

LeadGen 

Consumer



Consumer

Web Service

Reference Guide

Melissa Data Corporation

Copyright

Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Melissa Data Corporation. This document and the software it describes are furnished under a license agreement, and may be used or copied only in accordance with the terms of the license agreement.

Copyright © 2015 by Melissa Data Corporation. All rights reserved.

Information in this document is subject to change without notice. Melissa Data Corporation assumes no responsibility or liability for any errors, omissions, or inaccuracies that may appear in this document.

Trademarks

LeadGen Cloud API and the Consumer Web Service are trademarks of Melissa Data Corp. Windows is a registered trademark of Microsoft Corp.

The following are registered trademarks of the United States Postal Service®: United States Postal Service, USPS, ZIP, ZIP Code, and ZIP + 4.

All other brands and products are trademarks of their respective holder(s).

Melissa Data Corporation

22382 Avenida Empresa
Rancho Santa Margarita, CA 92688-2112
Phone: 1-800-MELISSA (1-800-635-4772)
Fax: 949-589-5211

E-mail: info@MelissaData.com
Internet: www.MelissaData.com

For the most recent version of this document, visit
<http://www.melissadata.com/>

Document Code: LSTWSCLRG
Revision Number: 23062015.17

Table of Contents

Using the Consumer Web Service 1

 Calling the Consumer Web Service 1

 The REST Protocol 2

 Request Format 3

 ZIP Code™ 3

 City 4

 County 4

 State 4

 Radius 5

 Street 5

 Cities by County 6

 Cities by State 6

 Counties by State 6

 States 7

 ZIPs by City 7

Options 8

 Shared 8

 Append 9

 Geo Selects 10

 Demographic Selects 10

Returns and Examples 18

 Returns 18

 Examples 20

Errors 24

 Error Codes 24

Using the Consumer Web Service

The Consumer Web Service is a REST based Web service that can be used to select, get counts, and purchase targeted consumer mailing lists, in realtime, anytime.

Geographic Selects Include:	Demographic Selects Include:	
Zip Code	Household Income	Number of Persons in Household
City	Contact Age Group	Number of Adults in Household
County	Home Owner/Renter	Number of Children in Household
Radius	Marital Status	Household Credit Card Info
Street	Length of Residency	Mail Responsive Info
State	Household Age	Net Worth
Target Home Valuation Mode In 1,000s		

The LeadGen Cloud APIs are available to mailers for private, in-house use, or as a portal for your customers to use to make their list selections. You can custom-design a website to meet your own needs, as well as those of your customers.

Take advantage of your buying power with Melissa Data, and pass the discounted prices, along with an impressive selection of mailing lists and sales leads, onto your customers. They'll appreciate the convenience, the price, and your service.

Please note that the Consumer Web Service requires a Customer ID that has been activated for the Consumer Web Service.

Calling the Consumer Web Service

The Consumer Web Service has two method calls:

1. Get Counts Call - This call returns an XML document showing the record counts for a specific request, allowing you to see how many records are available for the selected parameters.

2. Buy Lists Call - This call initiates the purchase of the list detailed by the request. This call is similar to the Get call, but has additional information returned. This information is the order ID, number of usage, and download url. If the purchase of the list was successful, you will receive a link to the list file in the format you selected in the options via email. To specify how many records you want to buy, use the qty parameter. If no quantity is specified the maximum and default is 100,000 for all file types, except for excel and csv files (that have a 65,535 record limit).

The REST Protocol

The Consumer Web Service uses the REST protocol, which uses an HTTP query string to pass a request with selected options. An HTTPS query works just the same as an HTTP query.

Using REST may require that you encode certain characters using the proper URL entities before adding them to a URL. Some characters like slashes and ampersands, if they exist as literals in the original data, must be replaced by special codes so they do not conflict with parts of the REST query.

The following table shows the replacement codes for the most common characters.

Character	URL Code
Space	%20 or +
*	%2A
#	%23
&	%26
%	%25
\$	%28
+	%2B
, (comma)	%2C
/	%2F
:	%3A
;	%3B
<	%3C
=	%3D

Character	URL Code
>	%3E
?	%3F
@	%40
[%5B
]	%5D
~	%7E

Request Format

The REST Protocol has a specific format for all requests. The basic format is:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/  
Request Type?Customer ID& Options
```

An example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
zip?id=customer@domain.com&zip=926882112,92612
```

This example uses the REST protocol with the Consumer Web Service to access Melissa Data's databases to make a Get call with a ZIP Request, tied to the customer@domain.com account, for ZIP Code 92688-2112 and 92612.

Geo Selects has the following request types.

ZIP Code™

Get counts and buy lists can be requested for one or more ZIP™ codes. This takes +4 and returns on a +4 level. To use more than one ZIP Code, comma separate them with no spaces. If you have the +4 for a ZIP code, append it without a dash. For example: "926882212,92612"

Get Counts Call

Syntax:

```
/get/zip?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
zip?id=customer@domain.com&zip=926882112,92612
```

Buy List Call

Syntax:

```
/buy/zip?
```


Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/buy/  
zip?id=customer@domain.com&zip=926882112,92612
```

City

City requests will return records from a specified city within a specified state, from selected ZIP codes. Multiple entries are delimited with a “,” comma.

Get Counts Call**Syntax:**

```
/get/city?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
city?id=customer@domain.com&city=ca;irvine,tx;dallas
```

Buy List Call**Syntax:**

```
/buy/city?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/buy/  
city?id=customer@domain.com&city=ca;irvine,tx;dallas
```

County

County requests will return records from a specified county within a specified state. Multiple entries are delimited with a “,” comma.

Get Counts Call**Syntax:**

```
/get/county?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
county?id=customer@domain.com&county=ca;lake,ak;anchorage
```

Buy List Call**Syntax:**

```
/buy/county?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/buy/  
county?id=customer@domain.com&county=ca;lake,ak;anchorage
```

State

State requests will return records from a specified state. Multiple entries are delimited with a “,” comma.

Get Counts Call**Syntax:**

/get/state?

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
state?id=customer@domain.com&st=ca,ak
```

Buy List Call**Syntax:**

/buy/state?

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/buy/  
state?id=customer@domain.com&st=ca,ak
```

Radius

Radius requests will return the nearest records to the selected address, up to 100,000 records (65,535 when requesting Excel files).

Get CountsCall**Syntax:**

/get/radius?

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
radius?id=customer@domain.com&addr=22382 avenida  
empresa&zip=92688&records=100&mile=1
```

Buy List Call**Syntax:**

/buy/radius?

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/Buy/  
radius?id=customer@domain.com&addr=22382 avenida  
empresa&zip=92688&records=100&mile=1
```

Street

Street requests will return records from a specified street within a specified ZIP Code.

Get Counts Call**Syntax:**

/get/street?

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
street?id=customer@domain.com&zip=92688&str=empresa
```

Buy List Call**Syntax:**

```
/buy/street?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/Buy/  
Street?id=customer@domain.com&zip=92688&str=empresa
```

Others

The following requests to the Consumer Web Service will return lists of cities, counties, states, and ZIP Codes.

Cities by County

Cities by County request returns a complete list of cities within the specified U.S. County.

Get Counts Call**Syntax:**

```
/get/CitiesByCounty?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
CitiesByCounty?id=customer@domain.com&st=ca&county=orange
```

Cities by State

Cities by State request returns a complete list of cities within the specified U.S. State.

Get Counts Call**Syntax:**

```
/get/CitiesByState?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
CitiesByState?id=customer@domain.com&st=ak
```

Counties by State

Counties by State request returns a complete list of counties within the specified U.S. State.

Get Counts Call**Syntax:**

```
/get/CountiesByState?
```

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
CountiesByState?id=customer@domain.com&st=ak
```

States

States request returns a complete list of Sates within the United States of America.

Get Counts Call

Syntax:

`/get/States?`

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
States?id=customer@domain.com
```

ZIPs by City

ZIPs by City request returns a complete list of ZIP Codes within the specified City. State is separated from city with a “;” semi-colon. Multiple entries are delimited with a “,” comma.

Get Counts Call

Syntax:

`/get/ZipsByCity?`

Example:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/  
ZipsByCity?id=customer@domain.com&city=ca;irvine,ca;tustin
```

Options

Shared

Geo Selects and Demographic Selects both have the following options:

id..... Required.

User ID or Email address.

multi..... Defines the multiple usage of a list. Default value is 1, may be increased up to 5.

qty..... Specifies how many records to buy. Default (and Maximum) value is 100,000. Excel and CSV formats are limited to 65,535.
If a request includes multiple geographic areas (ZIP codes, cities, counties, or states), and the quantity requested is less than the actual number of records in the areas, returned records would be randomly spread across the areas.

file..... Defines the desired file format of a purchased list.

Value	File Format
1	Zipped .dbf
2	Zipped .txt
3	Zipped .csv
5	Zipped .xls (Excel)
6	.dbf
7	.txt
8	.csv
10	.xls (Excel)

dwell Advantage Dwelling Type. Default value is 1.

Value	Advantage Dwelling Type
1	Single and Multi-Family
2	Single Family Only
3	Multi-Family Only

gender Defines the gender. Default value is F+M.

Value	Gender
F	Female
M	Male

po Defines the customer's purchase order number.
May be defined in buy calls.

zip4 True (1): Return only records with zip+4, resulting in fewer records.
False (0): **Default**. Returns all records.

debug Set to 1 to have every parameter in the url spell-checked.

Append

The following are boolean values that can be appended.

one True (1): **Default**. One contact per address.
False (0): All contacts' information.

phone True (1): Append phone number.
False (0): **Default**. Do not append phone number.

birth True (1): Append birth date.
False (0): **Default**. Do not append birth date.

fam True (1): Append family relationship.
False (0): **Default**. Do not append family relationship.

struct True (1): Append structure year.
False (0): **Default**. Do not append structure year.

Geo Selects

Geographic Selects have the following options:

str Street Type: Required.

If exact street name cannot be found, similar street names will be returned. Ex: If no street called “Gilberto” is found, similar streets like “Gilberto Ave” and “Gilberton” will be returned.

Radius Type: Optional.

Exact street name will be searched within the radius. If nothing found, nothing is returned.

records Radius Only. Required.

Defines the number of records to be returned.

mile Radius Only.

Defines the number of miles from **&addr** and **&zip** a record can be returned. If used with **&records** the search will be for the specified number of records within the given number of miles. Maximum value is 10.

strzip Radius Only.

All addresses within the radius and with this zip will be returned.

Demographic Selects

Demographic Selects have the following options:

hInc Household Income. Will be appended to file.

hInc-d Household Income. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Income Range
1	Less than \$14,999
2	\$15,000 - \$19,999
3	\$20,000 - \$29,999
4	\$30,000 - \$39,999
5	\$40,000 - \$49,999
6	\$50,000 - \$74,999

Value	Income Range
7	\$75,000 - \$99,999
8	\$100,000 - \$124,999
9	\$125,000 - \$149,999
10	\$150,000 - \$174,999
11	\$175,000 - \$199,999
12	\$200,000 - \$249,999
13	\$250,000 or more.

cAge. Contact Age Group. Will be appended to file.

cAge-d. Contact Age Group. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Age Group
1	18-20 years
2	21-25 years
3	26-30 years
4	31-35 years
5	36-40 years
6	41-45 years
7	46-50 years
8	51-55 years
9	56-60 years
10	61-65 years
11	66-70 years
12	71-75 years
13	76+ years

ownRent Home Owner/ Renter. Will be appended to file.

ownRent-d Home Owner/ Renter. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Owner/Renter Status
1	Home Owner
2	Renter

marital. Marital Status. Will be appended to file.

marital-d Marital Status. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Marital Status
1	Unknown
2	Household contains at least 1 married person
3	Household contains at least 1 single person
4	Household contains both married & single persons

resLen Length of Residency. Will be appended to file.

resLen-d Length of Residency. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Time Range
1	0-6 Months
2	7-12 Months
3	1-2 Years
4	3-5 Years
5	6-10 Years
6	11-15 Years
7	16-20 Years

Value	Time Range
8	20+ Years

hAge. Household Age Code. Will be appended to file.

hAge-d. Household Age Code. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Age Range
1	18-24 Years Old
2	25-34 Years Old
3	35-44 Years Old
4	45-54 Years Old
5	55-64 Years Old
6	65-74 Years Old
7	75+ Years Old

people. Total number of people in Household. Will be appended to file.

people-d. Total number of people in Household. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Number of persons in Household
1	1 Person
2	2 Persons
3	3 Persons
4	4 Persons
5	5 Persons
6	6 Persons
7	7 Persons

Value	Number of persons in Household
8	8 Persons
9	9+ Persons

Adults Number of Adults in Household. Will be appended to file.

Adults-d Number of Adults in Household. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Number of Adults in Household
1	1 Adult
2	2 Adults
3	3 Adults
4	4 Adults
5	5+ Adults

Kids Number of Children in Household. Will be appended to file.

Kids-d Number of Children in Household. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Number of Children in Household
1	1 Child
2	2 Children
3	3 Children
4	4 Children
5	5 Children
6	6 Children
7	7 Children

Value	Number of Children in Household
8	8 Children
9	9+ Children

cCard Household Credit Card Info. Will be appended to file.

cCard-d Household Credit Card Info. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Type of Credit Card
1	Miscellaneous Credit Card
2	Standard Retail Card
3	Standard Specialty Card
4	Upscale Retail Card
5	Upscale Specific Retail Card
6	Bank Card
7	Oil/Gas Card
8	Finance Company Card
9	Travel/Entertainment Card

mail Mail Responsive Info. Will be appended to file.

mail-d Mail Responsive Info. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Mail Responsive Info
1	Mail Buyer Indicator
2	Mail Donor Indicator

worth. Net Worth. Will be appended to file.

worth-d. Net Worth. With no input, it will include all. When 0, this option will not be considered. May be dash delimited to include specific ranges.

Value	Net Worth
1	Less than \$25,000
2	\$25,000 - \$49,999
3	\$50,000 - \$74,999
4	\$75,000 - \$99,999
5	\$100,000 - \$149,999
6	\$150,000 - \$249,999
7	\$250,000 - \$499,999
8	\$500,000 - \$749,999
9	\$750,000 - \$999,999
10	\$1,000,000 or More

home. Target Home Valuation Mode In 1,000s. Will be appended to file.

home-d. Target Home Valuation Mode In 1,000s. With no input, it will include all. When 0, this option will not be considered.

Value	Home Value
1	\$1 - \$49,999
2	\$50,000 - \$99,999
3	\$100,000 - \$149,999
4	\$150,000 - \$199,999
5	\$200,000 - \$249,999
6	\$250,000 - \$299,999
7	\$300,000 - \$349,999
8	\$350,000 - \$399,999
9	\$400,000 - \$449,999

Value	Home Value
10	\$450,000 - \$499,999
11	\$500,000 - \$574,999
12	\$575,000 - \$649,999
13	\$650,000 - \$724,999
14	\$725,000 - \$799,999
15	\$800,000 - \$899,999
16	\$900,000 - \$999,999
17	\$1,000,000 - \$1,999,999
18	\$2,000,000 or More

Returns and Examples

Returns

Upon completing a Get Count Call, the Consumer Web Service returns an XML document including the following elements detailing the call:

Tag Code	Tag Description
<Consumer>	Tag encapsulating the whole XML document.
<Geography>	Geo Selects.
<Address>	Requests within Geo Selects. Tags returned depend on Geo type.
<City>	
<County>	
<State>	
<ZIP>	
<Records>	Returned if Radius is called with &records specification.
<Miles>	Returned if Radius is called with &mile specification.
<Options>	Demographic Selects Options.
<ContactPersonInfo>	Specific options tag within Demographic Selects.
<ContactPersonInfo>	
<BuildingType>	
<Details>	Depends on request options. Tags included from this tag to <TargetHomeValuationModelIn1000s> inclusive.
<HouseholdIncome>	
<IncludeAll> OR <Details>	
<AppendToFile>	
<ContactAgeGroup>	

Tag Code	Tag Description
<HomeOwnerRenter>	
<MaritalStatus>	
<LengthOfResidentCode>	
<HouseholdAgeCode>	
<Gender>	
<NumPersonsInHousehold>	
<NumAdultsInHousehold>	
<NumChildrenInHousehold>	
<HouseholdCreditCardInfo>	
<MailResponsiveInfo>	
<NetWorthModel>	
<TargetHomeValuationModeIn1000s>	
<AdditionalAppend>	
<CountDetails>	Contains a breakdown of the total count.
<StreetRange>	Contains a breakdown of the street range.
<Street>	Individual streets.
<StartNumber>	For Geo types radius and street. Shows the starting street number and ending street number in the count.
<EndNumber>	
<Geography>	Location of count detail. The location varies by the request type: <ul style="list-style-type: none"> • ZIP type returns Zip plus 4. • City type returns city, state. • County type returns county, state. • State type returns state. • Street type returns street, city, state. • Radius type returns street, city, state.
<Zip>	For radius and street types.
<Count>	Number of records that <StartNumber>, <EndNumber>, <Geography>, and <Zip> apply to.

Tag Code	Tag Description
<Addresses>	<ul style="list-style-type: none"> Returned with geo type Street. Only returned with Radius when strzip is specified. Contains a breakdown of the addresses returned.
<Address>	Individual addresses.
<TotalCount>	Total number of records returned with the selected options.
<Order>	Buy Requests Only. Order details for a Buy request.
<Id>	Order Id.
<Usage>	
<DownloadURL>	The URL to download the purchased file.
<PONumber>	Returned if po is specified.
<DeliveredQty>	
<RequestedQty>	Returned if &qty is specified.
<Result>	Returns the status code.
<StatusCode>	Indicates the status of the ID. Returns Declined, Approved, or Err. If Err is returned, see Errors tag for greater detail.

Examples

The following code is the input to make a Get call with options:

```
http://list.melissadata.net/V1/Consumer/rest/Service.svc/get/
radius?id=customer@domain.com&addr=22382 avenida empresa
&zip=92688&records=100&dwell=2&hinc-d=1-3-5
```

The following is an example of the XML output from the previous Get call:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<Consumer>
  <Geography>
    <Address>22382 Avenida Empresa</Address>
    <City>Rancho Santa Margarita</City>
    <State>CA</State>
    <Zip>92688</Zip>
    <Records>100</Records>
  </Geography>
  <Options>
```

```

<ContactPersonInfo>
  <ContactPersonInfo>One per Address</ContactPersonInfo>
</ContactPersonInfo>
<BuildingType>
  <Details>Single Family Only</Details>
</BuildingType>
<HouseholdIncome>
  <Details>$0 - 14,999;$20,000 - 29,999;$40,000 - 49,999
  </Details>
  <AppendToFile>False</AppendToFile>
</HouseholdIncome>
<ContactAgeGroup>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</ContactAgeGroup>
<HomeOwnerRenter>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</HomeOwnerRenter>
<MaritalStatus>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</MaritalStatus>
<LengthOfResidentCode>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</LengthOfResidentCode>
<HouseholdAgeCode>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</HouseholdAgeCode>
<Gender>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</Gender>
<NumPersonsInHousehold>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</NumPersonsInHousehold>
<NumAdultsInHousehold>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</NumAdultsInHousehold>
<NumChildrenInHousehold>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</NumChildrenInHousehold>
<HouseholdCreditCardInfo>
  <IncludeAll>True</IncludeAll>
  <AppendToFile>False</AppendToFile>
</HouseholdCreditCardInfo>

```

```

    <MailResponsiveInfo>
      <IncludeAll>True</IncludeAll>
      <AppendToFile>False</AppendToFile>
    </MailResponsiveInfo>
    <NetWorthModel>
      <IncludeAll>True</IncludeAll>
      <AppendToFile>False</AppendToFile>
    </NetWorthModel>
    <TargetHomeValuationModeIn1000s>
      <IncludeAll>True</IncludeAll>
      <AppendToFile>False</AppendToFile>
    </TargetHomeValuationModeIn1000s>
    <AdditionalAppend/>
  </Options>
  <CountDetails>
    <StreetRange>
      <Street>
        <StartNumber>60</StartNumber>
        <EndNumber>96</EndNumber>
        <Geography>ABRIGO , Rancho Santa Margarita CA
        </Geography>
        <Count>5</Count>
      </Street>
      ...
    </StreetRange>
    <TotalCount>
      <Count>100</Count>
    </TotalCount>
    <Result>
      <StatusCode>Approved</StatusCode>
    </Result>
  </CountDetails>
</Consumer>

```

The following code is the input to make a Buy call with options:

```

http://list.melissadata.net/V1/Consumer/rest/Service.svc/buy/
radius?id=customer@domain.com&addr=22382 avenida empresa
&zip=92688&records=100&dwell=2&hinc-d=1-3-5

```

A Buy request returns everything a Get request does, including an order tag that has Id information and a download URL.

The following is an example of the Order tag XML output from the previous Buy call:

```
....  
</TotalCount>  
<Order>  
  <Id>1234567</Id>  
  <Usage>1</Usage>  
  <DownloadURL>  
    ftp://www.url.com/location/filename.type  
  </DownloadURL>  
  <DeliveredQty>100</DeliveredQty>  
</Order>  
<Result>  
....
```

Errors

If the Result element returns “Err,” the Consumer Web Service will return one or more Error elements within an Errors element detailing the cause of the problem.

```
<Errors>
  <Error>
    <ErrorCode>100</ErrorCode>
    <ErrorDescription>Unrecognized ZIP Code. #####</ErrorDescription>
  </Error>
</Errors>
```

Error Codes

The following is a list of the available error codes:

Error Code	Error Description
100	Unrecognized ZIP Code.
101	Unrecognized city or state.
102	Unrecognized county or state.
103	Unrecognized address.
104	User ID or password not recognized.
106	Invalid user information.
108	Order failed, please try later.
109	Insufficient geographic input.
111	Request exceeds 100,000 record maximum.
112	Unrecognized state.
113	Error, please try again.
115	For the Radius geography type, please enter a number of records.
116	Sorry, you don't have permission to access this service.
117	ZIP Code is not a valid input for the requested geography type.

Error Code	Error Description
121	The list cannot be used more than 5 times.
122	Order count exceeds 65,535 record maximum for Excel files.
123	Invalid option.
124	Order count exceeds 65,535 record maximum for comma limited files.