
gtin Documentation

Release 0.0.0

David Belais

March 05, 2016

1	gtin.GTIN	3
1.1	Examples	4

A python package for parsing GTINs (“Global Trade Item Numbers”—also known as UPC/EAN/JAN/ISBN).

To install:

```
$ pip install gtin
```

gtin.GTIN

This class represents a Global Trade Item Number, and can be used to:

- Identify a trade item's GCP (GS1 Company Prefix), Item Reference, and Indicator Digit.
- Validate a GTIN's check-digit.
- Calculate a check-digit from a raw GTIN.

Parameters:

gtin A string or number representing a GTIN, including the check-digit.

- When the *gtin* parameter is provided, the last (rightmost) digit is used to validate the GTIN if no value is provided for the parameter *check_digit*.

length The length of the GTIN.

- If no value is passed for *length*, and *gtin* is a *str*—*length* is inferred based on the character length of *gtin*.
- If no value is passed for *length*, *gtin* is *None*, and *raw* is a *str*—*length* is inferred based on the length of *raw* (adding 1, to account for the absent check-digit).
- If no length is passed, and none can be inferred from *gtin* or *raw*, *length* defaults to 14.

raw A string or number representing the GTIN, excluding the check-digit.

- If a value is provided for the parameter *gtin*, this parameter is not used, but is instead derived from *gtin*.

In lieu of passing a complete GTIN, with or without the check-digit, using the above parameters—it is possible to pass the components of the GTIN separately: the indicator digit, GCP (GS1 Company Prefix), item reference, and (optionally) the check-digit.

indicator_digit This is the first (leftmost) digit of a GTIN-14.

- “0” indicates a base unit.
- “1” through “8” are used to define the packaging hierarchy of a product with the same item reference.
- “9” indicates a variable-measure trade item.

gcp The GS1 Company Prefix is a globally unique identifier assigned to a company by GS1 Member Organizations to create the identification numbers of the GS1 System. Company Prefixes, which vary in length, are comprised of a GS1 Prefix and a Company Number.

item_reference The item reference is the part of the GTIN that is allocated by the user to identify a trade item for a given Company Prefix. The Item Reference varies in length as a function of the Company Prefix length.

check_digit A mod-10 algorithm digit used to check for input errors. To understand how this digit is calculated, refer to: <http://www.gs1.org/how-calculate-check-digit-manually>. If this parameter is provided, it is matched against the calculated check-digit, and an error is raised if it does not match the calculated check-digit.

1.1 Examples

```
>>> from gtin import GTIN
```

A *GTIN* initialized without any arguments:

```
>>> print(repr(GTIN()))
GTIN('00000000000000')
```

Typical usage will require converting your *GTIN* to a *str* prior to use in your application.

```
>>> print(str(GTIN()))
00000000000000
```

Given a raw GTIN, the check-digit is calculated and appended.

```
>>> print(str(GTIN(raw='0978289450809')))
09782894508091
```

Given a valid GTIN *str* for *gtin*, the return value of *str(GTIN(gtin))* is equal to *gtin*.

```
>>> print(str(GTIN('04000101613600')))
04000101613600
```

Non-numeric characters are ignored/discarded.

```
>>> print(str(GTIN('0-4000101-6136-0')))
04000101613600
```

Given a an *int* for the parameter *raw*, the length defaults to 14.

```
>>> print(str(GTIN(raw=7447010150)))
00074470101505
```

```
>>> print(str(GTIN(74470101505)))
00074470101505
```

Given a GTIN, and a length:

```
>>> print(str(GTIN(raw=7447010150, length=12)))
074470101505
```

```
>>> print(str(GTIN(74470101505, length=12)))
074470101505
```

```
>>> print(str(GTIN('74470101505', length=14)))
00074470101505
```

Get the GCP of a GTIN:


```
>>> print(GTIN('00041333704647').gcp)
0041333
```

```
>>> print(GTIN('00811068011972').gcp)
081106801
```

```
>>> print(GTIN('00188781000171').gcp)
0188781000
```

Get the component parts of a *GTIN* instance as a tuple containing *GTIN.indicator_digit*, *GTIN.gcp*, *GTIN.item_reference*, and *GTIN.check_digit*:

```
>>> print(tuple(GTIN(raw='0400010161360')))
('0', '4000101', '61360', '0')
```