

# CHAR Function

Generates the Unicode character corresponding to an inputted Integer value.

**Unicode** is a digital standard for the consistent encoding of the world's writing systems, so that representation of character sets is consistent around the world.

- The first 256 Unicode characters (0, 255) correspond to the ASCII character set.
- Input values for the `CHAR` function should be of integer type. Decimal type column data can be used as input. However, if the data contains digits to the right of the decimal point, the `CHAR` function returns a missing value.
- If the function cannot evaluate the numeric data, a null value is returned.

**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

### Column reference example:

```
char(MyCharIndex)
```

**Output:** Returns the Unicode value for the number in the `MyCharIndex` column.

### String literal example:

```
char(65)
```

**Output:** Returns the string: A.

## Syntax and Arguments

```
char(index_value)
```

Argument	Required?	Data Type	Description
<code>index_value</code>	Y	integer (positive)	Unicode index value of the character

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

### `index_value`

Unicode index value of the character to generate or match.

- The Unicode character set contains up to 1,114,112 characters. Most uses rely on the first 10,000 characters.
- Value must be less than `end_index`.

### Usage Notes:

Required?	Data Type	Example Value
Yes	Integer (non-negative)	65

## Examples

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

### Example - char and unicode functions

In this example, you can see how the `CHAR` function can be used to convert numeric index values to Unicode characters, and the `UNICODE` function can be used to convert characters back to numeric values.

#### Source:

The following column contains some source index values:

index
1
33
33.5
34
48
57
65
90
97
121
254
255
256
257
9998
9999

#### Transformation:

When the above values are imported to the Transformer page, the column is typed as integer, with a single mismatched value (33.5). To see the corresponding Unicode characters for these characters, enter the following transformation:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	<code>CHAR(index)</code>

<b>Parameter: New column name</b>	'char_index'
-----------------------------------	--------------

To see how these characters map back to the index values, now add the following transformation:

<b>Transformation Name</b>	New formula
<b>Parameter: Formula type</b>	Single row formula
<b>Parameter: Formula</b>	UNICODE(char_index)
<b>Parameter: New column name</b>	'unicode_char_index'

**Results:**

index	char_index	unicode_char_index
1		1
33	!	33
33.5		
34	"	34
48	0	48
57	9	57
65	A	65
90	Z	90
97	a	97
122	z	122
254	þ	254
255	ÿ	255
256		256
257		257
9998		9998
9999		9999

Note that the floating point input value was not processed.