



ZIP Code Lookup & Address Verification Component

The Software Company, Inc.
www.SoftwareCompany.com



NetZipCode for .NET allows you to quickly and easily build ZIP Code lookup, address verification plus address correction and parsing into your custom applications. The delivery address is checked for proper Postal Service format and standardized if necessary. Spelling corrections are made and the address is split into USPS standard components.

NetZipCode for .NET not only returns a standardized and corrected address but, also provides the following additional information about each address:

- ZIP+4 Add-On
- Congressional District
- County Name
- Delivery Point Bar Code
- Time Zone Including Daylight Savings Time Flag

An array of diagnostic flags is returned with each address letting you know exactly what strategy was used in order to achieve the best match.

Benefits

- **Save \$\$\$ on Postage** - verify addresses before you ship or mail
- **Catch Data-Input Errors** – virtually eliminating undeliverable addresses
- **Save Keying Time** - city/state is returned from 5-digit ZIP Code
- **Unlimited Use** - no escalating charges
- **Free Database Updates** for a full year

Features

- Addresses are Standardized to USPS Recommended Abbreviations
- No Automatic Database Expiration
- Proper Case Conversion for More Attractive Data Presentation
- Attractive USPS Database Subscription Pricing
- Designed for Use with All .NET Compatible Programming Languages

NetZipCode for .NET follows the US Postal Service “one component failure rule” to find a correct address match. This rule permits an address match if no more than one address component mismatch exists, where a “component” is defined as a predirectional, street name, street suffix, or a postdirectional.

Consider the following address: 100 E MAIN ST N

This address has all four components. If one component can be added, changed, or deleted to achieve a unique match, NetZipCode will correct the address and return the ZIP+4. Under current USPS address matching rules, if more than one component is incorrect, no match is allowed. If adding, changing or deleting a component results in multiple matches, no match can be returned. The priority is to add a suffix first, before adding a directional. If there is no match, change or delete the suffix. Finally, change or delete a directional.

If there is no exact address match within the city, NetZipCode applies the “one component failure rule” within the input ZIP Code. If there is still no match, then NetZipCode applies the “one component failure rule” to all addresses within the postal finance number. If there is still no match, the address is rejected.

Example

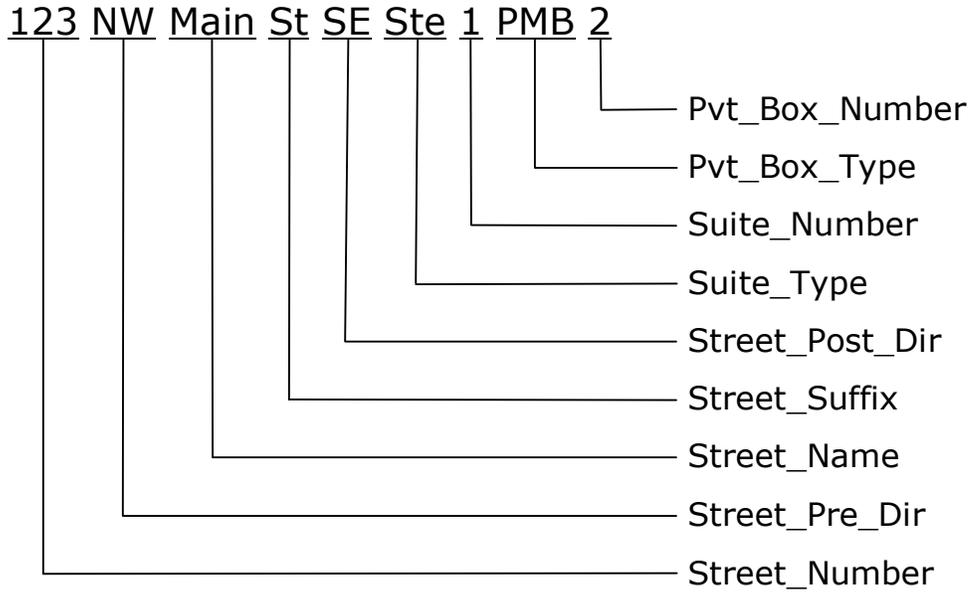
Input Address: 4950 INTNL
32819

Output Address: 4950 International Dr
Orlando, FL 32819-9415
County: Orange
Time Zone: EST

Results:

- Street name corrected
- Missing street suffix added
- City/state names added
- ZIP+4 “add-on” added
- County name and time zone identified

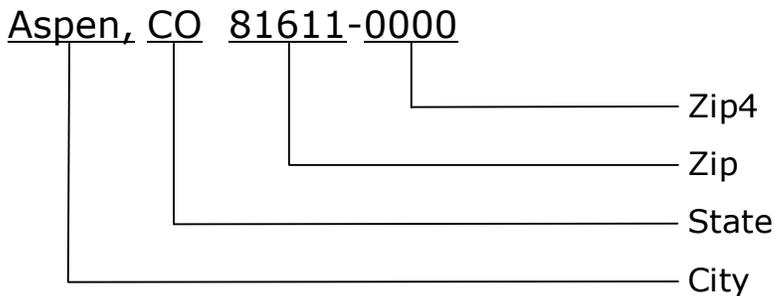
Street Address



Box Address



City/State/Zip



Address_In

Syntax: Address_In = String

Description:

Set this property to the address string to be processed. When the “Lookup” method is invoked, the Address_In and CSZ_In strings are matched to the USPS National Address Database. If a match is found, the standardized and corrected address is placed into the Address_Out and CSZ_Out properties. In addition, each element of the Address_In and CSZ_In strings is placed into the appropriate address and city/state/zip component property.

CSZ_In

Syntax: CSZ_In = String

Description:

Set this property to the city, state and/or ZIP Code string to be processed. When the “Lookup” method is invoked, the CSZ_In string is matched to the USPS National Address Database. If a match is found, the standardized and corrected city/state/zip is placed into the CSZ_Out property. In addition, each element of the CSZ_In string is placed into the appropriate city/state/zip component property. *If this property is set to the city and state, the ZIP Code is optional. If set to the 5-digit ZIP Code, the city and state are optional.*

Firm_Name_In (optional)

Syntax: Firm_Name_In = String

Description:

Set this property to the Firm or High-rise name string to be processed. This will produce the highest level of match for the most precise 4-digit ZIP+4 “add-on”.

PR_Urb_In (optional)

Syntax: PR_Urb_In = String

Description:

Set this property to the Puerto Rican urbanization string to be processed.

Capitalization (changed in v4.4+)
(replaces obsolete "Output_Case" property)

Syntax: Capitalization = StringLiteral

Description:

Set this property to "Upper", "Lower" or "Mixed" to indicate your capitalization preference for the output address and its components. **Default is "Upper"**.

Pvt_Box_Types (new in v4.4+)

Syntax: Pvt_Box_Types = StringArray

Description:

Set this property to a list of private box types you want NetZipCode to recognize. Some common types are: "PMB", "MAC", etc. See "*Pvt_Box_Type*" and "*Pvt_Box_Number*" properties.

Database_Path

Syntax: Database_Path = String

Description:

Set this property to the full path to the USPS National Address Database files. These files are located in the NetZipCode installation folder under the names: "Z1.dat" thru "Z8.dat". You can relocate these files to any other folder as long as you set this property to the full path. **Default is first the folder of the invoking application: "AppDomain.CurrentDomain.BaseDirectory" then the NetZipCode installation folder.**

Static_Key_Name

Syntax: Static_Key_Name = String

Description:

Set this property to the name portion of the static key assignment or blank.

Static_Key

Syntax: Static_Key = String

Description:

Set this property to the key portion of the static key assignment or blank.

Address_Out (read only)

Syntax: String = Address_Out

Description:

After invoking the “Lookup” method, this property will contain the standardized and corrected address string from the Address_In property including a suite number if present. If no match was found in the USPS database, this property will be blank.

Address_Out_Street (read only)

Syntax: String = Address_Out_Street

Description:

After invoking the “Lookup” method, this property will contain the street portion of Address_Out.

Address_Out_Suite (read only)

Syntax: String = Address_Out_Suite

Description:

After invoking the “Lookup” method, this property will contain the suite portion of Address_Out.

Street_Number (read only)

Syntax: String = Street_Number

Description:

After invoking the “Lookup” method, this property is set to the primary address number component of Address_Out commonly referred to as house number, street number or range.

Street_Pre_Dir (read only)

Syntax: String = Street_Pre_Dir

Description:

After invoking the “Lookup” method, this property is set to the Predirectional component of Address_Out. Values will be a valid directional (N, NE, S, SE, etc.) or blank.

Street_Name (read only)

Syntax: String = Street_Name

Description:

After invoking the “Lookup” method, this property is set to the Street Name component of Address_Out. Value will be alphanumeric.

Street_Suffix (read only)

Syntax: String = Street_Suffix

Description:

After invoking the “Lookup” method, this property is set to the Street Suffix component of Address_Out. Values will be a valid suffix (St, Ave, Rd, etc.) or blank.

Street_Post_Dir (read only)

Syntax: String = Street_Post_Dir

Description:

After invoking the “Lookup” method, this property is set to the Postdirectional component of Address_Out. Values will be a valid directional (N, NE, S, SE, etc.) or blank.

Suite_Type (read only)

Syntax: String = Suite_Type

Description:

After invoking the “Lookup” method, this property is set to the Suite Type component of Address_Out. Values will be only valid suite types (Apt, Suite, Unit, etc.) or blank.

Suite_Number (read only)

Syntax: String = Suite_Number

Description:

After invoking the “Lookup” method, this property is set to the Suite Number component of Address_Out. Values will be alphanumeric suite number or blank.

Box_Type (read only)

Syntax: String = Box_Type

Description:

After invoking the “Lookup” method, this property is set to the Box Type component of Address_Out. Values will be only valid box types (PO Box, RR, HC, etc.) or blank.

Box_Type_Number (read only)

Syntax: String = Box_Type_Number

Description:

After invoking the “Lookup” method, this property is set to the Box Type Number component of Address_Out. Values will be alphanumeric box type number or blank.

Box (read only)

Syntax: String = Box

Description:

After invoking the “Lookup” method, this property is set to the Box component of Address_Out. Value will be “Box” or blank.

Box_Number (read only)

Syntax: String = Box_Number

Description:

After invoking the “Lookup” method, this property is set to the Box Number component of Address_Out. Values will be alphanumeric box number or blank.

Pvt_Box_Type (read only) (new in v4.4+)

Syntax: String = Pvt_Box_Type

Description:

After invoking the “Lookup” method, this property is set to the Private Box Type component of Address_Out. Values will be only valid box types as specified in the Pvt_Box_Types property.

Pvt_Box_Number (read only) (new in v4.4+)

Syntax: String = Pvt_Box_Number

Description:

After invoking the “Lookup” method, this property is set to the Private Box Number component of Address_Out. Values will be alphanumeric private box number or blank.

CSZ_Out (read only)

Syntax: String = CSZ_Out

Description:

After invoking the “Lookup” method, this property will contain the standardized city/state/zip (last line) from the CSZ_In property.

City (read only)

Syntax: String = City

Description:

After invoking the “Lookup” method, this property is set to the City component of CSZ_Out. See “Preferred_City_Name” property.

State (read only)

Syntax: String = State

Description:

After invoking the “Lookup” method, this property is set to the State component of CSZ_Out. Values will be only valid USPS state abbreviations (FL, AZ, CO, etc.) or blank.

Zip (read only)

Syntax: String = Zip

Description:

After invoking the “Lookup” method, this property is set to the 5-digit ZIP Code component of CSZ_Out. Values will be a 5-digit numeric ZIP Code or blank.

Zip4 (read only)

Syntax: String = Zip4

Description:

After invoking the “Lookup” method, this property is set to the 4-digit ZIP+4 “add-on” component of CSZ_Out. Values will be a 4-digit numeric ZIP+4 “add-on” code (sector/segment) or blank.

Preferred_City (read only)

Syntax: String = Preferred_City

Description:

After invoking the “Lookup” method, this property is set to the “preferred” city name for the 5-digit “Zip” property. See “*Preferred_City_Name*” and “*Preferred_State*” properties.

Preferred_State (read only)

Syntax: String = Preferred_State

Description:

After invoking the “Lookup” method, this property is set to the “preferred” state abbreviation for the 5-digit “Zip” property. Values will be valid USPS state abbreviations (FL, AZ, CO, etc.) or blank. See “*Preferred_City_Name*” and “*Preferred_City*” properties.

Match_Count (read only)

Syntax: Integer = Match_Count

Description:

After invoking the “Lookup” method, this property is set to an integer count of the number of matching addresses when the “Multiple_Match” property is set to true. See “*GetFirstMatch*”, “*GetNextMatch*” methods and “*Multiple_Match*” property.

Address_Type (read only)

Syntax: String = Address_Type

Description:

After invoking the “Lookup” method, this property is set to one of the following:

S	Street	(1 N Main St, 2 US Highway 285, etc.)
A	Suite	(Apt 1, Suite 2, etc.)
P	Post Office Box	(PO Box 1)
R	Rural Route	(RR 1 Box 2)
H	Highway Contract	(HC 1 Box 2)
G	General Delivery	(General Delivery, Gen Del, GD, etc.)
M	Military	(CMR 1 Box 2, etc.)
N	Not a valid address	Address_Match will also be set to “False”

Zip_Type (read only)

Syntax: String = Zip_Type

Description:

After invoking the “Lookup” method, this property is set to one of the following:

P	PO Box Only	(no street addresses in this ZIP Code)
M	Military Zip	(state codes AA, AE or AP)
U	Unique Zip	(assigned to a single business or organization)
S	Standard Zip	(any combination of street addresses and PO Boxes)
N	Not a valid Zip	(CSZ_Match will also be set to “False”)

County (read only)

Syntax: String = County

Description:

After invoking the “Lookup” method, this property is set to the name of the county in which the address resides. Values will be alphanumeric county name or blank.

County_FIPS (read only)

Syntax: String = County_FIPS

Description:

After invoking the “Lookup” method, this property is set to the Federal Information Processing Standard (FIPS) code for the county in which the address resides. Values will be a three-digit numeric string or blank.

State_FIPS (read only)

Syntax: String = State_FIPS

Description:

After invoking the “Lookup” method, this property is set to the Federal Information Processing Standard (FIPS) code for the state in which the address resides. Values will be a two-digit numeric string or blank.

Time_Zone (read only)

Syntax: String = Time_Zone

Description:

After invoking the “Lookup” method, this property is set to the most prevalent time zone within the ZIP Code area. Values will be alphanumeric time zone or blank.

Time_Zone_DST (read only)

Syntax: Time_Zone_DST = True

Description:

After invoking the “Lookup” method, this property is set to Boolean (True/False) indicating that addresses within the ZIP Code area participate in daylight savings time. Value will be Boolean (True/False).

DPBC (read only)

Syntax: String = DPBC

Description:

After invoking the “Lookup” method, this property is set to the 12-digit Delivery Point Bar Code string for this ZIP Code. Values will be numeric DPBC or blank.

Congressional_District (read only)

Syntax: String = Congressional_District

Description:

After invoking the “Lookup” method, this property is set to the congressional district code for this address. Values will be alphanumeric congressional district or blank.

Firm_Name_Out (read only)

Syntax: String = Firm_Name_Out

Description:

After invoking the “Lookup” method, this property is set to the standardized firm or high-rise building name from the optional Firm_Name_In string. Values will be alphanumeric or blank.

PR_Urb_Out (read only)

Syntax: String = PR_Urb_Out

Description:

After invoking the “Lookup” method, this property is set to the Puerto Rican Urbanization name from optional the PR_Urb_In string. Values will be alphanumeric or blank.

Address Flags (read only)

Description:

After invoking the “Lookup” method, the following Boolean (True/False) properties are returned indicating the status of the address lookup.

Syntax:

Address_Match = True (CSZ_Match is also set to True)

A match was found in the USPS National Address Database for the Address_In and CSZ_In properties. This condition also sets one or more of the following properties indicating any corrections that were made:

Address_Corrected = True

Street_Number, Street_Pre_Dir, Street_Name, Street_Suffix or Street_Post_Dir components were corrected in order to achieve a match.

Suite_Corrected = True

Suite_Type or Suite_Number components were corrected in order to achieve a match.

POB_Corrected = True

Box_Type component was corrected in order to achieve a match.

PR_Urb_Corrected = True

Puerto Rican urbanization component was corrected in order to achieve a match.

CSZ_Match = True

A match was found in the USPS National Address Database for the CSZ_In property. This condition also sets one or more of the following properties indicating any corrections that were made:

City_Corrected = True

City name component was added or corrected in order to achieve a match.

State_Corrected = True

State component was added or corrected in order to achieve a match.

Zip_Corrected = True

ZIP Code component was added or corrected in order to achieve a match.

Zip4_Corrected = True

ZIP+4 component was added or corrected in order to achieve a match.

Address_Match = False

A unique match could not be found in the USPS National Address Database for the Address_In and CSZ_In properties. This condition also sets one or more of the following properties indicating the reason a unique match could not be found:

Multiple_Match = True

More than one address was matched in the USPS National Address Database.

See *"GetFirstMatch"*, *"GetNextMatch"* methods and *"Match_Count"* property.

Street_Number_Unknown = True

The Street_Number was out-of-range for this street in the USPS National Address Database.

Street_Name_Unknown = True

The Street_Name was not matched in the USPS National Address Database.

CSZ_Unknown = True

City State ZIP Code mismatch in the USPS National Address Database.

Insufficient_Address_Data = True

There was not enough data to find a match in the USPS National Address Database.

CSZ_Match = False (CSZ_Unknown is also set to True)

No match was found in the USPS National Address Database for the CSZ_In property.

No_USPS_Delivery = True

Address was verified, but USPS doesn't deliver to this address.

Preferred_City_Name = True

The "City" property returned is the USPS "preferred" city name for the 5-digit ZIP Code.

See *"Preferred_City"* and *"Preferred_State"* properties.

Highrise = True

Address is a high-rise apartment or office building.

Firm = True

Address is a firm (business). When "Firm_Name_In" property is set to a firm name, the firm name is matched to the USPS database and the "Firm" flag is set to True when a match occurs. This is the finest level of match and will, in some cases, return a different ZIP+4 add-on code as a result of the match.

Return_Code (read only)

Syntax: String = Return_Code

Description:

After invoking the “Lookup” method, this property is set to blank upon successful completion. Most exceptions occur on the first invocation. The most common ones are listed below. *This property should be examined on each return from NetZipCode.*

Common Return Codes:

- Z35** One or More Database Files Not Found (*see “Database_Path” property*)
- Z01 – Z16** Database Initialization Failed
- Z17** “Unzdll.dll” Version Does Not Match Database Version
- L00** Evaluation Period Expired
- L01** Static Key Validation Failed (*see “Static_Key” property*)
- L50 – L69** Evaluation License Error

Clear

Syntax: NetZipCode.Clear

Description:

When this method is invoked, all properties are set to null with the exception of `Static_Key`, `Static_Key_Name` and `Database_Path`.

Lookup

Syntax: NetZipCode.Lookup

Description:

When this method is invoked, the `Address_In` and `CSZ_In` properties are compared to the USPS National Address Database. When a match is found, the address is standardized and corrected then placed into the `Address_Out` and `CSZ_Out` properties. In addition, each standardized element of the `Address_Out` and `CSZ_Out` properties is placed into the appropriate address component property. Next, the Address Flags are set along with the `Address_Quality` property to indicate the status of the address lookup as well as how complete and correct the address is. The `Return_Code` property is also set and should be checked after each invocation of the “Lookup” method. *See “Return_Code” property.*

GetFirstMatch

Syntax: NetZipCode.GetFirstMatch

Description:

This method can only be invoked when the “`Multiple_Match`” flag is set to `True`. After invoking the “`GetFirstMatch`” method, the first address from the current group of matching addresses is returned and the “`Address_Match`” flag is set to `True`. *See “Multiple_Match” and “Match_Count” properties.*

GetNextMatch

Syntax: NetZipCode.GetNextMatch

Description:

This method can only be invoked *after* the “`GetFirstMatch`” method has been invoked and when the “`Multiple_Match`” flag is set to `True`. After invoking the “`GetNextMatch`” method, the next address from the current group of matching addresses is returned and the “`Address_Match`” flag is set to `True`. When no more addresses remain, the `Address_Match` property is set to `False`. *See “Multiple_Match” and “Match_Count” properties.*

GetFirstCityInZip

Syntax: NetZipCode.GetFirstCityInZip

Description:

When this method is invoked, the 5-digit CSZ_In property is compared to the USPS National Address Database. When a match is found, the first city name (alphabetically) in the list of cities within this ZIP Code is returned in the City property and the CSZ_Match property is set to "True". If this is the USPS "preferred" city name for this ZIP Code, then the Preferred_City_Name property is set to "True" and the Preferred_City property is set to the city name. See "*Preferred_City_Name*", "*Preferred_City*" and "*Preferred_State*" properties.

GetNextCityInZip

Syntax: NetZipCode.GetNextCityInZip

Description:

This method can only be invoked *after* the "GetFirstCityInZip" method has been invoked and a match was found (CSZ_Match = True). After invoking the "GetNextCityInZip" method, the next city name (alphabetically) in the list of cities within this ZIP Code is returned in the City property and the CSZ_Match property is set to "True". If this is the USPS "preferred" city name for this ZIP Code, then the Preferred_City_Name property is set to "True" and the Preferred_City property is set to the city name. See "*Preferred_City_Name*", "*Preferred_City*" and "*Preferred_State*" properties.

If Installation Doesn't Start Automatically:

- Select **Start > Run** from the Task Bar.
- Type CD/DVD drive letter followed by “:\NetZipCode44.msi” and press enter.

In the folder: “Program Files\The Software Company\NetZipCode 4.4” you will find C# and VB.NET sample programs. There is also a compiled version called: “VBSample.exe” that you can run to demonstrate NetZipCode.

Deploying Your Applications

Be sure to include the following in your deployment package:

NetZipCode.dll – usually placed in the application folder*

Unzdll32.dll – usually placed in the application folder or Windows\System folder*

Z1.dat thru Z8.dat – usually placed in the application folder**

* Other builds can be found in the NetZipCode “Versions” folder.

** “Z1.dat” thru “Z8.dat” database files can be relocated anywhere on the target machine as long as the full path to them is specified in the Database_Path property.

In addition to the above, there is a common runtime that can be placed in the application folder or the Global Assembly Cache (GAC) of the target machine.

Fujitsu.COBOL.dll

Fujitsu.COBOL.InteropServices.Win32.dll

The evaluation license is valid for a period of 30 days or up to 1,000 calls.

Sales@SoftwareCompany.com

303/838-1223 (voice)

303/838-1224 (fax)