# Python number formatting examples

Last updated: 04 May 2020

#### **Table of Contents**

- Round float to 2 decimal places
- Format float as percentage
- Truncate float at 2 decimal places
- Left padding with zeros
- Right padding with zeros
- Use commas as thousands separator
- Interpolate variable with f-strings
- ValueError: zero length field name in format



All examples assume Python 3+



See the <u>difference between truncating and rounding a number</u> (<a href="http://www.mathwords.com/t/truncating\_a\_number.htm">http://www.mathwords.com/t/truncating\_a\_number.htm</a>)

All examples can be viewed <u>on this jupyter notebook (https://nbviewer.jupyter.org/github/queirozfcom/python-sandbox/blob/master/python3/notebooks/number-formatting-post/main.ipynb?flush\_cache=true)</u>

### Round float to 2 decimal places

```
print('{:.2f}'.format(5.39120))
# >>> '5.40'
```

# Format float as percentage

Format a number as a percentage, with 2 decimal places

```
'{:.2f}%'.format(10.12345)
# >>> '10.12%'
```





-73%		-42%	-34%



View the full code for a generalized version on this notebook

(https://nbviewer.jupyter.org/github/queirozfcom/python-sandbox/blob/master/python3/notebooks/number-for-matting-post/main.ipynb?flush\_cache=true#truncate-to-at-most-2-decimal-places)



Drop digits after the second decimal place (if there are any).

```
import re
# see the notebook for a generalized version
def truncate(num):
    return re.sub(r'^(\d+\.\d{,2})\d*$',r'\l',str(num))

truncate(8.499)
# >>> '8.49'

truncate(8.49)
# >>> '8.49'

truncate(8.4)
# >>> '8.4'

truncate(8.0)
# >>> '8.0'

truncate(8)
# >>> '8'
```



# Left padding with zeros

**Example** make the full size equal to 9 (all included), filling with zeros to the left:

-1 +1 +-+-7 -+	:: AT ! FACT 0 /:7		£:11: +
-73%	-42%	-34%	

# Right padding with zeros

**Example** make the full size equal to 11, filling with zeros to the right:

```
'{:<011}'.format(3.499)
# >>> '3.499000000'
```

### Use commas as thousands separator

```
"{:,}".format(100000)
# >>> '100,000'
```

# Interpolate variable with f-strings

You can use any of the other formatting options with **f-strings**:

Example: truncate to 2 decimal places in f-string

```
num = 1.12745

formatted = f"{num:.2f}"
formatted
# >>> '1.13'
```

### **OSU Data Analytics Certificate**

Ad 4-course program. \$675 per course with early registration. Classe 10.

The Ohio State University

Open

# ValueError: zero length field name in format

In some Python versions such as 2.6 and 3.0, you must specify positional indexes in the format string:

-73%			-42%		-34%		

```
# ValueError in python 2.6 and 3.0
a=1
b=2
"{}-{}".format(a,b)

# NO ERROR in any python version
"{0}-{1}".format(a,b)
# >>> "1-2"
```

### References

• <u>Python 3 docs: Format Specification Mini Language (https://docs.python.org/3.6/library/string.html#format-specification-mini-language)</u>

Felipe () ■ 05 Nov 2017 ■ 04 May 2020 • python (/tag/python) • formatting (/tag/formatting)

« Linux File Processing: head and tail examples (/entries/linux-file-processing-head-and-tail-examples)

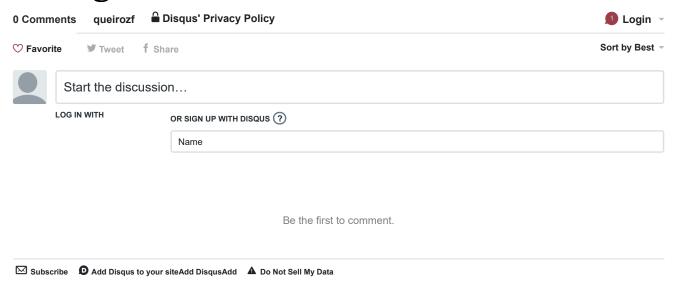
Archive (/archive)

Paper Summary: Multi-instance multi-label learning for automatic tag recommendation » (/entries/paper-summary-multi-instance-multi-label-learning-for-automatic-tag-recommendation)

### Related content

-73%		-42%	-34%	

# Dialogue & Discussion



1

### ABOUT THIS SITE

Technology reference and information archive. More > (/about)

#### **OTHER**

()

Contact (/contact)

Atom Feed (/atom.xml)

sitemap.xml (/sitemap.xml)

### **CREDITS**

()

Theme by Phlow (http://phlow.de/)

Favicon by Webalys (https://www.iconfinder.com/webalys)

Created with JEKYLL (HTTP://JEKYLLRB.COM/).







(http(//ptiplshoftygra/nov/igunleishtirkfsbarr/eim/éte/ippeeipbart/ficeinha)

-73%			-42%		-34%		