLastName

string

The last name of the listing agent.

ListAgentMiddleName

string

The middle name of the listing agent.

ListAgentMlsId

string

The local, well-known identifier for the member. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

ListAgentMobilePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentNamePrefix

string

Prefix to the name (e.g. Dr. Mr. Ms. etc.)

ListAgentNameSuffix

string

Suffix to the ListAgentLastName (e.g. Esq., Jr., III etc.)

ListAgentOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentOfficePhoneExt

string

The extension of the given phone number (if applicable).

ListAgentPager

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentPreferredPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentPreferredPhoneExt

string

The extension of the given phone number (if applicable).

ListAgentStateLicense

string

The license of the listing agent. Separate multiple licenses with a comma and space.

ListAgentTollFreePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentURL

string

The website URI of the listing agent.

ListAgentVoiceMail

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListAgentVoiceMailExt

string

The extension of the given phone number (if applicable).

ListAOR

string

The responsible Board or Association of REALTORS for this listing.

ListingAgreement

string

The nature of the agreement between the seller and the listing agent. Examples are Exclusive Agency, Open Listing, etc.

ListingContractDate

string

The effective date of the agreement between the seller and the seller's broker. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

ListingId

string

The well known identifier for the listing. The value may be identical to that of the Listing Key, but the Listing ID is intended to be the value used by a human to retrieve the information about a specific listing. In a multiple originating system or a merged system, this value may not be unique and may require the use of the provider system to create a synthetic unique value.

ListingKey

string

A unique identifier for this record from the immediate source. This is a string that can include URI or other forms. Alternatively use the ListingKeyNumeric for a numeric only key field. This is the local key of the system. When records are received from other systems, a local key is commonly applied. If conveying the original keys from the source or originating systems, see SourceSystemKey and OriginatingSystemKey.

ListingKeyNumeric

number

A unique identifier for this record from the immediate source. This is the numeric only key and used as an alternative to the ListingKey fields. This is the local key of the system. When records are received from other systems, a local key is commonly applied. If conveying the original keys from the source or originating systems, see SourceSystemKey and OriginatingSystemKey.

ListingService

string

Defines the type or level of service the listing member will be providing to the selling home owner. This will typically be a single selection. Examples include Full Service, Limited Service or Entry Only.

ListingTerms

[string]

Terms of the listing such as Lien Release, Subject to Court Approval or Owner Will Carry. Also may include options that describe the financing terms that are acceptable to the seller, i.e. cash, assumable, FHA loan, etc.

ListOfficeAOR

string

The Listing Office's Board or Association of REALTORS.

ListOfficeEmail

string

The email address of the Listing Office.

ListOfficeFax

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListOfficeKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey.

ListOfficeKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Office resource's OfficeKey. This is the numeric only key and used as an alternative to the ListOfficeKey field.

ListOfficeMlsId

string

The local, well-known identifier. This value may not be unique, specifically in the case of aggregation systems, this value should be the identifier from the original system.

ListOfficeName

string

The legal name of the brokerage representing the seller.

ListOfficePhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ListOfficePhoneExt

string

The extension of the given phone number (if applicable).

ListOfficeURL

string

The website URI for the listing office.

ListPrice

number

The current price of the property as determined by the seller and the seller's broker. For auctions this is the minimum or reserve price.

ListPriceLow

number

The lower price used for Value Range Pricing. The List Price must be greater than or equal to the ListPriceLow.

ListTeamKey

string

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Teams resource's TeamKey.

ListTeamKeyNumeric

number

A system unique identifier. Specifically, in aggregation systems, the Key is the system unique identifier from the system that the record was just retrieved. This may be identical to the related xxxId identifier, but the key is guaranteed unique for this record set. This is a foreign key relating to the Teams resource's TeamKey. This is the numeric only key and used as an alternative to the ListTeamKey field.

ListTeamName

string

The name of the team representing the seller.

LivingArea

number

The total livable area within the structure.

LivingAreaSource

string

The source of the measurements. This is a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

LivingAreaUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

LockBoxLocation

string

A field describing the location of the lock box.

LockBoxSerialNumber

string

The serial number of the lockbox placed on the property.

LockBoxType

[string]

A field describing the type of lock box.

Longitude

number

The geographic longitude of some reference point on the property, specified in degrees and decimal parts. Positive numbers must not include the plus symbol.

LotDimensionsSource

string

The source of the measurements. This may be a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

LotFeatures

[string]

A list of features or description of the lot included in the sale/lease.

LotSizeAcres

number

The total Acres of the lot. This field is related to the Lot Size Area and Lot Size Units and must be in sync with the values represented in those fields. Lot Size Source also applies to this field when used.

LotSizeArea

number

The total area of the lot. See Lot Size Units for the units of measurement (Square Feet, Square Meters, Acres, etc.).

LotSizeDimensions

string

The dimensions of the lot minimally represented as length and width (i.e. 250 x 180) or a measurement of all sides of the polygon representing the property lines of the property. i.e. 30 x 50 x 120 x 60 x 22.

LotSizeSource

string

The source of the measurements. This may be a pick list of options showing the source of the measurement. i.e. Agent, Assessor, Estimate, etc.

LotSizeSquareFeet

number

The total square footage of the lot. This field is related to the Lot Size Area and Lot Size Units and must be in sync with the values represented in those fields. Lot Size Source also applies to this field when used.

LotSizeUnits

string

A pick list of the unit of measurement for the area. i.e. Square Feet, Square Meters, Acres, etc.

MainLevelBathrooms

number

The number of bathrooms located on the main or entry level of the property.

MainLevelBedrooms

number

The number of bedrooms located on the main or entry level of the property.

MaintenanceExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

MajorChangeTimestamp

string

Timestamp of the last major change on the listing (see also MajorChangeType).

MajorChangeType

string

Description of the last major change on the listing, i.e. “price reduction”, “back on market”, etc. May be used to display on a summary view of listing results to quickly identify listings that have had major changes recently.

Make

string

Make of the mobile or manufactured home.

MaloneId

number

The malone ID associated with a property listing, given its address information.

ManagerExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations. This is for an individual manager. Use ProfessionalManagementExpense for a management company.

MapCoordinate

string

A map coordinate for the property, as determined by local custom. This is not necessarily the same as the geographic coordinate but may depend on the coordinate system used by whatever mapping service is customarily used by the listing service.

MapCoordinateSource

string

Name of the map or map book publisher.

MapURL

string

URI to a map of the property.

Media

[array]

An object containing media information such as id, url and type.

Media.APIModificationTimestamp

string

A timestamp representing when last this listing was modified in our system.

Media.BridgeModificationTimestamp

string

A timestamp representing when last this record was modified in the Bridge system.

Media.ChangedByMemberID

string

ID of the user, agent, member, etc., that uploaded the media this record refers to.

Media.ChangedByMemberKey

string

The primary key of the member who uploaded the media this record refers to. This is a foreign key relating to the Member resource's MemberKey.

Media.ChangedByMemberKeyNumeric

number

The primary key of the member who uploaded the media this record refers to. This is a foreign key relating to the Member resource's MemberKey. This is the numeric only key and used as an alternative to the ChangedByMemberKey field.

Media.ClassName

string

The class or table of the listing or other record the media. Residential, Lease, Agent, Office, Contact, etc.

Media.Group

string

A placeholder for media classification such as elevation, exterior, interior, community, view, plan, plat. The purpose is to allow media items to be grouped.

Media.id

string

The unique media ID

Media.ImageHeight

number

The height of the image expressed in pixels.

Media.ImageOf

string

When the media is an image, a list of possible matches such as kitchen, bathroom, front of structure, etc. This field may be used to identify a required image under association or MLS rules.

Media.ImageSizeDescription

string

A text description of the size of the image. i.e. Small, Thumbnail, Medium, Large, X-Large. The largest image must be described as "Largest". Thumbnail must also be included. Pick List will remain open/extendable.

Media.ImageWidth

number

The width of the image expressed in pixels.

Media.LongDescription

string

The full robust description of the object.

Media.MediaCategory

string

Category describing the , Photos, Documents, Video, Unbranded Virtual Tour, Branded Virtual Tour, Floor Plan, Logo

Media.MediaHTML

string

The JavaScript or other method to embed a video, image, virtual tour or other media.

Media.MediaKey

string

A unique identifier for this record from the immediate source. This may be a number, or string that can include URI or other forms. This is the system you are connecting to and not necessarily the original source of the record.

Media.MediaKeyNumeric

number

A unique identifier for this record from the immediate source. This may be a number, or string that can include URI or other forms. This is the system you are connecting to and not necessarily the original source of the record. This is the numeric only key and used as an alternative to the MediaKey field.

Media.MediaModificationTimestamp

string

This timestamp is updated when a change to the object has been made, which may differ from a change to the Media Resource.

Media.MediaObjectID

string

ID of the image, supplement or other object specified by the given media record.

Media.MediaStatus

string

The status of the media item referenced by this record. (Updated, Deleted, etc.,\_

Media.MediaType

string

Media Types as defined by IANA. http://www.iana.org/assignments/media-types/index.html

Media.MediaURL

string

The URI to the media file referenced by this record.

Media.MimeType

string

Media Types as defined by IANA. http://www.iana.org/assignments/media-types/index.html

Media.ModificationTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the date/time the media record was last modified.

Media.Order

number

Only a positive integer including zero. Element zero is the primary photo per RETS convention.

Media.OriginatingSystemID

string

The RESO OUID's OrganizationUniqueId of the Originating record provider. The Originating system is the system with authoritative control over the record. For example; the name of the MLS where the listing was input. In cases where the Originating system was not where the record originated (the authoritative system), see the Originating System fields.

Media.OriginatingSystemMediaKey

string

Unique identifier from the originating system which is commonly a key to that system. In the case where data is passed through more than one system, this is the originating system key. This is a foreign key relating to the system where this record was originated.

Media.OriginatingSystemName

string

The name of the originating record provider. Most commonly the name of the MLS. The place where the listing is originally input by the member. The legal name of the company. To be used for display.

Media.Permission

[string]

Public, Private, IDX, VOW, Office Only, Firm Only, Agent Only,

Media.PreferredPhotoYN

boolean

When set to true, the media record in question is the preferred photo. This will typically mean the photo to be shown when only one of the photos is to be displayed.

Media.ResizeMediaURL

string

The URI to a resizable version of the media file referenced by this record.

Media.ResourceName

string

The resource or table of the listing or other record the media relates to. i.e. Property, Member, Office, etc.

Media.ResourceRecordID

string

The well known identifier of the related record from the source resource. The value may be identical to that of the Listing Key, but the Listing ID is intended to be the value used by a human to retrieve the information about a specific listing. In a multiple originating system or a merged system, this value may not be unique and may require the use of the provider system to create a synthetic unique value.

Media.ResourceRecordKey

string

The primary key of the related record from the source resource. For example the ListingKey, AgentKey, OfficeKey, TeamKey, etc. This is the system you are connecting to and not necessarily the original source of the record. This is a foreign key from the resrouce selected in the ResrouceName field.

Media.ResourceRecordKeyNumeric

number

The primary key of the related record from the source resource. For example the ListingKey, AgentKey, OfficeKey, TeamKey, etc. This is the system you are connecting to and not necessarily the original source of the record. This is a foreign key from the resource selected in the ResourceName field. This is the numeric only key and used as an alternative to the ResourceRecordKey field.

Media.ShortDescription

string

The short text given to summarize the object. Commonly used as the short description displayed under a photo.

Media.SourceSystemID

string

The RESO OUID's OrganizationUniqueId of the Source record provider. The source system is the system from which the record was directly received. In cases where the source system was not where the record originated (the authoritative system), see the Originating System fields.

Media.SourceSystemMediaKey

string

The system key, a unique record identifier, from the Source System. The Source System is the system from which the record was directly received. In cases where the Source System was not where the record originated (the authoritative system), see the Originating System fields.

Media.SourceSystemName

string

The name of the immediate record provider. The system from which the record was directly received. The legal name of the company.

MiddleOrJuniorSchool

string

The name of the junior or middle school having a catchment area that includes the associated property.

MiddleOrJuniorSchoolDistrict

string

The name of the junior or middle school district having a catchment area that includes the associated property.

MLSAreaMajor

string

The major marketing area name, as defined by the MLS or other non-governmental organization. If there is only one MLS Area in use, it must be the MLSAreaMajor.

MLSAreaMinor

string

The minor/sub marketing area name, as defined by the MLS or other non-governmental organization. If there is only one MLS Area in use, it must be the MLSAreaMajor.

MlsStatus

string

Local or regional status that are well known by business users. Each MlsStatus must map to a single StandardStatus. Multiple MlsStatus may map to a single StandardStatus.

MobileDimUnits

string

A pick list of the unit linear measurement. i.e. feed, meters, yards, kilometers, miles, etc.

MobileHomeRemainsYN

boolean

Is the mobile home to remain and be included in the sale of the property.

MobileLength

number

Length of the mobile/manufactured home.

MobileWidth

number

Width of the mobile/manufactured home.

Model

string

Model of the mobile or manufactured home.

ModificationTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the date/time the listing was last modified.

NetOperatingIncome

number

Net operating income is the revenue from a property after operating expenses have been deducted, but before deducting income taxes and financing expenses (interest and Principal Payments). For example, Gross Income - Operating Expenses = Net Operating Income (NOI).

NewConstructionYN

boolean

Is the property newly constructed and has not been previously occupied?

NewTaxesExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

NumberOfBuildings

number

Total number of separate buildings included in the income property.

NumberOfFullTimeEmployees

number

The current number of individuals employed by the business on a full-time basis.

NumberOfLots

number

Total number of lots on the property or included in the sale. Land properties are often sold with multiple lots. It is important to be able to describe how many lots are in the property and not in all cases do lots have separate Parcel IDs.

NumberOfPads

number

The number of pads or spaces in the mobile home park.

NumberOfPartTimeEmployees

number

The current number of individuals employed by the business on a part-time basis.

NumberOfSeparateElectricMeters

number

Total number of separate meters on the property.

NumberOfSeparateGasMeters

number

Total number of separate meters on the property.

NumberOfSeparateWaterMeters

number

Total number of separate meters on the property.

NumberOfUnitsInCommunity

number

The total number of units in the building, complex or community. This is not the number of units being sold, but rather the size of the community in which the dwelling being sold is located.

NumberOfUnitsLeased

number

Total number of units currently under a lease agreement.

NumberOfUnitsMoMo

number

The total number of units leasable month to month.

NumberOfUnitsTotal

number

Total number of units included in the income property, occupied or unoccupied.

NumberOfUnitsVacant

number

The number of units currently vacant.

OccupantName

string

Name of the current occupant, if any, of the property being sold.

OccupantPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

OccupantType

string

A field that describes the type of occupant, i.e. Owner, Tenant, Vacant.

OffMarketDate

string

The date the listing was taken off market. Where possible, this date is reflective of the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

OffMarketTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the most recent date/time the listing's status was set to and off market status (not Active or Backup)

OnMarketDate

string

The date the listing was placed on market. Where possible, this date is reflective of the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

OnMarketTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the most recent date/time the listing's status was set to Active or Backup. This also includes initial input of the listing to Active/Backup or from a draft or approval status to Active/Backup.

OpenParkingSpaces

number

The number of open or uncovered parking spaces included in the sale.

OpenParkingYN

boolean

A flag indicating that any parking spaces associated with the property are not covered by a roof.

OperatingExpense

number

The costs associated with the operation and maintenance of an income-producing property.

OperatingExpenseIncludes

[string]

When individual expense fields are not used and only a total is entered, this lists the expenses that are included in the OperatingExpense field.

OriginalEntryTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the date/time the listing was entered and made visible to members of the MLS.

OriginalListPrice

number

The original price of the property on the initial agreement between the seller and the seller's broker.

OriginatingSystemID

string

The RESO OUID's OrganizationUniqueId of the Originating record provider. The Originating system is the system with authoritative control over the record. For example; the name of the MLS where the listing was input. In cases where the Originating system was not where the record originated (the authoritative system), see the Originating System fields.

OriginatingSystemKey

string

The system key, a unique record identifier, from the Originating system. The Originating system is the system with authoritative control over the record. For example, the Multiple Listing Service where the listing was input. There may be cases where the Source System (how you received the record) is not the Originating System. See Source System Key for more information.

OriginatingSystemName

string

The name of the Originating record provider. Most commonly the name of the MLS. The place where the listing is originally input by the member. The legal name of the company.

OtherEquipment

[string]

A list of other equipment that will be included in the sale of the property.

OtherExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

OtherParking

string

Other types of parking available to, or part of, the property.

OtherStructures

[string]

A list of structures other than the main dwelling. For example, Guest House, Barn, Shed, etc.

OwnerName

string

Name of the owner of the property being sold.

OwnerPays

[string]

A list of expenses for the property paid for by the owner as opposed to the tenant (e.g. Water, Trash, Electric).

OwnerPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

Ownership

string

A text description of the manner in which title to a property is held. Trust, Corporation, Joint Tennant, Individual.

OwnershipType

string

Current type of ownership of the business being sold. i.e. Corporation, LLC, Sole P, Partnership, etc.,

ParcelNumber

string

A number used to uniquely identify a parcel or lot. This number is typically issued by the county or county assessor. The AP number format varies from county to county. It is recommended that all Parcel Numbers be transmitted without dashes or hyphens.

ParkingFeatures

[string]

A list of features or description of the parking included in the sale/lease.

ParkingTotal

number

The total number of parking spaces included in the sale.

ParkManagerName

string

Name of the manager of the mobile home park.

ParkManagerPhone

string

North American 10 digit phone numbers should be in the format of ###-###-#### (separated by hyphens). Other conventions should use the common local standard. International numbers should be preceded by a plus symbol.

ParkName

string

Name of the mobile home park or corporate/commercial park.

PastureArea

number

Measurement or percentage of the property that has been allocated as pasture or grazing area.

PatioAndPorchFeatures

[string]

A list of features or description of the patio or porch included in the sale/lease.

PendingTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the most recent date/time the listing's status was set to Pending.

PestControlExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

PetsAllowed

[string]

Are pets allowed at the property being leased? A list of yes, no and more detailed restrictions/allowances.

PhotosChangeTimestamp

string

System generated timestamp of when the last update or change to the photos for this listing was made.

PhotosCount

number

The total number of pictures or photos included with the listing.

PoolExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

PoolFeatures

[string]

A list of features or description of the pool included in the sale/lease.

PoolPrivateYN

boolean

The property has a privately owned pool that is included in the sale/lease.

Possession

[string]

A list defining when possession will occur. i.e. COE, COE+1, etc.

PossibleUse

[string]

A list of the type(s) of possible or best uses of the property. Probable use gives a good indication of what the best use or potential use of the property could be.i.e. Primary, Vacation, Investment, Rental, Retirement

PostalCity

string

The official city per the USPS. May be different from the "City".

PostalCode

string

The postal code portion of a street or mailing address.

PostalCodePlus4

string

The postal code +4 portion of a street or mailing address.

PowerProductionType

[string]

This field is a list of the types of power production system(s) available on the property. The key characteristics of the system are expected to appear as the "[type]" in the related power production fields in a flattened implementation (RETS 1.x only) of the power production fields. A relational implementation of power production must omit the type from the field name and use PowerProductionType to create a vertical representation of the various types of power production available. \*\*Note that PV Solar is the only type of power production currently justified in multiple markets and thus shown. Up and coming renewables that could be added in the future depending on uptake: Wind, Geothermal, Thin Film Solar.

PreviousListPrice

number

The most recent previous ListPrice of the listing.

PriceChangeTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the date/time the listing's price was last changed.

PrivateOfficeRemarks

string

A remarks field that is only visible to members of the same offices as the listing agent.

PrivateRemarks

string

Remarks that may contain security or proprietary information and should be restricted from public view.

ProfessionalManagementExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations. This is for a management company. Use ManagerExpense for a individual manager.

PropertyAttachedYN

boolean

A flag indicating that the primary structure is attached to another structure that is not included in the sale. i.e. one unit of a duplex. This flag may be T/F, Y/N or a list of attached or detached. As with all flags, the field may be null. In some systems this information may be part of the Property Sub Type.

PropertyCondition

[string]

A list describing the condition of the property and any structures included in the sale.

PropertySubType

string

A list of types of residential and residential lease properties, i.e. SFR, Condo, etc. Or a list of Sub Types for Mobile, such as Expando, Manufactured, Modular, etc.

PropertyType

string

A list of types of properties such as Residential, Lease, Income, Land, Mobile, Commercial Sale, etc...

PropertyUniversalID

string

The PropertyUniversalID is a unique identifier for all properties in the US and Canada based on country and local identification methods with additional handling for special or more granular cases.

PublicRemarks

string

Text remarks that may be displayed to the public. In an MLS, it is the field where information is entered for the public. This information is intended to be visible on-line. This is typically information that describes the selling points of the building and/or land for sale. Local conditions and rules will determine what such content can contain. Generally, the following information is excluded: any information pertaining to entry to the property, the seller and/or tenant, listing member contact information. In other systems, these remarks will be determined by local business rules.

PublicSurveyRange

string

This field specifically identifies the Range identified by the Public Land Survey System (PLSS).

PublicSurveySection

string

This field specifically identifies the Section identified by the Public Land Survey System (PLSS).

PublicSurveyTownship

string

This field specifically identifies the Township identified by the Public Land Survey System (PLSS).

PurchaseContractDate

string

With for-sale listings, the date an offer was accepted and the listing was no longer on market. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS. With lease listings this may represent a meeting of the minds to lease, but some contractual requirements are yet to be fulfilled, such as contract signing or receipt of the deposit.

RangeArea

number

Measurement or percentage of the property that has been allocated as range.

RentControlYN

boolean

Is the property in a rent control area?

RentIncludes

[string]

A list of services or items that the tenant is not responsible to pay.

RoadFrontageType

[string]

Pick list of types of Road frontage. i.e. Freeway frontage, No Road Frontage, etc. The road frontage of the property is an important factor in determining value of the property and it’s appropriateness for intended use.

RoadResponsibility

[string]

The person or entity responsible for road maintenance (e.g., City, County, Private).

RoadSurfaceType

[string]

Pick list of types of surface of the Road to access the property. The surface of the road(s) for access to the property is an important factor in determining value of the property and it’s appropriateness for intended use.

Roof

[string]

A list describing the type or style of roof. For example Spanish Tile, Composite, Shake, etc.

RoomBasementArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBasementDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBasementFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBasementLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBasementLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBasementWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroom1Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroom1Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroom1Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroom1Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroom1Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroom1Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroom2Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroom2Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroom2Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroom2Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroom2Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroom2Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroom3Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroom3Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroom3Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroom3Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroom3Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroom3Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroom4Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroom4Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroom4Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroom4Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroom4Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroom4Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroom5Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroom5Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroom5Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroom5Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroom5Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroom5Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBathroomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBathroomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBathroomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBathroomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBathroomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBathroomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroom1Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroom1Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroom1Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroom1Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroom1Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroom1Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroom2Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroom2Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroom2Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroom2Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroom2Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroom2Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroom3Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroom3Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroom3Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroom3Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroom3Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroom3Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroom4Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroom4Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroom4Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroom4Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroom4Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroom4Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroom5Area

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroom5Description

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroom5Features

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroom5Length

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroom5Level

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroom5Width

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomBedroomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomBedroomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomBedroomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomBedroomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomBedroomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomBedroomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomDiningRoomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomDiningRoomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomDiningRoomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomDiningRoomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomDiningRoomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomDiningRoomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomFamilyRoomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomFamilyRoomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomFamilyRoomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomFamilyRoomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomFamilyRoomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomFamilyRoomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomKitchenArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomKitchenDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomKitchenFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomKitchenLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomKitchenLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomKitchenWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomLivingRoomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomLivingRoomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomLivingRoomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomLivingRoomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomLivingRoomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomLivingRoomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomMasterBathroomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomMasterBathroomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomMasterBathroomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomMasterBathroomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomMasterBathroomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomMasterBathroomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomMasterBedroomArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomMasterBedroomDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomMasterBedroomFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomMasterBedroomLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomMasterBedroomLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomMasterBedroomWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomOfficeArea

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenArea. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomArea with Kitchen in the relational table's RoomType field.

RoomOfficeDescription

string

Provides a link to the specific property’s high-performance rating or scoring details directly from and hosted by the sponsoring body of the program. Typically provides thorough details, for example, which points where achieved and how, or in the case of a score what specifically was tested and the results. This is a repeating element. If desired replace [Type] with the name of the certification from the GreenBuildingVerificationType list.

RoomOfficeFeatures

[string]

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenFeatures. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomFeatures with Kitchen in the relational table's RoomType field.

RoomOfficeLength

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLength. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLength with Kitchen in the relational table's RoomType field.

RoomOfficeLevel

string

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenLevel. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomLevel with Kitchen in the relational table's RoomType field.

RoomOfficeWidth

number

[type] This field is a repeating element for each type of room selected in the RoomType field. For every RoomType there are two possible implementations. For a flat implementation (RETS 1.x only), each RoomType used is expected to appear as the "[type]" in the related rooms field name. i.e. RoomKitchenWidth. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. i.e. RoomWidth with Kitchen in the relational table's RoomType field.

RoomsTotal

number

The number of rooms in the dwelling.

RoomType

[string]

This field is a list of the types used in the rooms repeating elements. The Type is a list of possible room types. i.e. Bedroom, Bathroom, Living Room, Workshop, etc. Each selected are expected to appear as the "[type]" in the related rooms fields in a flattened implementation (RETS 1.x only) of the room fields. A relational implementation of rooms must omit the type from the field name and use RoomType to create a vertical representation of the various rooms. \*\*Note that Garage or Basement should not be added as a room type and are represented by the ParkingFeatures and Basement fields respectively.

RVParkingDimensions

string

The dimensions of the RV parking area minimally represented as length and width (i.e. 25 x 18) or a measurement of all sides of the polygon representing the usable RV parking space. i.e. 33 x 15 x 12 x 60.

SeatingCapacity

number

The seating capacity of the business being sold.

SecurityFeatures

[string]

A list describing the security features included in the sale/lease.

SeniorCommunityYN

boolean

The community is a senior community.

SerialU

string

Serial number of the mobile or manufactured home. For the first or only unit/section use Serial U over Serial X or Serial XX.

SerialX

string

Serial number of the mobile or manufactured home. For two units/sections, Serial U should be used first, Serial X second over or Serial XX.

SerialXX

string

Serial number of the mobile or manufactured home. For two units/sections, Serial U should be used first, Serial X second over or Serial XX.

Sewer

[string]

A list describing the sewer or septic features of the property.

ShowingAdvanceNotice

number

The hours of advance notice required to schedule a showing.

ShowingAttendedYN

boolean

Does this home require an attended showing? i.e. Yes = licensed agent representing the seller must be present during showing.

ShowingContactName

string

The name of the contact for the showing of the listed property.

ShowingContactPhone

string

A telephone number that should be called to arrange showing the property.

ShowingContactPhoneExt

string

The extension of the given phone number (if applicable).

ShowingContactType

[string]

The type of contact for the showing. i.e. Agent, Broker, Seller.

ShowingDays

[string]

The days of the week that the property is available for showing. i.e. Sundays, Mondays, Tuesdays, Wednesdays, Thursdays, Fridays, Saturdays

ShowingEndTime

string

From the days selected in the ShowingDays field, the end time that the property is available for showing.

ShowingInstructions

string

Remarks that detail the seller's instructions for showing the subject property. Showing instructions may include: contact information, showing times, notice required or other information. These remarks are privileged and are not for public viewing.

ShowingRequirements

[string]

A pick list of types of notice required to see the home. i.e. Appointment Required, Courtesy Call Only, Go Direct, etc.

ShowingStartTime

string

From the days selected in the ShowingDays field, the start time that the property is available for showing.

SignOnPropertyYN

boolean

Is there a sign on the property.

Skirt

[string]

A list of types of mobile home skirting.

SourceSystemID

string

The RESO OUID's OrganizationUniqueId of the Source record provider. The source system is the system from which the record was directly received. In cases where the source system was not where the record originated (the authoritative system), see the Originating System fields.

SourceSystemKey

string

The system key, a unique record identifier, from the Source System. The Source System is the system from which the record was directly received. In cases where the Source System was not where the record originated (the authoritative system), see the Originating System fields.

SourceSystemName

string

The name of the immediate record provider. The system from which the record was directly received. The legal name of the company.

SpaFeatures

[string]

A list of features or description of the spa included in the sale/lease.

SpaYN

boolean

The property has a spa.

SpecialLicenses

[string]

Special licenses required/used by the business being sold. i.e. Beer/Wine, Class H, Professional, Gambling, None.

SpecialListingConditions

[string]

A list of options that describe the type of sale. i.e. Standard, REO, Short Sale, Probate, Auction, NOD, etc., at the time of listing.

StandardStatus

string

The status of the listing as it reflects the state of the contract between the listing agent and seller or an agreement with a buyer (Active, Active Under Contract, Canceled, Closed, Expired, Pending, Withdrawn). This is a Single Select field.

StateOrProvince

string

Text field containing the accepted postal abbreviation for the state or province.

StateRegion

string

A sub-section or area of a defined state or province. Examples would be the Keys in FL or Hudson Valley in NY.

StatusChangeTimestamp

string

The transactional timestamp automatically recorded by the MLS system representing the date/time the listing's status was last changed.

Stories

number

The number of floors in the property being sold.

StoriesTotal

number

The total number of floors in the building. In the case of multi-dwelling structures, this is the entire structure and not the individual dwelling being sold.

StreetAdditionalInfo

string

Information other than a prefix or suffix for the street portion of a postal address.

StreetDirPrefix

string

The direction indicator that precedes the listed property's street name.

StreetDirSuffix

string

The direction indicator that follows a listed property's street address.

StreetName

string

The street name portion of a listed property's street address.

StreetNumber

string

The street number portion of a listed property's street address. In some areas the street number may contain non-numeric characters. This field can also contain extensions and modifiers to the street number, such as "1/2" or "-B". This street number field should not include Prefixes, Direction or Suffixes.

StreetNumberNumeric

number

The integer portion of the street number.

StreetSuffix

string

The suffix portion of a listed property's street address.

StreetSuffixModifier

string

The Street Suffix Modifier allows the member to enter a unique Street Suffix that was not found in the Street Suffix pick list or to extend or prefix the suffix.

StructureType

[string]

The type of structure.

SubAgencyCompensation

string

The total commission to be paid to the Sub Agency, expressed as either a percentage or a constant currency amount.

SubAgencyCompensationType

string

A list of types to clarify the value entered in the SubAgencyCompensation field. For example $, % or some other clarification of the SubAgencyCompensation.

SubdivisionName

string

A neighborhood, community, complex or builder tract.

SuppliesExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

SyndicateTo

[string]

When permitted by the broker, the options made by the agent on behalf of the seller, where they would like their listings syndicated. i.e. Zillow, Trulia, Homes.com, etc.

SyndicationRemarks

string

Becoming more common in the industry, MLS's are hosting a separate "Public Remarks" for syndication purposes. This field should be defaulted to containing the Public Remarks, but upon broker decision, modified to include contact and other information denied by IDX rules, but allowed under local and national regulations.

TaxAnnualAmount

number

The annual property tax amount as of the last assessment made by the taxing authority.

TaxAssessedValue

number

The property value as of the last assessment made by the taxing authority.

TaxBlock

string

A type of legal description for land in developed areas where streets or other rights-of-ways delineate large parcels of land referred to as divided into lots on which homes or other types of developments are built. An example would read "Lot 12 of Block 45 of Tract 3002 of the City of San Dunes, Desert County." Such a description would also reference an official plat filed with the clerk or recorder for that area which shows the location of the block and often the dimensions of the lots therein.

TaxBookNumber

string

Some systems of parcel identification incorporate a method which utilizes a county identifier, a tax book number, a tax map number and a parcel identification number.

TaxExemptions

[string]

A list of tax exemptions as they relate to the property.

TaxLegalDescription

string

A type of legal description for land in developed areas where streets or other rights-of-ways delineate large parcels of land referred to as divided into lots on which homes or other types of developments are built. An example would read "Lot 12 of Block 45 of Tract 3002 of the City of San Dunes, Desert County." Such a description would also reference an official plat filed with the clerk or recorder for that area which shows the location of the block and often the dimensions of the lots therein. The text here is also an index into the property as described by the County Recorder.

TaxLot

string

A type of legal description for land in developed areas where streets or other rights-of-ways delineate large parcels of land referred to as divided into lots on which homes or other types of developments are built. An example would read "Lot 12 of Block 45 of Tract 3002 of the City of San Dunes, Desert County." Such a description would also reference an official plat filed with the clerk or recorder for that area which shows the location of the block and often the dimensions of the lots therein.

TaxMapNumber

string

Some systems of parcel identification incorporate a method which utilizes a county identifier, a tax book number, a tax map number and a parcel identification number.

TaxOtherAnnualAssessmentAmount

number

Any other annual taxes, not including the tax reported in the TaxAmount field, as of the last assessment made by the taxing authority.

TaxParcelLetter

string

Some systems of parcel identification incorporate a method which utilizes a county identifier, a tax book number, a tax map number and a parcel identification number.

TaxStatusCurrent

[string]

The current tax status of the mobile home in cases where the land or space is included in the sale.

TaxTract

string

A type of legal description for land in developed areas where streets or other rights-of-ways delineate large parcels of land referred to as divided into lots on which homes or other types of developments are built. An example would read "Lot 12 of Block 45 of Tract 3002 of the City of San Dunes, Desert County." Such a description would also reference an official plat filed with the clerk or recorder for that area which shows the location of the block and often the dimensions of the lots therein.

TaxYear

number

The year in with the last assessment of the property value/tax was made.

Telephone

[string]

A list of telephone-service related features of the property (e.g. Installed, Public, Available). Note: the previous "Telephone" field was renamed to DistanceToPhoneServiceComments

TenantPays

[string]

A list of services or items that the tenant is responsible to pay.

Topography

string

The state of the surface of the land included with the property. i.e. flat, rolling, etc.

TotalActualRent

number

Total actual rent currently being collected from tenants of the income property.

Township

string

A subdivision of the county.

TransactionBrokerCompensation

string

The total commission to be paid to the transaction facilitator, expressed as either a percentage or a constant currency amount.

TransactionBrokerCompensationType

string

A list of types to clarify the value entered in the TransactionBrokerCompensation field. For example $, % or some other clarification of the TransactionBrokerCompensation.

TrashExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

UnitNumber

string

Text field containing the number or portion of a larger building or complex. Unit Number should appear following the street suffix or, if it exists, the street suffix direction, in the street address. Examples are: "APT G", "55", etc.

UnitsFurnished

string

Are the units furnished? i.e. All Units, Varies By Unit, None.

UnitType1BedroomBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitType1BedroomBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitType1BedroomDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitType1BedroomFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitType1BedroomGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitType1BedroomGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitType1BedroomUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitType2BedroomBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitType2BedroomBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitType2BedroomDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitType2BedroomFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitType2BedroomGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitType2BedroomGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitType2BedroomUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitType3BedroomBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitType3BedroomBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitType3BedroomDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitType3BedroomFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitType3BedroomGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitType3BedroomGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitType3BedroomUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitType4BedroomOrMoreUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitTypeLoftBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitTypeLoftBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitTypeLoftDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitTypeLoftFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitTypeLoftGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitTypeLoftGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitTypeLoftUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitTypeStudioBathsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBathsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBathsTotal with Studio in the relational table's UnitType field.

UnitTypeStudioBedsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioBedsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeBedsTotal with Studio in the relational table's UnitType field.

UnitTypeStudioDescription

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioDescription. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeDescription with Studio in the relational table's UnitType field.

UnitTypeStudioFurnished

string

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioFurnished. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeFurnished with Studio in the relational table's UnitType field.

UnitTypeStudioGarageAttachedYN

boolean

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageAttachedYN. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageAttachedYN with Studio in the relational table's UnitType field.

UnitTypeStudioGarageSpaces

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioGarageSpaces. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeGarageSpaces with Studio in the relational table's UnitType field.

UnitTypeStudioUnitsTotal

number

[type] This field is a repeating element for each type of unit selected in the UnitType field. For every UnitType there are two possible implementations. For a flat implementation (RETS 1.x only), each UnitTypeType used is expected to appear as the "[type]" in the related rooms field name. i.e. UnitTypeStudioUnitsTotal. A relational implementation of UnitType must omit the type from the field name and use UnitTypeType to create a vertical representation of the various rooms. i.e. UnitTypeUnitsTotal with Studio in the relational table's UnitType field.

UnitTypeType

[string]

This field is a list of the types used in the Unit Type repeating elements. The Type is a list of possible Unit Types. i.e. 1, 2, 3 or 2 Bed, Studio, Special Loft, etc. Each selected are expected to appear as the "[type]" in the related UnitType fields in a flattened implementation (RETS 1.x only) of the room fields. A relational implementation of UnitTypes must omit the type from the field name and use UnitTypeType to create a vertical representation of the various unit types. The fact that the field repeats the word "type" is intentional.

UnparsedAddress

string

The UnparsedAddress is a text representation of the address with the full civic location as a single entity. It may optionally include any of City, StateOrProvince, PostalCode and Country.

Utilities

[string]

A list of the utilities for the property being sold/leased.

VacancyAllowance

number

An estimate of the amount of rent that may be foregone because of unoccupied units.

VacancyAllowanceRate

number

An estimate of the percent of rent that may be foregone because of unoccupied units.

Vegetation

[string]

A list of the type(s) of vegetation on the property. Note that this is not for farm crops, but more residential type vegetation.

VideosChangeTimestamp

string

System generated timestamp of when the last update or change to the videos for this listing was made.

VideosCount

number

The total number of videos or virtual tours included with the listing.

View

[string]

A view as seen from the listed property.

ViewYN

boolean

The property has a view.

VirtualTourURLBranded

string

A text field that holds the URL for a branded virtual tour of the property.

VirtualTourURLUnbranded

string

A text field that holds the URL for an unbranded virtual tour of the property.

VirtualTourURLZillow

string

Zillow 3D virtual tour URL

WalkScore

number

A walkability index based on the time to walk from a property to near by essentials such as grocery stores, schools, churches, etc. See www.walkscore.com for more information and requirements for using WalkScore.

WaterBodyName

string

The name, if known, of the body of water on which the property is located. (E.g., lake name, river name, ocean name, sea name, canal name).

WaterfrontFeatures

[string]

Features of the waterfront on which the property is located.

WaterfrontYN

boolean

The property is on the waterfront.

WaterSewerExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

WaterSource

[string]

A list of the source(s) of water for the property

WindowFeatures

[string]

A list of features or description of the windows included in the sale/lease.

WithdrawnDate

string

Date the listing was withdrawn from the market. This is not when a listing contact was cancelled or closed, but a withdrawal from the market while the contract between the seller and listing agent is still in effect and an offer has not been accepted. This is the date entered by the agent reflecting when the change occurred contractually, not a timestamp of when the change was made in the MLS.

WoodedArea

number

Measurement or percentage of the property that is wooded or forest.

WorkmansCompensationExpense

number

The annual expense that is not paid directly by the tenant and is included in the Operating Expense calculations.

YearBuilt

number

The year that an occupancy permit is first granted for the house or other local measure of initial habitability of the build. The type definition permits an empty value with an attribute noting that it is an unknown date or that the building is new construction. While constraints have not been applied, convention at the time of adoption has this as a four (4) digit year value.

YearBuiltDetails

string

A description of the details behind the year the structure was built.

YearBuiltEffective

number

The year a major rebuild/renovated of the structure occurred.

YearBuiltSource

string

Add a list of sources of the year built. i.e. Appraiser, Assessor, Builder, Estimated, etc.,

YearEstablished

number

The year the business being sold was established.

YearsCurrentOwner

number

The number of years the current owner has had possession of the business.

Zoning

string

A division of the city or county into areas of different permissible land uses. This Zone field should be used for the short code that is commonly used. For full textual descriptions please use the ZoningDescription field.

ZoningDescription

string

A list of descriptions of the zoning of the property. The zoning codes are often non-descriptive and variant. Zoning Description is a more descriptive form of the zoning for the property, i.e. Agricultural, Residential, Rezone Possible, etc. Specific zone codes must be added to the Zoning field.

Example Code

EXAMPLE URL

https://api.bridgedataoutput.com/api/v2/OData/test/Property?access\_token=6baca547742c6f96a6ff71b138424f21

EXAMPLE RESPONSE

{

@odata.context: "https://api.bridgedataoutput.com/api/v2/OData/test/$metadata#Property",

@odata.nextLink: "https://api.bridgedataoutput.com/api/v2/OData/test/",

value: [

{

@odata.context: "https://api.bridgedataoutput.com/api/v2/OData/test/$metadata#Property",

@odata.id: "https://api.bridgedataoutput.com/api/v2/OData/test/Property('P\_5af601c3fc76173b348291e9')",

ListingKey: "P\_5af601c3fc76173b348291e9",

ListingId: "5af601c3fc76173b348291ea",

OriginatingSystemKey: "test",

LotSizeAcres: 6,

LotSizeSquareFeet: 1746,

LotSizeSource: "Agent",

LotSizeDimensions: "75 x 673 x 114 x 284",

LotSizeDimensionsSource: "Assessor",

AssociationFee: 185,

ListPrice: 518538,

BedroomsPossible: 8,

BathroomsTotalInteger: 13,

BedroomsTotal: 3,

MainLevelBathrooms: 2,

FireplacesTotal: 1,

FireplaceFeatures: [

"One",

"Living Room Fireplace"

],

GarageSpaces: 4,

BathroomsHalf: 2,

Stories: 1,

StoriesTotal: 3,

PublicRemarks: "Nam enim nisi fugit ab. Earum doloribus nesciunt cupiditate reiciendis. Sed expedita voluptatem quas sequi. Aut optio velit itaque. Officia fugiat adipisci quia sit dolore velit est libero. Cum debitis tempore sint expedita quidem est doloremque sit. Esse sequi repudiandae rerum soluta vitae inventore et quia. Voluptatem rerum itaque dolores libero placeat. Fuga quasi porro libero voluptas iste odio. Soluta et modi et et quisquam accusamus fuga. Totam architecto incidunt ea incidunt eligendi modi. Magni cupiditate unde consequatur natus perspiciatis nihil. Et rerum odit quis recusandae sequi iste cum nam.",

LivingArea: 5211,

LivingAreaUnits: "SquareFeet",

YearBuilt: 1985,

YearBuiltDetails: "Ea pariatur modi debitis sunt nihil provident ut aut. Dicta aut voluptatem rerum aspernatur rerum et exercitationem. Consequatur sint qui ad rerum. Quaerat officia consequatur culpa atque dolores numquam accusamus libero.",

YearBuiltSource: "Agent",

StandardStatus: "Active",

MlsStatus: "Active",

PropertyType: "Business",

PropertySubType: "Single Family Residence",

ListingContractDate: "2018-05-11T20:49:07.085Z",

PreviousListPrice: 300393,

OriginalListPrice: 650285,

Contingency: "Quidem mollitia dicta unde perspiciatis laboriosam dolore dolore. Odit tenetur dolorem reiciendis culpa qui. Aut modi qui ipsam cupiditate fuga et vero provident.",

Disclosures: [

"Bylaw Infractions"

],

NumberOfUnitsTotal: 176,

ListingTerms: [

"Cash"

],

OnMarketDate: "2017-12-04T20:49:07.084Z",

DaysOnMarket: 158,

CloseDate: "2017-08-01T07:24:19.329Z",

StatusChangeTimestamp: "2018-05-11T20:49:07.085Z",

Ownership: "Partnership",

FinancialDataSource: [

"Listing Broker",

"Not Available"

],

Appliances: [

"Dishwasher",

"Disposal"

],

Telephone: [

"Installed"

],

View: [

"View"

],

ViewYN: false,

WaterfrontYN: true,

WaterSource: [

"Municipal"

],

Zoning: "AE",

ZoningDescription: "Agricultural",

AboveGradeFinishedArea: 2221,

BelowGradeFinishedArea: 3513,

RoomsTotal: 15,

ExteriorFeatures: [

"Balcony",

"Fenced Yard",

"Sprinkler"

],

Flooring: [

"Hardwood"

],

FoundationDetails: "Pile",

FrontageType: [

"Golf Course"

],

FrontageLength: 275,

RoadResponsibility: [

"City"

],

Gas: [

"Available"

],

AccessibilityFeatures: [

"Ramp"

],

Heating: [

"Electric",

"Wood"

],

AdditionalParcelsYN: false,

OccupantName: "Albertha Guillermo Wiza",

OccupantPhone: "224-423-6596 x4231",

OccupantType: "Vacant",

PoolPrivateYN: false,

Possession: [

"Close of Escrow"

],

RoadSurfaceType: "Cement",

Roof: [

"Metal"

],

ConstructionMaterials: [

"Wood"

],

Sewer: [

"Yes Connected"

],

CarportSpaces: 2,

OpenParkingYN: true,

LaundryFeatures: [

"In Kitchen",

"Gas Dryer Hookup",

"Washer"

],

PrivateRemarks: "Vero delectus sapiente neque. Quidem debitis libero nulla placeat fuga fugiat consectetur. Doloribus molestiae harum dolorem. Asperiores enim rerum voluptates vitae eum in. Magni nulla laborum voluptatem. Deleniti quo dolore ut officia nostrum alias. Facilis omnis quia eos voluptas magni est. A fuga impedit molestiae. Expedita quos beatae sunt. Dignissimos rerum atque quae aut quasi voluptates molestias tenetur. Id voluptatem non cupiditate voluptate sit. Quo eveniet doloremque autem atque. Sit ducimus et amet delectus facilis neque. Delectus sed blanditiis nisi quidem iusto non error non. In eum commodi eveniet. Debitis est provident eos iusto. Velit necessitatibus necessitatibus quasi et voluptate illo.",

TaxAssessedValue: 856307,

TaxAnnualAmount: 573226,

TaxYear: 2017,

TaxStatusCurrent: [

"Assessed"

],

SubdivisionName: "est",

ModificationTimestamp: "2018-02-04T21:45:33.357Z",

BuyerFinancing: [

"Assumed",

"Cash",

"Contract",

"Conventional",

"FHA",

"FHA 203(b)",

"FHA 203(k)"

],

NewConstructionYN: false,

HomeWarrantyYN: true,

ArchitecturalStyle: [

"distinctio",

"impedit"

],

CommunityFeatures: [

"doloremque"

],

PatioAndPorchFeatures: [

"veritatis",

"pariatur",

"molestias",

"quos",

"provident",

"veniam",

"temporibus",

"totam",

"ut"

],

OtherStructures: [

"laboriosam",

"qui",

"ut",

"illo",

"porro",

"nemo",

"et"

],

ParcelNumber: 1578,

AttachedGarageYN: true,

CarportYN: false,

ContractStatusChangeDate: "2017-10-24T16:44:40.821Z",

ContingentDate: "2017-09-17T09:17:06.633Z",

OriginalEntryTimestamp: "2017-06-02T00:38:48.682Z",

AssociationYN: false,

AssociationName: "voluptatem",

AssociationFeeFrequency: "Monthly",

PhotosCount: 2,

ListingService: "Limited Service",

ListingAggreement: "Exclusive Agency",

LeaseConsideredYN: true,

CopyrightNotice: "Nemo accusantium aliquid eveniet impedit ipsa ut soluta id. Ab impedit soluta vel sunt minus accusamus vitae nesciunt. Eum harum est omnis. Nihil recusandae sit vitae. Id dolores iste animi et odio esse libero. Maxime facilis nobis omnis quis maxime rerum distinctio qui. Velit et vel impedit in et optio laborum. Velit blanditiis sequi dolor laboriosam amet est perferendis. Et laudantium nemo quaerat odio fugiat eos consequuntur. Enim qui laudantium distinctio sunt.",

Disclaimer: "Expedita aperiam placeat eligendi aspernatur sit. Eius sit tenetur sequi sit et. Ullam ullam at fugit. Repudiandae explicabo iusto eaque dolor suscipit reprehenderit ut. Sequi est perspiciatis labore cumque tempore quod id eos. Ratione quae rerum vitae ut debitis. Et eligendi et ullam quo odit corporis omnis reprehenderit. Doloribus ea sint est aliquid eum et.",

PendingTimestamp: "2017-11-08T23:17:50.104Z",

PriceChangeTimestamp: "2017-08-15T03:05:25.153Z",

MajorChangeType: "Price reduction",

MajoreChangeTimestamp: "2017-08-25T19:45:11.311Z",

DualVariableCompensationYN: false,

SignOnPropertyYN: false,

InternetEntireListingDisplayYN: false,

InternetAddressDisplayYN: false,

InternetConsumerCommentYN: true,

InternetAutomatedValuationDisplayYN: true,

SyndicateTo: [

"Zillow",

"Trulia",

"hotspads.com"

],

PrivateOfficeRemarks: "Cupiditate odit beatae sit quis placeat doloremque aut vero. Nihil facilis aut possimus sapiente. Qui deserunt quis qui quis autem. A veritatis asperiores cupiditate recusandae quisquam vel. Sed et veniam enim fugit doloremque inventore. Et repudiandae temporibus saepe totam ex quam eaque placeat. Incidunt delectus perferendis corrupti adipisci enim impedit laudantium aspernatur. Et voluptas quia rerum tenetur natus ab. Aut voluptate sunt excepturi nisi optio excepturi ullam. Aut molestiae omnis corrupti vitae facilis expedita et aut. Corrupti qui non unde veritatis quo aut dolore. Voluptas perspiciatis in saepe nobis tempore nam excepturi provident. Et nesciunt voluptas dolorem suscipit minus ducimus aut eaque.",

ShowingInstructions: "Esse laborum non consectetur et voluptas quis at. Aut voluptate qui eaque est dolorem. Iste eveniet vel voluptates repellat. Qui neque ratione tempora neque corrupti voluptatibus neque molestiae. Harum et sint labore delectus atque ut id. Dolor temporibus officia voluptate aperiam explicabo esse inventore. Est qui maxime similique. Vel est officiis minima. Doloremque beatae corporis itaque facilis. Aspernatur fugiat ad recusandae soluta. Tempore placeat consequatur cupiditate quibusdam. Non neque ea et magni qui. Impedit modi itaque ut voluptas excepturi quisquam. Officia ducimus qui dolor autem dolor mollitia. Aut mollitia libero consectetur est totam assumenda molestias veniam. Nam eligendi ducimus dolores nihil qui. Quia et est sit quam ex. Enim nam aut nobis qui voluptatibus vel. Blanditiis vero libero quibusdam voluptatibus reprehenderit et libero.",

ShowingContactPhone: "612-812-6467",

ShowingContactName: "Dameon Howard Schulist",

ShowingContactType: "Agent",

LockBoxLocation: "Recusandae non expedita corrupti officia molestiae fugit. Voluptate voluptatem officiis deleniti laborum omnis fuga. Aperiam ad voluptatem molestiae sunt reiciendis qui atque.",

LockBoxType: "Rerum sint accusamus aliquid voluptas. Voluptatem sunt et consequatur. Fugit quaerat id temporibus optio ut fugit.",

LockBoxSerialNumber: "uhn4dkn1sj1tx78bkmy81e717",

AccessCode: "9y4x",

Exclusions: "Ut recusandae sunt qui odio quia sit nostrum. Odit maiores voluptates rem modi. Eveniet voluptatem magnam fugiat. Sed maxime et in. Commodi pariatur molestiae facilis architecto.",

Inclusions: "Pariatur vel nisi quo similique dolorem. Porro velit suscipit modi deleniti dolor rem voluptatem. Accusantium voluptatem eos eos quidem ut non. Et voluptas enim id molestiae consectetur labore et facilis. Praesentium voluptatem ipsum consequatur et sed molestias.",

SpecialListingConditions: [

"Standard",

"REO",

"Short Sale",

"Probate",

"Auction"

],

ConcessionComments: "Illum enim ipsa ut possimus sit mollitia. Quo autem quo iste quo sit. Nulla voluptatem autem voluptatem laborum ratione. Placeat corrupti reiciendis natus magnam velit.",

Directions: "Officiis ut alias ex libero. Ea est aperiam qui soluta animi eum ipsam vel. Nulla quis impedit velit amet autem voluptatem. Ut exercitationem dolorem qui ipsa necessitatibus. Nisi repellat aut perspiciatis maiores iusto. Tempora quisquam perspiciatis autem est dignissimos. Neque placeat autem voluptates quo.",

ElementarySchool: "Watsica Group",

ElementarySchoolDistrict: "New Hampshire",

MiddleOrJuniorSchool: "Schultz-Rempel",

MiddleOrJuniorSchoolDistrict: "Texas",

HighSchool: "Graham and Sons",

HighSchoolDistrict: "South Carolina",

LandLeaseYN: true,

LandLeaseAmount: 47535197463,

LandLeaseAmountFrequency: "Bi-Weekly",

LandLeaseExpireactionDate: "2018-06-26T11:40:32.694Z",

HorseYN: true,

SeniorCommunityYN: false,

PropertyAttachedYN: true,

GarageYN: true,

Levels: [

"One Level",

"Two Level",

"Split Level"

],

LivingAreaSource: "Estimate",

AboveGradeFinishedAreaSource: "Estimate",

AboveGradeFinishedAreaUnits: "Square Feet",

FoundationArea: 505038,

CoveredSpaces: 4,

ParkingTotal: 1,

EntryLocation: "Top Level",

BuilderName: "Pacocha, Welch and Ziemann",

BuilderModel: "Rustic Steel Computer",

InteriorFeatures: "Atque doloribus qui non mollitia voluptatibus qui nostrum et. Nemo quaerat qui vitae quidem rerum quos vel repellat. Ullam amet sint et totam accusantium temporibus.",

DirectionFaces: "North-East",

GreenEnergyEfficient: [

"Energy efficient doors",

"Energy efficient windows",

"Energy efficient stove"

],

Walkscore: 78,

TaxLot: "Lot 19 of Block 23 of Tract 2544 of East Jena",

TaxBlock: "Lot 1 of Block 0 of Tract 2270 of Yasmineton",

TaxTract: "Lot 13 of Block 50 of Tract 878 of Schuliststad",

TaxLegalDescription: "Ab eum eos et eligendi et in aut. Enim saepe temporibus sapiente perspiciatis qui est. Voluptatem in doloribus saepe id. Delectus repellat doloremque aut impedit velit soluta consequuntur et. Enim officiis repellat qui laudantium modi modi reiciendis similique. Sapiente aut aut placeat eveniet. Inventore deserunt nesciunt repudiandae qui rerum qui. Est nihil provident est accusantium maxime minus qui.",

Media: [

{

MediaURL: "https://dvvjkgh94f2v6.cloudfront.net/8v8b238/24i9v02ojs.jpg",

MediaObjectID: "ME\_5af601c3fc76173b348291e7",

Order: 1,

MimeType: "image/jpeg",

ShortDescription: "Ventito catena tempus acsi ceno."

},

{

MediaURL: "https://dvvjkgh94f2v6.cloudfront.net/8v8b238/24i9v02oj1.jpg",

MediaObjectID: "ME\_5af601c3fc76173b348291e8",

Order: 2,

MimeType: "image/jpeg",

ShortDescription: "Denuo tui clementia quis unde."

}

],

Coordinates: [

-122.4248504,

37.76335862

],

City: "Mission Dolores",

CountyOrParish: "San Francisco Co.",

Country: "US",

PostalCode: "94110",

StateOrProvince: "CA",

Longitude: -122.4248504,

Latitude: 37.76335862,

Elevation: 75573,

ElevationUnits: "Meters",

AnchorsCoTenants: [

"Keebler Group"

],

AssociationFee2: 122,

AssociationFee2Frequency: "One Time",

AssociationFeeIncludes: [

"Water",

"Sewer",

"Security",

"Snow Removal",

"Cable TV",

"Earthquake Insurance",

"Insurance",

"Trash",

"Maintenance Grounds",

"Pest Control",

"Utilities",

"Gas"

],

AssociationName2: "Mosciski-Blick",

AssociationPhone: "1-172-732-9966",

AvailabilityDate: "2018-06-12T23:39:16.548Z",

BathroomsFull: 4,

BathroomsOneQuarter: 2,

BathroomsPartial: 5,

BathroomsThreeQuarter: 1,

BathroomsTotalDecimal: 10.25,

BelowGradeFinishedAreaSource: "Assessor",

BelowGradeFinishedAreaUnits: "Square Meteres",

BuildingAreaSource: "Assessor",

BuildingAreaTotal: 9202119,

BuildingAreaUnits: "Square Feet",

BuildingFeatures: [

"Cafeteria"

],

BuildingName: "Mrs. Haylie Bauch Co",

BusinessName: "Kshlerin-Jaskolski",

BusinessType: "Arts and Entertainment",

CityRegion: "Rathstad",

CommonWalls: [

"End Unit",

"No One Above",

"2+ Common Walls",

"No Common Walls",

"No One Below",

"1 Common Wall"

],

CoolingYN: false,

DocumentsCount: 1,

DoorFeatures: [

"Truck Doors",

"French Doors"

],

ElectricOnPropertyYN: true,

Fencing: [

"Chain Link",

"Wood",

"Block",

"Wrought Iron"

],

FireplaceYN: false,

HeatingYN: true,

ListAgentFirstName: "Cyrus",

ListAgentFullName: "Cyrus Jaren Krajcik",

ListAgentKey: "M\_5af601aecd966d3afc88eb0a",

ListAgentLastName: "Krajcik",

ListAgentMiddleName: "Jaren",

ListAgentMlsId: "5af601aecd966d3afc88eb0b",

ListAgentPreferredPhone: "339.831.5242",

ListOfficeKey: "O\_5af6019c75f1043ad481d9d8",

ListOfficeMlsId: "5af6019c75f1043ad481d9d9",

ListOfficeName: "Tremblay, Renner and Ullrich Realty",

LotDimensionsSource: "Assessor",

OriginatingSystemID: "test",

OriginatingSystemName: "Test",

OtherEquipment: [

"",

"Generator"

],

IDXParticipationYN: false

}

],

@odata.count: 1

}

GET

/{dataset}/Property('{ListingKey}')

ReferenceAPI Explorer

/Member

GET

/{dataset}/Member

ReferenceAPI Explorer

GET

/{dataset}/Member('{MemberKey}')

ReferenceAPI Explorer

/Office

GET

/{dataset}/Office

ReferenceAPI Explorer

GET

/{dataset}/Office('{OfficeKey}')

ReferenceAPI Explorer

/OpenHouse

GET

/{dataset}/OpenHouse

ReferenceAPI Explorer

GET

/{dataset}/OpenHouse('{OpenHouseKey}')

ReferenceAPI Explorer

/Lookup

GET

/{dataset}/Lookup

ReferenceAPI Explorer

Bridge Web API

/listings

GET

/{dataset}/listings

ReferenceAPI Explorer

GET

/{dataset}/listings/{listingId}

ReferenceAPI Explorer

/agents

GET

/{dataset}/agents

ReferenceAPI Explorer

GET

/{dataset}/agents/{agentId}

ReferenceAPI Explorer

/offices

GET

/{dataset}/offices

ReferenceAPI Explorer

GET

/{dataset}/offices/{officeId}

ReferenceAPI Explorer

/openhouses

GET

/{dataset}/openhouses

ReferenceAPI Explorer

GET

/{dataset}/openhouses/{openhouseId}

ReferenceAPI Explorer

Public Data

/parcels

GET

/pub/parcels

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/assessments

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/transactions

ReferenceAPI Explorer

/assessments

GET

/pub/assessments

ReferenceAPI Explorer

/transactions

GET

/pub/transactions

ReferenceAPI Explorer

Zestimates

/zestimates

GET

/zestimates\_v2/zestimates

ReferenceAPI Explorer

Zillow Group Econ Data

/marketreport

GET

/zgecon/marketreport

ReferenceAPI Explorer

/marketreport/replication

GET

/zgecon/marketreport/replication

ReferenceAPI Explorer

/region

GET

/zgecon/region

ReferenceAPI Explorer

/cut

GET

/zgecon/cut

ReferenceAPI Explorer

/type

GET

/zgecon/type

ReferenceAPI Explorer

Zillow Agent Reviews

/Review

GET

/reviews/Reviews

ReferenceAPI Explorer

/Reviewee

GET

/reviews/Reviewees

Bridge Web API

/listings

GET

/{dataset}/listings

ReferenceAPI Explorer

GET

/{dataset}/listings/{listingId}

ReferenceAPI Explorer

/agents

GET

/{dataset}/agents

ReferenceAPI Explorer

GET

/{dataset}/agents/{agentId}

ReferenceAPI Explorer

/offices

GET

/{dataset}/offices

ReferenceAPI Explorer

GET

/{dataset}/offices/{officeId}

ReferenceAPI Explorer

/openhouses

GET

/{dataset}/openhouses

ReferenceAPI Explorer

GET

/{dataset}/openhouses/{openhouseId}

ReferenceAPI Explorer

Public Data

/parcels

GET

/pub/parcels

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/assessments

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/transactions

ReferenceAPI Explorer

/assessments

GET

/pub/assessments

ReferenceAPI Explorer

/transactions

GET

/pub/transactions

ReferenceAPI Explorer

Zestimates

/zestimates

GET

/zestimates\_v2/zestimates

ReferenceAPI Explorer

Zillow Group Econ Data

/marketreport

GET

/zgecon/marketreport

ReferenceAPI Explorer

/marketreport/replication

GET

/zgecon/marketreport/replication

ReferenceAPI Explorer

/region

GET

/zgecon/region

ReferenceAPI Explorer

/cut

GET

/zgecon/cut

ReferenceAPI Explorer

/type

GET

/zgecon/type

ReferenceAPI Explorer

Zillow Agent Reviews

/Review

GET

/reviews/Reviews

ReferenceAPI Explorer

/Reviewee

GET

/reviews/Reviewees

Reference

Public Data

/parcels

GET

/pub/parcels

ReferenceAPI Explorer

Description

Retrieves a list of parcels.

Request

Required fields

Name

Type

Description

access\_token

string

Token to identify the user or application.

offset

number

Skips this number of results.

limit

number

Limits the size of the result set.

sortBy

string

Response field to sort query by.

order

string

Order of responses: "asc/desc".

fields

string

Filters Response fields eg. fields=id,price.

near

string

Coord or location eg. near=-73.98,40.73 or near=San Diego.

radius

string

Search Radius in miles, km, or degrees (no units).

box

string

Coordinates representing a box eg. box=-112.5,33.75,-123,39

poly

string

Minimum 3 pairs of coordinates representing a polygon eg. poly=-112.5,33.75,-123,39,-120,38

geohash

string

Alphanumeric geohash eg. geohash=ezs42

Response

Name

Type

Description

address

object

Address attributes of the parcel.

address.city

string

City name.

address.full

string

Full address.

address.house

string

House number.

address.houseExt

string

House number extension.

address.state

string

State name.

address.street

string

Street name.

address.streetPost

string

Additional street information.

address.streetPre

string

Street prefix.

address.streetSuffix

string

Street suffix.

address.unit

string

Unit number.

address.unitType

string

Unit type.

address.zip

string

Postal code.

address.zip4

string

Zip4.

apn

string

Assessor Parcel Number, the unique identifier assigned by the tax authority.

areas

[array]

Area data.

areas.areaSquareFeet

number

Area in square feet.

areas.type

string

Type of areas.

BridgeModificationTimestamp

string

A timestamp representing when last this record was modified in the Bridge system.

building

[array]

Building data.

building.airConditioning

string

Air conditioning feature of the building.

building.architecturalStyle

string

The architectural style of the building.

building.baths

number

The total number of baths in the building.

building.bedrooms

number

The total number of bedrooms in the building.

building.class

string

The class of the building.

building.classDescription

string

Description of the building's class.

building.comments

string

Comments about the building.

building.condition

string

The condition of the building.

building.effectiveYearBuilt

number

The year in which the building was efficively built.

building.fireplace

string

Fireplace feature of the building.

building.foundation

string

Foundation feature of the building.

building.fullBaths

number

The total number of full baths in the building.

building.halfBaths

number

Number of half baths in the building

building.heating

string

Heating feature of the building.

building.kitchens

number

The total number of kitchens in the building.

building.noOfUnits

number

The total number of units in the building if applicable.

building.occupancyStatus

string

The occupancy status of the building.

building.quality

string

The quality of the building.

building.quarterBaths

number

Number of quarter baths in the building

building.sewer

string

Sewer or septic features of the building.

building.threeQuarterBaths

number

Number of three quarter baths in the building

building.totalRooms

number

The total number of rooms in the building.

building.totalStories

number

The total number of stories in the building.

building.water

string

Water source of the building.

building.yearBuilt

number

The year in which the building was built.

coordinates

[number]

Longitude and latitude coordinates, eg. [-122.441, 37.717].

county

string

County name.

fips

string

Five digit county code ID.

garages

[array]

Information about garages

garages.areaSquareFeet

number

The garage area in square feet

garages.carCount

number

The number of cars that can fit in the garage

garages.type

string

The type of garage

id

number

Unique ID to identify the parcel.

landUseCode

string

Five character code describing this parcel's land use type.

landUseDescription

string

Description that matches the land use code.

landUseGeneral

string

Two character land use code (example: "RR" for Residential, "AG" for Agricultural).

legal

object

Legal details of the parcel.

legal.block

string

The block.

legal.city

string

The name of the city.

legal.district

string

The name of the district.

legal.lot

string

The lot.

legal.lotDescription

string

The lot description.

legal.municipality

string

The name of the municipality.

legal.otherLot

string

The other lot.

legal.phase

string

The phase.

legal.section

string

The section.

legal.subdivisionName

string

The name of the subdivision.

legal.township

string

The name of the township.

legal.tract

string

The tract.

legal.unit

string

The unit.

lotSizeAcres

number

Size of the lot in acres.

lotSizeDepth

number

Average depth of lot perpendicular to frontage.

lotSizeFrontage

number

Distance in feet between the front of the building and the property boundary

lotSizeIrregular

string

Perimeter measurements for non-rectangular shaped lots.

lotSizeSquareFeet

number

Size of the lot in square feet.

lotTopography

string

Short description of lot topography.

numberOfBuildings

number

Number of buildings included in the parcel.

ownerName

[string]

Full names of the parcel owners.

parcelID

number

Unique ID to identify the parcel.

pools

[array]

Information about pools

pools.size

string

Size of pool

pools.type

string

Type of pool

state

string

Two character state abbreviation.

zoningCode

string

Municipality-specific zoning code indicates what part of the zoning or land use bylaw applies to this parcel.

zoningDescription

string

Short description of zoning code.

zpid

string

Zillow ID

Example Code

EXAMPLE URL

https://api.bridgedataoutput.com/api/v2/pub/parcels?access\_token=SERVER\_TOKEN

EXAMPLE RESPONSE

{

success: true,

status: 200,

bundle: [

{

id: "1000",

fips: "01001",

apn: "01-04-19-0-000-009.000",

county: "AUTAUGA",

state: "AL",

landUseGeneral: "RR",

landUseCode: "RR103",

landUseDescription: "Mobile Home",

zoningCode: null,

zoningDescription: null,

lotSizeAcres: 8,

lotSizeSquareFeet: 348480,

lotSizeFrontage: null,

lotSizeDepth: null,

lotSizeIrregular: null,

lotTopography: null,

numberOfBuildings: 0,

address: {

full: "100 TEST ST 20 W TESTCITY AL 36051",

house: "100",

houseExt: null,

streetPre: null,

street: "TEST ST 20",

streetSuffix: null,

streetPost: "W ",

unitType: null,

unit: null,

city: "TESTCITY",

state: "AL",

zip: "36051",

zip4: "3009"

},

legal: {

lotDescription: "A TRACT OF LAND IN THE SW 1/4 NW 1/4 OF SEC 19T 20N R 16E. BEG AT NW CR OF SAID 40; S TO NR.O.W. OF CO RD; NE'LY ALONG SAID",

lot: null,

otherLot: null,

block: null,

subdivisionName: null,

unit: null,

section: null,

phase: null,

tract: null,

district: null,

municipality: null,

city: null,

township: null

},

coordinates: [

-122.000001,

47.600001

],

building: [

{

noOfUnits: 0,

occupancyStatus: "Owner Occupied or Primary Residence",

class: null,

classDescription: null,

quality: "C",

condition: "Good",

architecturalStyle: null,

heating: "Forced air",

airConditioning: "Refrigeration",

foundation: "Concrete",

fireplace: null,

water: "Well",

sewer: "Septic",

yearBuilt: 1998,

effectiveYearBuilt: null,

totalStories: 1,

totalRooms: 7,

bedrooms: 0,

kitchens: 1,

fullBaths: 2,

baths: 3,

comments: null

}

],

areas: [

{

type: "Living Building Area",

areaSquareFeet: 2048

},

{

type: "Perimeter Building Area",

areaSquareFeet: 202

}

],

ownerName: [

"JANE DOE",

"JOE SMITH"

],

transactionsUrl: "https://api.bridgedataoutput.com/api/v2/pub/parcels/372176/transactions",

assessmentsUrl: "https://api.bridgedataoutput.com/api/v2/pub/parcels/372176/assessments",

url: "https://api.bridgedataoutput.com/api/v2/pub/parcels/372176"

}

],

total: 1

}

GET

/pub/parcels/{parcelId}

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/assessments

ReferenceAPI Explorer

GET

/pub/parcels/{parcelId}/transactions

ReferenceAPI Explorer

/assessments

GET

/pub/assessments

ReferenceAPI Explorer

/transactions

GET

/pub/transactions

ReferenceAPI Explorer

Zestimates

/zestimates

GET

/zestimates\_v2/zestimates

ReferenceAPI Explorer

Zillow Group Econ Data

/marketreport

GET

/zgecon/marketreport

ReferenceAPI Explorer

/marketreport/replication

GET

/zgecon/marketreport/replication

ReferenceAPI Explorer

/region

GET

/zgecon/region

ReferenceAPI Explorer

/cut

GET

/zgecon/cut

ReferenceAPI Explorer

/type

GET

/zgecon/type

ReferenceAPI Explorer

Zillow Agent Reviews

/Review

GET

/reviews/Reviews

ReferenceAPI Explorer

/Reviewee

GET

/reviews/Reviewees

Zillow Group Econ Data

/marketreport

GET

/zgecon/marketreport

ReferenceAPI Explorer

/marketreport/replication

GET

/zgecon/marketreport/replication

ReferenceAPI Explorer

/region

GET

/zgecon/region

ReferenceAPI Explorer

/cut

GET

/zgecon/cut

ReferenceAPI Explorer

/type

GET

/zgecon/type

ReferenceAPI Explorer

Zillow Agent Reviews

/Review

GET

/reviews/Reviews

ReferenceAPI Explorer

/Reviewee

GET

/reviews/Reviewees

ReferenceAPI Explorer

Frequently asked questions

How do I set up a Bridge account?

Typically an invitation for access to the Bridge platform is initiated by the data providers (MLSs).

How long does it take for an MLS to approve my data access?

Any data access is wholly at the discretion of the MLS, and each will have their own processes and criteria for approval.

What MLSs do you currently have API integrations for?

Currently available datasets can be seen on the dashboard after you have created an account. We are always in the process of adding more datasets. If you are a brokerage and a member of an MLS that you do not see on the dashboard, please reach out to us to discuss further options.

Is the RESO Web API and Data Dictionary RESO certifed?

Yes, our RESO Web API is Platinum compliant and certified. MLS data is normalized to RESO Data Dictionary standards.

How does the MLS data get returned?

MLS data is normalized to the RESO Data Dictionary and returned in JSON format (from the Web API), XML or CSV (from RETS), Any fields that are not able to be normalized (e.g. 'custom' or 'native' fields) are made available as well, at the MLS's discretion.

I am a broker and a member of multiple MLSs - can I get all my data with a single feed?

Yes, you may create a virtual dataset that will allow you to retrieve listing data across all of your datasets.

How often do you refresh the MLS listing data?

Typically we refresh listing data every 10 minutes or less, depending on the resource and the MLS.

What MLS listing data will I have access to after access has been approved?

The API can serve all data from all resources (including Properties, Members, Offices, Open Houses and any others like Rooms and UnitTypes), as well as off-market data. However, the MLS has full control of what data is served through the API on a per application basis, and access is at their discretion.

Can I use the Bridge to replicate an entire MLS dataset?

Yes, we have tools that make replication easy, although this is only available at the discretion of the MLS that has approved your data access. See the documentation for more information.

Can I get access to all MLS listing data nationwide?

No - the Bridge is available on an MLS by MLS basis, and each of them individually control the data they distribute.

How can I access non-MLS datasets?

The process is essentially the same as for MLS data access. After registering an account, apply for access to any available dataset you are interested in. Access will be granted depending on your specific use-case.

How much does using the Bridge API cost?

Bridge Interactive does not charge vendors and brokerages any additional service fees when integrating with our partner MLSs. Note that MLSs may charge their own licensing fees, at their discretion.

There are three data transports available, which should I use?

For MLS data you may choose between the Bridge Web API, the RESO Web API, or RETS. We recommend using the RESO Web API for accessing MLS data as it uses the latest industry standard and has the best interoperability with other real estate API platforms. RETS can be used if you wish to leverage your existing data ingestion pipeline. The Bridge Web API is best for simple API queries and is required for any non-MLS datasets.