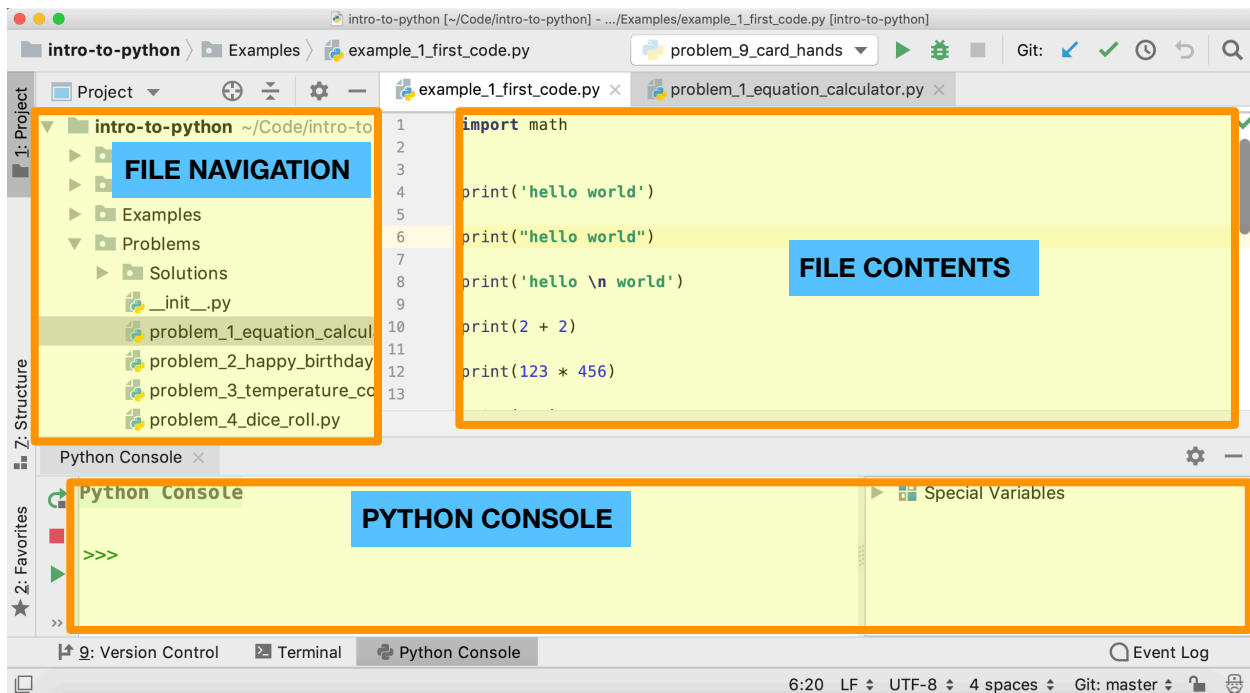# Reference material for Programming with Python: Beyond the Basics
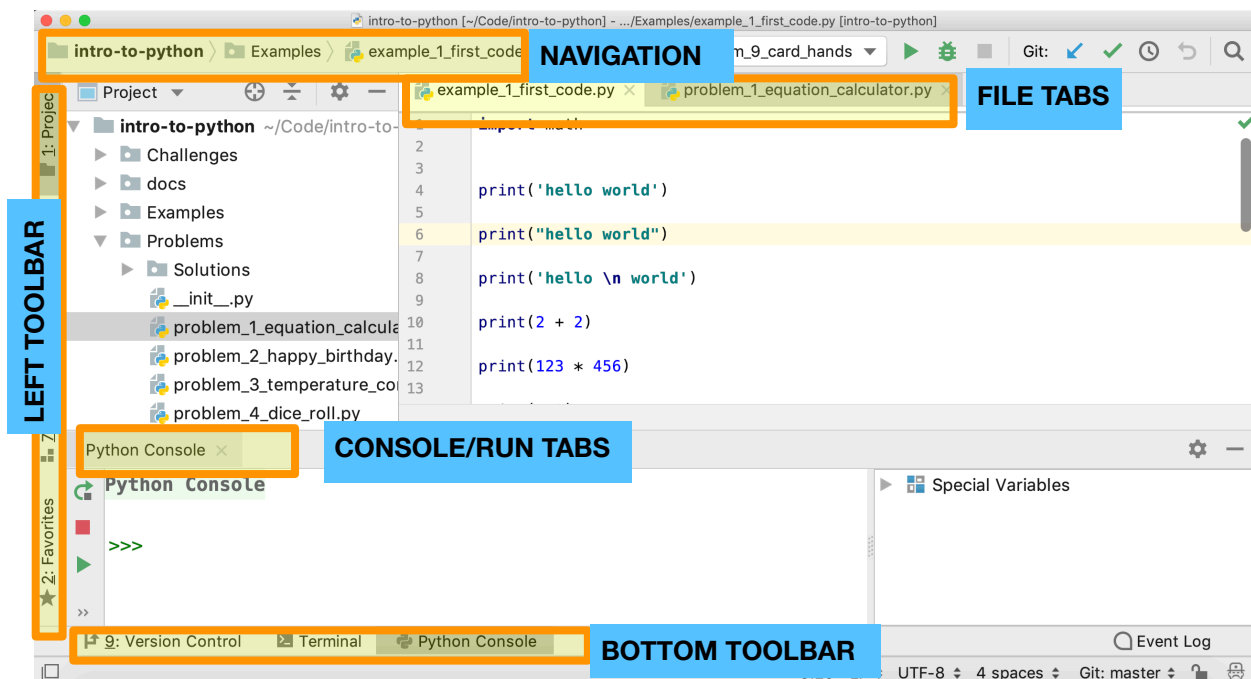
Arianne Dee

PyCharm Layout

Sections

Toolbars

Keyboard Shortcuts

References

https://www.jetbrains.com/help/pycharm/mastering-keyboard-shortcuts.html
Select operating system/key-map at top

https://resources.jetbrains.com/storage/products/pycharm/docs/PyCharm_ReferenceCard.pdf
Some differences between PC/Mac/PyCharm versions

Edit/browse shortcuts

Go to Settings > Keymap

Common shortcuts

A. PyCharm

Ctrl+Shift+A	Search for keyboard shortcut
Ctrl+,	Open settings
Alt+Enter	Show error dialogue

B. Most programs

Ctrl+C	Copy	**Ctrl+V**	Paste
Ctrl+Z	Undo	**Ctrl+Shift+Z**	Redo
Ctrl+F	Find in file	**Ctrl+R**	Replace in file

C. Navigation

Shift+Arrows	Select multiple characters/lines	**Tab+Up**	Widen selection

D. New lines

Ctrl+Back	Delete entire line		
Shift+Enter	Start new line below	**Ctrl+Enter**	Start new line above

E. Used in workshop

Ctrl+/	Comment / uncomment line		
Ctrl+D	Duplicate selection		
Ctrl+Shift+↑	Move line up	**Ctrl+Shift+↓**	Move line down
Ctrl+Alt+L	Reformat file		
Ctrl+B	Go to definition		

F. Advanced

Shift+F6	Rename	**F6**	Move
Ctrl+Shift+F	Find in project	**Ctrl+Shift+R**	Replace in project
Ctrl+Shift+]	Go to next tab	**Ctrl+Shift+[**	Go to previous tab
Ctrl+Shift+U	Toggle case		

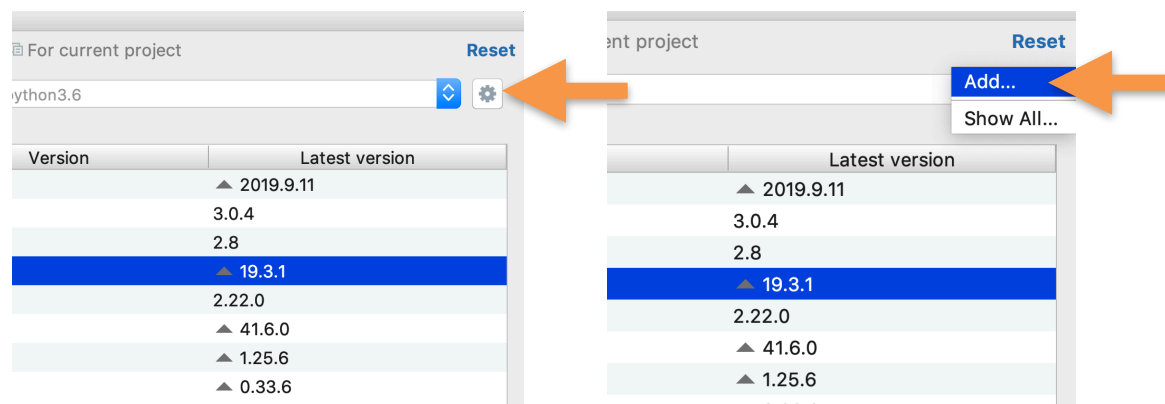# Creating a virtual environment in PyCharm

1. Open settings/preferences

Mac

PC
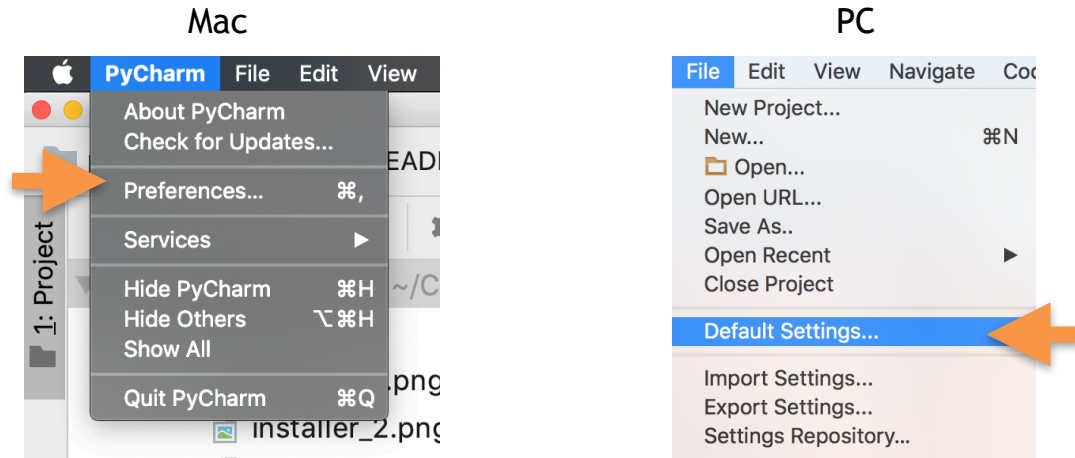
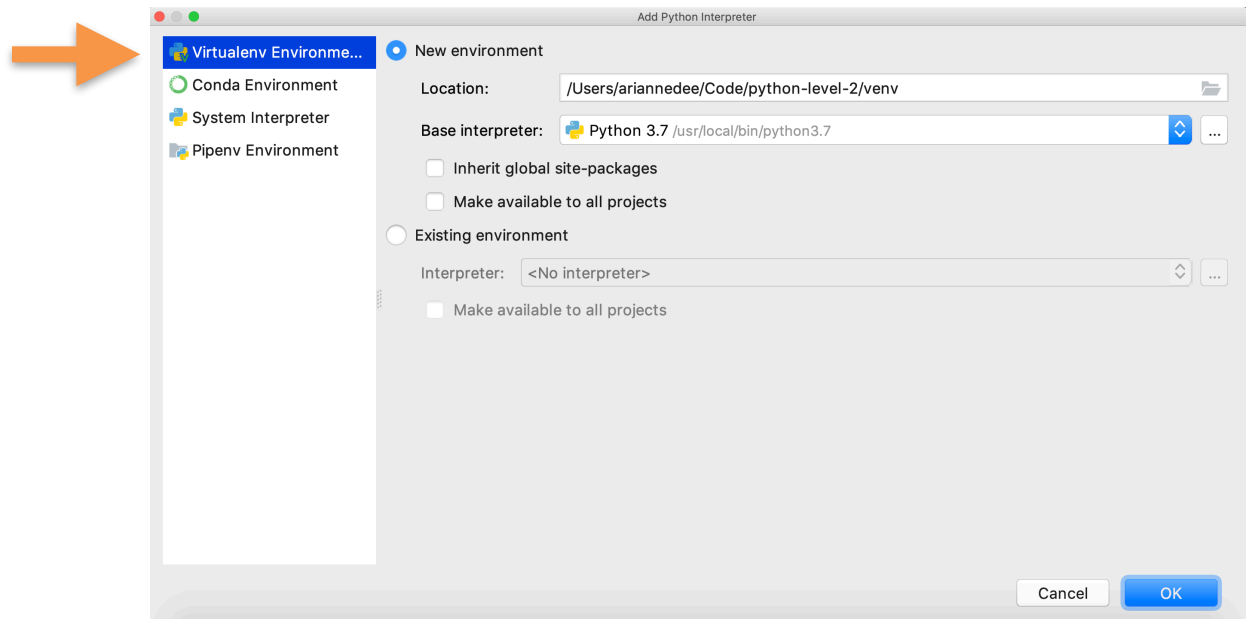## 2. Navigate to Project Interpreter
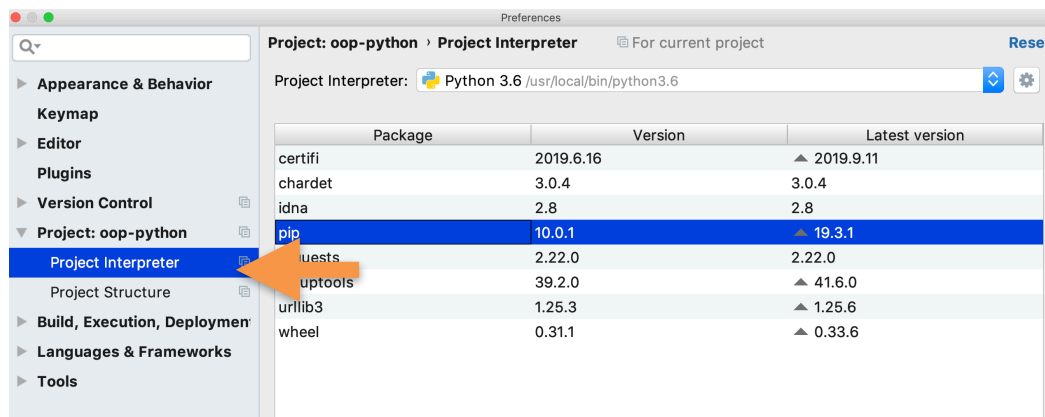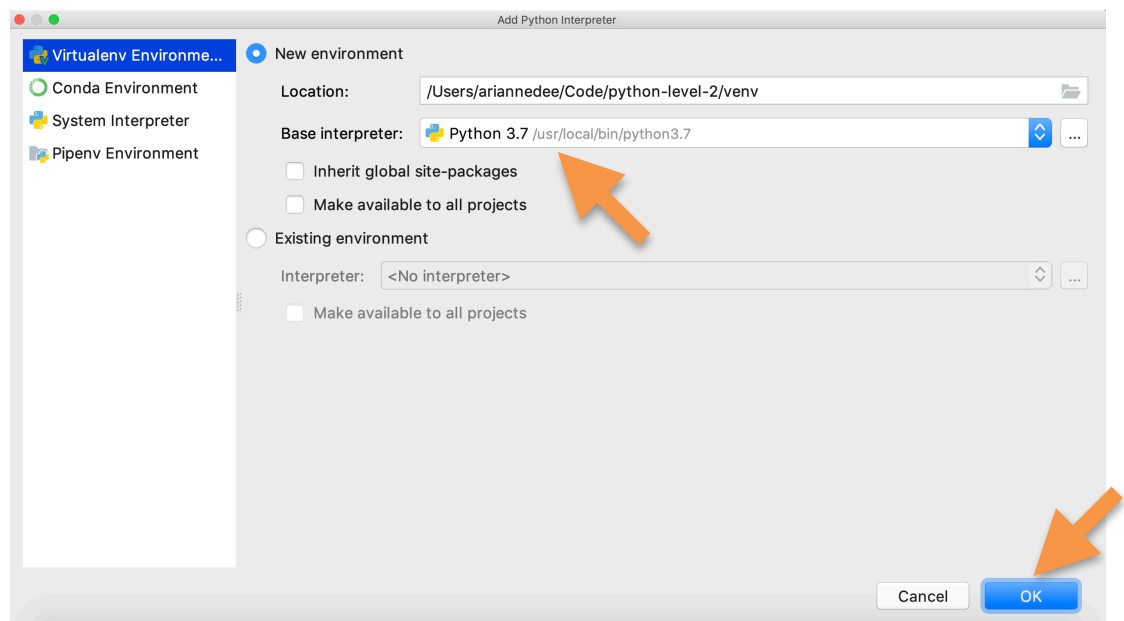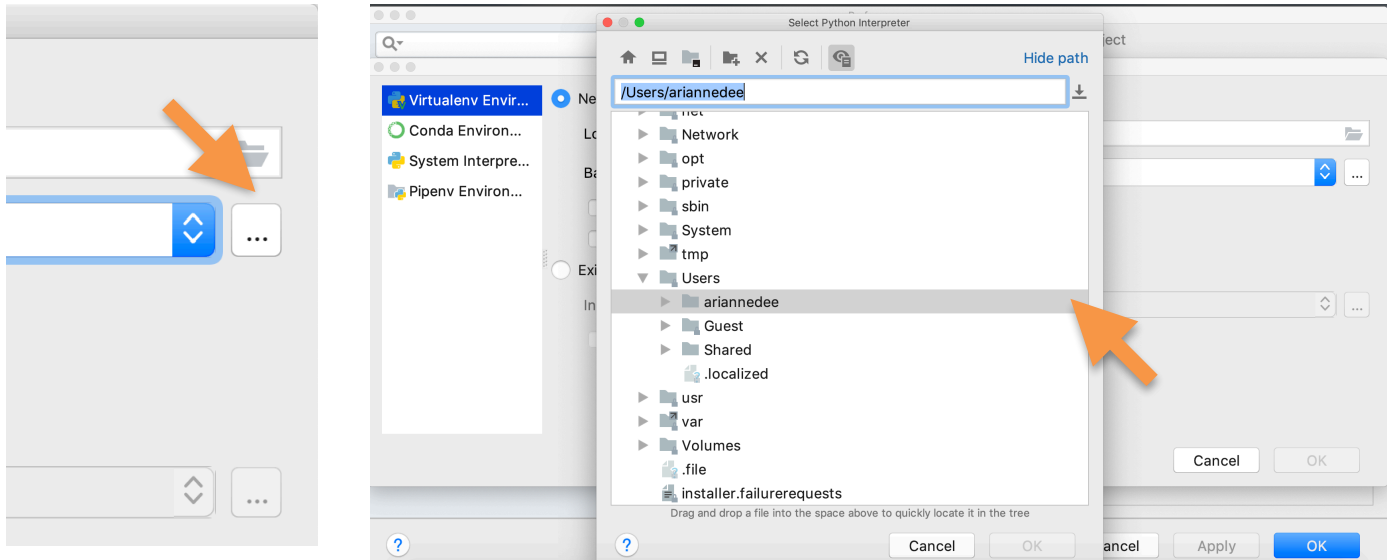

3. Add a new project interpreter

4. Create new virtual environment

5a. Select existing system interpreter

5b. Find system interpreter

If you're not sure where to find it:

Windows (look for python.exe)
- C:\Python37
- C:\Program Files\Python37
- C:\Users\username\AppData\Local\Programs\Python\Python37-XX
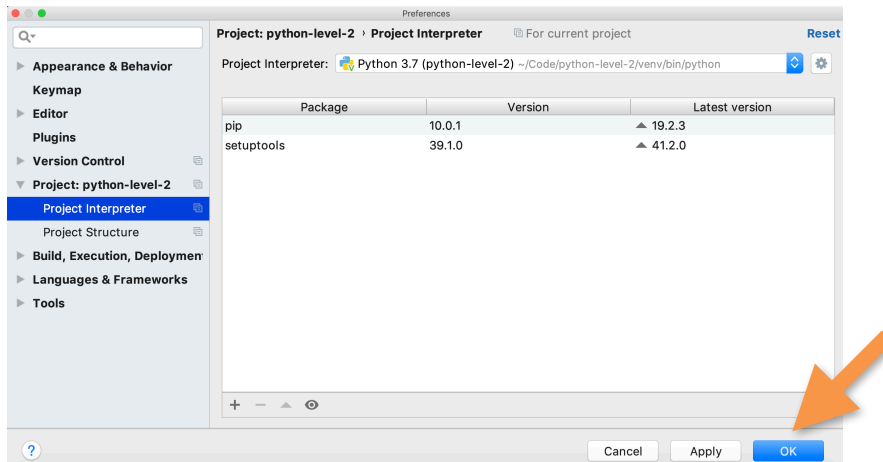
Mac (look for python3.7 or python3)
- /usr/local/bin/
- /Library/Frameworks/Python.framework/Versions/3.7/bin/
- /usr/local/Cellar/python/3.7.X_X/bin/
- /Users/username/anaconda/bin/
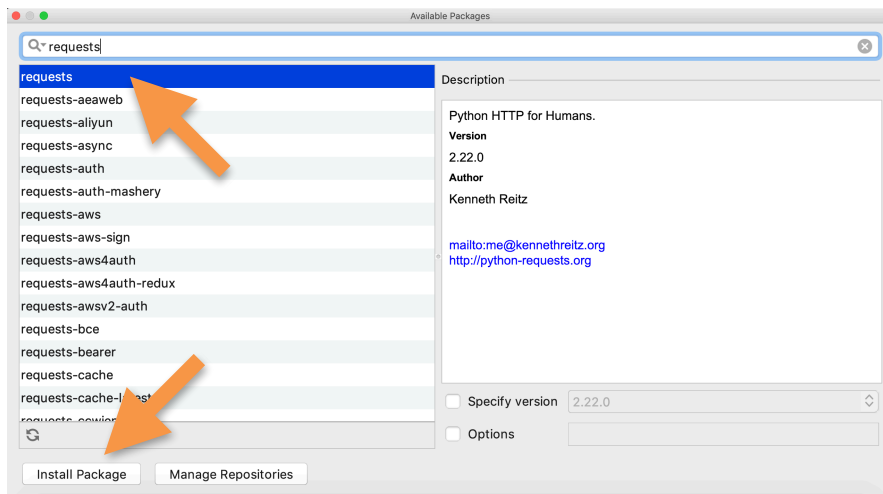- /anaconda3/bin/

Linux (look for python3.7 or python3)
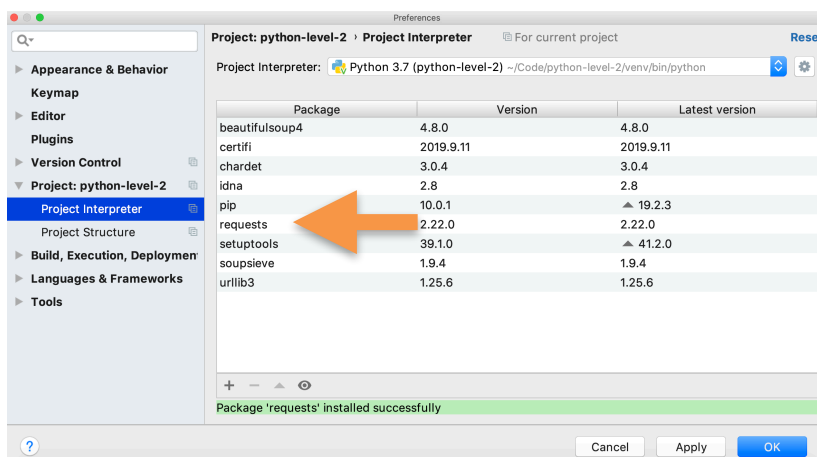- /usr/bin/
- /usr/local/bin

Installing packages

1. Navigate to add package view

2. Search for package, select, and install

3. Verify it worked

Beautiful Soup

Full documentation: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

Finding multiple elements

Find all elements (returns a list)
```
soup.find_all('a')
```

Find all elements with an attribute
```
soup.find_all("a", attrs={"class": "active"})
```

Fin all elements that contain some content
```
soup.find_all("a", string="About")
```

Finding an element

Get a child element
```
soup.a
```

Find the first element with an attribute
```
soup.find("a", attrs={"class": "active"})
```

Find an element that contains some content
```
soup.find("a", string="About")
```

Getting data from an element

Get the text of an element
```
link.string
```

Get an attribute of the element (if you know it exists)
```
link['href']
```

Get an attribute of the element (if it might not exist)
```
link.get('href')
```

Python 2 vs Python 3

If you haven't been able to get Python 3 working, here are some of the differences that may affect you throughout the workshop.

Getting user input
Python 3
```
name = input("What's your name?")
```

Python 2
```
name = raw_input("What's your name?")
```

String formatting
Python 3
```
print(f"hello {name}")
```

Python 2
```
print("hello {}".format(name))
```

Integer vs. float division
Python 3
```
1/2   =>   0.5
1//2  =>   0
```

Python 2
```
1/2   =>   0
1./2  =>   0.5
```