Prefixed

Release 0.7.0

Jan 25, 2023

Contents

CHAPTER 1

SI (Decimal) Prefixes

Prefix	Name	Base
Q	Quetta	10^{30}
R	Ronna	10^{27}
Y	Yotta	10^{24}
Z	Zetta	10^{21}
E	Exa	10^{18}
P	Peta	10^{15}
T	Tera	10^{12}
G	Giga	10^{9}
M	Mega	10^{6}
k	Kilo	10^{3}
m	Milli	10^{-3}
μ	Micro	10^{-6}
n	Nano	10^{-9}
p	Pico	10^{-12}
f	Femto	10^{-15}
a	Atto	10^{-18}
z	Zepto	10^{-21}
y	Yocto	10^{-24}
r	Ronto	10^{-27}
q	Quecto	10^{-30}

CHAPTER 2

IEC (Binary) Prefixes

Prefix	Name	Base
Y	Yobi	2^{80}
Z	Zebi	2^{70}
E	Exbi	2^{60}
P	Pedi	2^{50}
T	Tebi	2^{40}
G	Gibi	2^{30}
M	Mebi	2^{20}
K	Kibi	2^{10}

CHAPTER 3

Format Specification

```
format_spec      ::=  [[fill]align][sign][#][0][![!]][width][grouping_option][%[-
↪]margin][.precision][type]
fill             ::=  <any character>
align            ::=  "<" | ">" | "=" | "^"
sign             ::=  "+" | "-" | " "
width            ::=  digit+
grouping_option  ::=  "_" | ","
precision        ::=  digit+
type             ::=  "e" | "E" | "f" | "F" | "g" | "G" | "h" | "H" | "k" | "K" | "m"␣
↪| "M" | "n" | "%"
```

Prefixed-specific fields are defined below. Descriptions of standard fields can be found in the Format Specification Mini-Language documentation.

3.1 Prefixed-specific fields

3.1.1 Flags

Flag	Meaning
`'!'`	Add a single space between number and prefix
`'!!'`	Same as `'!'`, but drop space if there is no prefix

3.1.2 Margin

By default, a prefix will be used when the magnitude of that prefix is reached. For example, `format(Float(999), '.1h')` will result in `'999.0'` and `format(Float(1000), '.1h')` will result in `'1.0k'`.

Margin specifies the percentage to raise or lower these thresholds.

```
>>> f'{Float(950):%-5.2h}'
'0.95k'

>>> f'{Float(1000):%5.2h}'
'1000.00'
```

3.1.3 Presentation Types

Type	Meaning
`'h'`	SI format. Outputs the number with closest divisible SI prefix. (k, M, G, ...)
`'H'`	Same as `'h'` with precision indicating significant digits.
`'k'`	IEC Format. Outputs the number with closest divisible IEC prefix. (Ki, Mi, Gi, ...)
`'K'`	Same as `'k'` with precision indicating significant digits.
`'m'`	Short IEC Format. Same as `'k'` but only a single character. (K, M, G, ...)
`'M'`	Same as `'m'` with precision indicating significant digits.
`'j'`	Alias for `'k'` - DEPRECATED
`'J'`	Alias for `'m'` - DEPRECATED

CHAPTER 4

API Reference

class prefixed.**Float**([*x*])

Subclass of the built-in `float` class

Key differences:

- When a math operation is performed with another real number type (`float`, `int`), the result will be a *`prefixed.Float`* instance.
- Additional presentation types `'h'`, `'H'`, `'k'`, `'K'`, `'m'`, and `'M'` are supported for f-strings and `format()`.

Type	Meaning
`'h'`	SI format. Outputs the number with closest divisible SI prefix. (k, M, G, ...)
`'H'`	Same as `'h'` with precision indicating significant digits.
`'k'`	IEC Format. Outputs the number with closest divisible IEC prefix. (Ki, Mi, Gi, ...)
`'K'`	Same as `'k'` with precision indicating significant digits.
`'m'`	Short IEC Format. Same as `'k'` but only a single character. (K, M, G, ...)
`'M'`	Same as `'m'` with precision indicating significant digits.
`'j'`	Alias for `'k'` - DEPRECATED
`'J'`	Alias for `'m'` - DEPRECATED

- When initializing from strings, SI and IEC prefixes are honored

```
>>> Float('2k')
Float(2000.0)

>>> Float('2Ki')
Float(2048.0)
```

- An additional format flag '!' is available which adds a space before the prefix

```
>>> f'{Float(3250):!.2h}'
'3.25 k'
```

- When the `'H'`, `'K'`, or `'M'` presentation types are used, precision is treated as the number of significant digits to include. Standard rounding will occur for the final digit.

```
>>> f'{Float(1246):.3h}'
'1.246k'

>>> f'{Float(1246):.3H}'
'1.25k'
```

  By default, trailing zeros are removed.

```
>>> f'{Float(1000):.3H}'
'1k'
```

  To preserve trailing zeros, include the `'#'` flag.

```
>>> f'{Float(1000):#.3H}'
'1.00k'
```

- An additional field, margin, can be specified which lowers or raises the threshold for for each prefix by the given percentage. Margin is specified before precision with the syntax `%[-]digit+`.

```
>>> f'{Float(950):.2h}'
'950.00'

>>> f'{Float(950):%-5.2h}'
'0.95k'

>>> f'{Float(1000):%5.2h}'
'1000.00'

>>> f'{Float(1050):%5.2h}'
'1.05k'
```

CHAPTER 5

Overview

Prefixed provides an alternative implementation of the built-in `float` which supports formatted output with SI (decimal) and IEC (binary) prefixes.

```
>>> from prefixed import Float

>>> f'{Float(3250):.2h}'
'3.25k'

>>> '{:.2h}s'.format(Float(.00001534))
'15.34µs'

>>> '{:.2k}B'.format(Float(42467328))
'40.50MiB'

>>> f'{Float(2048):.2m}B'
'2.00KB'
```

Because *`prefixed.Float`* inherits from the built-in `float`, it behaves exactly the same in most cases.

When a math operation is performed with another real number type (`float`, `int`), the result will be a *`prefixed.Float`* instance.

5.1 Presentation Types

Additional presentation types `'h'`, `'H'`, `'k'`, `'K'`, `'m'`, and `'M'` are supported for f-strings and `format()`.

Type	Meaning
'h'	SI format. Outputs the number with closest divisible SI prefix. (k, M, G, . . .)
'H'	Same as 'h' with precision indicating significant digits.
'k'	IEC Format. Outputs the number with closest divisible IEC prefix. (Ki, Mi, Gi, . . .)
'K'	Same as 'k' with precision indicating significant digits.
'm'	Short IEC Format. Same as 'k' but only a single character. (K, M, G, . . .)
'M'	Same as 'm' with precision indicating significant digits.
'j'	Alias for 'k' - DEPRECATED
'J'	Alias for 'm' - DEPRECATED

5.2 String Initialization

When initializing from strings, SI and IEC prefixes are honored

```
>>> Float('2k')
Float(2000.0)

>>> Float('2Ki')
Float(2048.0)
```

5.3 Additional Flags

An additional format flag '!' is available which adds a space before the prefix

```
>>> f'{Float(3250):!.2h}'
'3.25 k'
```

5.4 Significant Digits

When the 'H', 'K, or 'M' presentation types are used, precision is treated as the number of *significant digits* to include. Standard rounding will occur for the final digit.

```
>>> f'{Float(1246):.3h}'
'1.246k'

>>> f'{Float(1246):.3H}'
'1.25k'
```

By default, trailing zeros are removed.

```
>>> f'{Float(1000):.3H}'
'1k'
```

To preserve trailing zeros, include the '#' flag.

```
>>> f'{Float(1000):#.3H}'
'1.00k'
```

5.5 Adjustable Thresholds

An additional field, margin, can be specified which lowers or raises the threshold for for each prefix by the given percentage. Margin is specified before precision with the syntax `%[-]digit+`.

```
>>> f'{Float(950):.2h}'
'950.00'

>>> f'{Float(950):%-5.2h}'
'0.95k'

>>> f'{Float(1000):%5.2h}'
'1000.00'

>>> f'{Float(1050):%5.2h}'
'1.05k'
```

Python Module Index

p

Index

F

P