# Working with Entity Refs and Lists

Last Modified on 12/20/2022 3:49 pm EST

## Entity Ref

> The end points below use POST and PUT **but they do not change any data** . They act like GET endpoints. The reason we use POST and PUT is to allow for passing an object as a parameter.

Many data objects have properties with a Data Type of Entity Ref. An Entity Ref is the relationship between 2 tables. For example customer has an Entity Ref for Sales Tax Code.

### Json Sample

```
{
  "Id": 1947,
  "Name": "Peter Pan LLC",
  "IsActive": true,
  "CompanyName": "Peter Pan LLC",
  "SalesTaxCodeRef": {
      "Id": 3,
      "Name":"Non";
   }
}
```

In order to populate the customer object we need either the Id (preferable) or the Name of the Entity Ref. The Order Time API exposes the entityRef end point that returns an array of EntityRef objects.

/entityref

POST

## Lists

The Lists in Order Time are used to find records. The Order Time API exposes the lists end point that returns an array of the objects requested.

/list

POST - Return an array of the objects requested.

PUT - Returns the record count.

# List Info

Both the Entity Ref and Lists end points use the ListInfo object as a parameter.  Please consider the following.

- The Type [RecordTypeEnum] parameter is required.   This tells the system which object to get.
- All filters must resolve to True.
- When using the In or Not In operators,  the FilterValueArray property should contain a comma delimited string of values.
- When using the Custom DataMacro,the  FilterValueArray property is required.
  - For from date : "2018-02-01,"
  - For to date:  ",2018-02-01"
  - For between dates:  "2018-01-01,2018-01-31"

### Sample ListInfo

```
{
  "Type": 250,
  "Filters" : [
   {
      "PropertyName": "Name",
      "Operator": 12,
      "FilterValueArray": "UPS"
   },
   {
      "PropertyName": "RecordInfo.ModifiedDate",
      "FieldType": 10,
      "DateMacro" : 9
   }
  ]
  "Sortation" : {
     "PropertyName": "Id",
      "Direction": 2
   },
  "PageNumber": 1,
  "NumberOfRecords": 20
}
```

The example above will get Ship Methods where the Name is Like 'UPS' and the Modified date was this year, sorted by Id descending.

### Sample Entity Ref Response

```
[
   {
      "Id": 15,
      "Name": "UPS Ground"
   },
   {
      "Id": 4,
      "Name": "UPS"
   }
]
```

### Sample List Response

```
[
    {
        "Name": "UPS Ground",
        "Id": 15,
        "RecordInfo": {
            "CreatedBy": null,
            "ModifiedBy": null,
            "CreatedDate": "2018-08-29T17:19:19.363",
            "ModifiedDate": "2018-08-29T17:19:19.363"
        },
        "Price": 3,
        "ShippingEstimatorMarkup": 0,
        "ItemRef": {
            "Id": 239,
            "Name": "Shipping",
            "ExternalId": "800000F0-1430404096"
        },
        "ProxyRef": null,
        "SalesTaxCodeRef": {
            "Id": 2,
            "Name": "Tax"
        },
        "ShippingEstimatorAssignRateToDoc": false,
        "ShippingCarrierEstimatorServiceRef": null,
        "ShippingEstimatorDestinationType": "",
        "IntegrationRef": {
            "Id": "",
            "EditSequence": "",
            "IsSynced": true
        },
        "IsActive": true
    },
    {
        "Name": "UPS",
        "Id": 4,
        "RecordInfo": {
            "CreatedBy": null,
            "ModifiedBy": null,
            "CreatedDate": "2015-01-30T17:22:37",
            "ModifiedDate": "2018-08-29T17:18:55.797"
        },
        "Price": 5.5,
        "ShippingEstimatorMarkup": 0,
        "ItemRef": {
            "Id": 239,
            "Name": "Shipping",
            "ExternalId": "800000F0-1430404096"
        },
        "ProxyRef": null,
        "SalesTaxCodeRef": {
            "Id": 2,
            "Name": "Tax"
        },
        "ShippingEstimatorAssignRateToDoc": false,
        "ShippingCarrierEstimatorServiceRef": {
            "Id": 4,
            "Name": "UPS Ground"
        },
        "ShippingEstimatorDestinationType": "US",
        "IntegrationRef": {
            "Id": "80000003-1422656557",
            "EditSequence": "1422656557",
            "IsSynced": true
        },
        "IsActive": true
    }
]
```

When filtering by CustomFields, the PropertyName should take the following form.   CustomField.{custom field name}: E.g.

```
 {
      "PropertyName": "CustomField.SDCust2",
      "FieldType": 1,
     "FilterValueArray": "BOB"
  }
```