

# HOST Function

Finds the host value from a valid URL. Input values must be of URL or String type and can be literals or column references.

In this implementation, a host value includes everything from the end of the protocol identifier (if present) to the end of the extension (e.g. .com).

- For more information, see *Structure of a URL*.

**Wrangle vs. SQL:** This function is part of Wrangle, a proprietary data transformation language. Wrangle is not SQL. For more information, see *Wrangle Language*.

## Basic Usage

### URL literal examples:

```
host('http://www.example.com')
```

**Output:** Returns the value `www.example.com`.

### Column reference example:

```
host(myURLs)
```

**Output:** Returns the host values extracted from the `myURLs` column.

## Syntax and Arguments

```
host(column_url)
```

Argument	Required?	Data Type	Description
column_url	Y	string	Name of column or String or URL literal containing the host value to extract

For more information on syntax standards, see *Language Documentation Syntax Notes*.

### column\_url

Name of the column or URL or String literal whose values are used to extract the host value.

- Missing input values generate missing results.
- Multiple columns and wildcards are not supported.

### Usage Notes:

Required?	Data Type	Example Value
Yes	String literal or column reference (URL)	<code>http://www.example.com</code>

## Examples

**Tip:** For additional examples, see *Common Tasks*.

### Example - Domain, Host, Subdomain, and Suffix functions

This examples illustrates how you can extract component parts of a URL using specialized functions for the URL data type.

#### Functions:

Item	Description
DOMAIN Function	Finds the value for the domain from a valid URL. Input values must be of URL or String type.
SUBDOMAIN Function	Finds the value a subdomain value from a valid URL. Input values must be of URL or String type.
HOST Function	Finds the host value from a valid URL. Input values must be of URL or String type and can be literals or column references.
SUFFIX Function	Finds the suffix value after the domain from a valid URL. Input values must be of URL or String type.
URLPARAMS Function	Extracts the query parameters of a URL into an Object. The Object keys are the parameter's names, and its values are the parameter's values. Input values must be of URL or String type.
FILTEROBJECT Function	Filters the keys and values from an Object data type column based on a specified key value.

#### Source:

Your dataset includes the following values for URLs:

URL
www.example.com
example.com/support
http://www.example.com/products/
http://1.2.3.4
https://www.example.com/free-download
https://www.example.com/about-us/careers
www.app.example.com
www.some.app.example.com
some.app.example.com
some.example.com
example.com
http://www.example.com?q1=broken%20record
http://www.example.com?query=khakis&app=pants
http://www.example.com?q1=broken%20record&q2=broken%20tape&q3=broken%20wrist

#### Transformation:

When the above data is imported into the application, the column is recognized as a URL. All values are registered as valid, even the numeric address.

To extract the domain and subdomain values:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	DOMAIN(URL)
<b>Parameter: New column name</b>	'domain_URL'

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	SUBDOMAIN(URL)
<b>Parameter: New column name</b>	'subdomain_URL'

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	HOST(URL)
<b>Parameter: New column name</b>	'host_URL'

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	SUFFIX(URL)
<b>Parameter: New column name</b>	'suffix_URL'

You can use the Pattern in the following transformation to extract protocol identifiers, if present, into a new column:

<b>Transformation Name</b>	Extract text or pattern
<b>Parameter: Column to extract from</b>	URL
<b>Parameter: Option</b>	Custom text or pattern
<b>Parameter: Text to extract</b>	`{start}%*://`

To clean this up, you might want to rename the column to protocol\_URL.

To extract the path values, you can use the following regular expression:

**NOTE:** Regular expressions are considered a developer-level method for pattern matching. Please use them with caution. See *Text Matching*.

<b>Transformation Name</b>	Extract text or pattern
<b>Parameter: Column to extract from</b>	URL
<b>Parameter: Option</b>	Custom text or pattern
<b>Parameter: Text to extract</b>	/[ <sup>^</sup> *:\//]\./.*\$/

The above transformation grabs a little too much of the URL. If you rename the column to `path_URL`, you can use the following regular expression to clean it up:

<b>Transformation Name</b>	Extract text or pattern
<b>Parameter: Column to extract from</b>	URL
<b>Parameter: Option</b>	Custom text or pattern
<b>Parameter: Text to extract</b>	/[! <sup>^</sup> \//].*\$/

Delete the `path_URL` column and rename the `path_URL1` column to the deleted one. Then:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	URLPARAMS(URL)
<b>Parameter: New column name</b>	'urlParams'

If you wanted to just see the values for the `q1` parameter, you could add the following:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	FILTEROBJECT(urlParams, 'q1')
<b>Parameter: New column name</b>	'urlParam_q1'

### Results:

For display purposes, the results table has been broken down into separate sets of columns.

Column set 1:

URL	host_URL	path_URL
www.example.com	www.example.com	
example.com/support	example.com	/support

http://www.example.com/products/	www.example.com	/products/
http://1.2.3.4	1.2.3.4	
https://www.example.com/free-download	www.example.com	/free-download
https://www.example.com/about-us/careers	www.example.com	/about-us /careers
www.app.example.com	www.app.example.com	
www.some.app.example.com	www.some.app.example.com	
some.app.example.com	some.app.example.com	
some.example.com	some.example.com	
example.com	example.com	
http://www.example.com?q1=broken%20record	www.example.com	
http://www.example.com?query=khakis&app=pants	www.example.com	
http://www.example.com?q1=broken%20record&q2=broken%20tape&q3=broken%20wrist	www.example.com	

Column set 2:

URL	protocol_URL	subdomain_URL	domain_URL	suffix_URL
www.example.com		www	example	com
example.com/support			example	com
http://www.example.com/products/	http://	www	example	com
http://1.2.3.4	http://			
https://www.example.com/free-download	https://	www	example	com
https://www.example.com/about-us/careers	https://	www	example	com
www.app.example.com		www.app	example	com
www.some.app.example.com		www.some.app	example	com
some.app.example.com		some.app	example	com
some.example.com		some	example	com
example.com			example	com
http://www.example.com?q1=broken%20record	http://	www	example	com
http://www.example.com?query=khakis&app=pants	http://	www	example	com
http://www.example.com?q1=broken%20record&q2=broken%20tape&q3=broken%20wrist	http://	www	example	com

Column set 3:

URL	urlParams	urlParam_q1
www.example.com		
example.com/support		
http://www.example.com/products/		
http://1.2.3.4		
https://www.example.com/free-download		

https://www.example.com/about-us/careers		
www.app.example.com		
www.some.app.example.com		
some.app.example.com		
some.example.com		
example.com		
http://www.example.com?q1=broken%20record	{"q1":"broken record"}	{"q1":"broken record"}
http://www.example.com?query=khakis&app=pants	{"query":"khakis","app":"pants"}	
http://www.example.com?q1=broken%20record&q2=broken%20tape&q3=broken%20wrist	{"q1":"broken record", "q2":"broken tape", "q3":"broken wrist"}	{"q1":"broken record"}