

zsh\_basics

2022 年 10 月 1 日

# 目录

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Terminology . . . . .	4
1.2	commands, aruguments and flags . . . . .	4
<b>2</b>	<b>Examples</b>	<b>5</b>
2.1	open . . . . .	5
2.2	find & which . . . . .	6
2.3	Pandoc . . . . .	7
2.4	nbconvert . . . . .	8
2.4.1	HTML . . . . .	9
2.4.2	PDF . . . . .	9
2.4.3	Python . . . . .	11
2.4.4	Latex . . . . .	11
<b>3</b>	<b>Work with Files and Directories</b>	<b>12</b>
3.1	Create a File . . . . .	12
3.2	Create a Directory . . . . .	12

本节尝试使用在 jupyter notebook 使用 zsh，顺便写一个 zsh 的食用指南，notebook 写真的方便很多啊。

```
[21]: import zsh_in_jupyter
```

```
[ ]:
```

## 1 Introduction

因为及其好用，比如具体而言

- 许多功能在图形界面不提供，只有通过命令行来实现。
- Finder 会隐藏许多你不太会需要的文件，而 Command line 则可以访问所有文件。
- Command line 可以通过 SSH 远程访问你的 Mac 或者 Linux。
- 普通用户可以通过 ‘sudo’ 命令获得 root 用户权限。
- 如果你开启手动输入用户名登陆模式，登陆时在用户名处输入 ‘>console’ 可以直接进入命令行界面。随后你仍然需要登录到一个账户。

```
[31]: !echo ~
```

```
/Users/nymath
```

```
[45]: get_ipython().system('echo ~')
```

```
/Users/nymath
```

```
[3]: # 打开当前文件夹
!temp=$(pwd);\
open $temp
```

```
[93]: def zsh(par='temp=$(pwd);open $temp'):
        if type(par) == str:
            return get_ipython().system(par)
        else:
            return 'type error'
```

```
[55]: func.system('temp=$(pwd);open $temp')
```

```
[52]: get_ipython().system('temp=$(pwd);open $temp')
```

```
[ ]: from zsh_in_jupyter import
```

## 1.1 Terminology

- **Command line:** A command-line interface is a way of giving instructions to a computer and getting results back
- **Shell:** A shell is a program that creates a user interface of one kind or another enabling you to interact with a computer
  - **csh:** the C shell, named for similarities to the C programming language
  - **zsh:** the Z shell, an advanced shell named after Yale professor Zhong Shao that incorporates features from tcsh, ksh, and bash, plus other capabilities
  - **dash:** the Debian Almquist shell, a lightweight shell that's been around for more than two decades, but was not included with macOS until Catalina
- **Terminal:** the devices people used to interact with computers back in the days of monolithic mainframes.

## 1.2 commands, arguments and flags

- **Command:** Commands are straightforward; they're the verbs of the command line (even though they may look nothing like English verbs).

```
pwd
date
```

- **Arguments:** Along with commands (verbs), we have arguments, which you can think of as nouns

比如你要处理一个文件，需要指定这个文件的名字

```
nano file1
```

- **Flags:** Besides verbs and nouns, we have adverbs! In English, I could say, “Eat cereal quickly!” or “Watch TV quietly.”

ls 展示文件，ls a 展示所有文件

```
[2]: !pwd; \N
date
```

```
/Users/nymath/Desktop/python/python_quant/Statistical_learning
```

```
Fri Sep 30 00:41:43 CST 2022
```

```
[26]: !ls -l -a
```

```
total 88
drwxrwxr-x@ 7 nymath  staff    224 Sep 30 00:49 .
drwxr-xr-x@ 16 nymath  staff    512 Sep 30 00:35 ..
drwxr-xr-x  4 nymath  staff    128 Sep 29 21:14
.ipynb_checkpoints
-rw-r--r--   1 nymath  staff  22692 Sep 30 00:22 Structures of Data.ipynb
-rw-r--r--@  1 nymath  staff    158 Sep 30 00:22 fib.py
-rw-----   1 nymath  staff    170 Sep 30 00:12 fib.py.save
-rw-r--r--   1 nymath  staff  11053 Sep 30 00:49 zsh_basics.ipynb
```

```
[5]: !ls -a; #
```

```
.                Structures of Data.ipynb
zsh_basics.ipynb
..               fib.py
.ipynb_checkpoints  fib.py.save
```

```
[12]: !ls -a f* # show you only the names of files beginning with the letter f
```

```
fib.py      fib.py.save
```

```
[ ]:
```

## 2 Examples

### 2.1 open

open 命令用于打开一个文件夹或者具体的文件，如果指定的是 applications 中的.app 文件，则可以直接打开。

```
[15]: !open /Applications/Notability.app
```

```
[16]: !open /Applications/Safari.app
```

```
[17]: !open /Applications/WeChat.app
```

```
[18]: !temp=$(pwd);\
open $temp
```

## 2.2 find & which

We can type the command “find” to find the files whose name contains the name given by input. To be specific, if we want to find site-packages in /opt/, we can

```
[5]: !find /opt/anaconda3/lib/python3.9 -name "*site-packages"
```

```
/opt/anaconda3/lib/python3.9/site-packages
```

```
[29]: !find /opt/anaconda3 -name "nbconvert"
```

```
/opt/anaconda3/pkgs/jupyter_server-1.13.5-pyhd3eb1b0_0/site-  
packages/jupyter_server/tests/nbconvert  
/opt/anaconda3/pkgs/jupyter_server-1.13.5-pyhd3eb1b0_0/site-  
packages/jupyter_server/tests/services/nbconvert  
/opt/anaconda3/pkgs/jupyter_server-1.13.5-pyhd3eb1b0_0/site-  
packages/jupyter_server/nbconvert  
/opt/anaconda3/pkgs/jupyter_server-1.13.5-pyhd3eb1b0_0/site-  
packages/jupyter_server/services/nbconvert  
/opt/anaconda3/pkgs/notebook-6.4.8-py39hecd8cb5_0/lib/python3.9/site-  
packages/notebook/nbconvert  
/opt/anaconda3/pkgs/notebook-6.4.8-py39hecd8cb5_0/lib/python3.9/site-  
packages/notebook/services/nbconvert  
/opt/anaconda3/pkgs/nbconvert-6.4.4-py39hecd8cb5_0/info/test/nbconvert  
/opt/anaconda3/pkgs/nbconvert-6.4.4-py39hecd8cb5_0/lib/python3.9/site-  
packages/nbconvert  
/opt/anaconda3/pkgs/nbconvert-6.4.4-py39hecd8cb5_0/share/jupyter/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/jupyter_server/tests/nbconvert  
/opt/anaconda3/lib/python3.9/site-  
packages/jupyter_server/tests/services/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/jupyter_server/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/jupyter_server/services/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/notebook/nbconvert  
/opt/anaconda3/lib/python3.9/site-packages/notebook/services/nbconvert  
/opt/anaconda3/share/jupyter/nbconvert
```

```
[33]: !find /Users/ -name "pandoc"
```

```
/Users//nymath/Library/TeXShop/Engines/Inactive/pandoc
```

```
[14]: !which jupyter
```

```
/opt/anaconda3/bin/jupyter
```

```
[13]: !which python # python 解释器所在位置，的确是 vscode 中用到的那个。
```

```
/opt/anaconda3/bin/python
```

```
[ ]:
```

## 2.3 Pandoc

```
[34]: # 获取 pandoc 的安装路径
```

```
!find /Users -name "pandoc"
```

```
/Users/nymath/Library/TeXShop/Engines/Inactive/pandoc
```

```
[56]: !open /Users/nymath/Library/TeXShop/Engines/Inactive/pandoc
```

```
[37]: # pandoc 这个解释器?或者说函数? 命令? 保存的位置
```

```
!which pandoc
```

```
/usr/local/bin/pandoc
```

开始 pandoc 的应用介绍了

```
[49]: !pandoc --version
```

```
pandoc 2.19.2
```

```
Compiled with pandoc-types 1.22.2.1, texmath 0.12.5.2, skylighting 0.13,  
citeproc 0.8.0.1, ipynb 0.2, hslua 2.2.1
```

```
Scripting engine: Lua 5.4
```

```
User data directory: /Users/nymath/.local/share/pandoc
```

```
Copyright (C) 2006-2022 John MacFarlane. Web: https://pandoc.org
```

```
This is free software; see the source for copying conditions. There is no  
warranty, not even for merchantability or fitness for a particular purpose.
```

```
[40]: !mkdir pandoc-test
```

```
[45]: !cd pandoc-test;\n      pwd;\n
```

/Users/nymath/Desktop/python/python\_quant/Statistical\_learning/pandoc-test

相关命令解读

```
pandoc test1.md -f markdown -t html -s -o test1.html
```

*test1.md* tells pandoc which file to convert.

-f markdown -t html: from md to html.

-s says to create a “standalone” file

-o test1.html says to put the output in the file test1.html.

```
pandoc -f html -t markdown hello.html
```

将 hello.html 转化为 hello.md

```
[57]: !pandoc zsh_basics.ipynb -f ipynb -t latex -s -o zsh_basics.tex
```

关于如何添加模版，可以参考 [Template](#)

```
[ ]:
```

```
[ ]:
```

```
[54]: zsh_in_jupyter.opendir()
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

## 2.4 nbconvert

[指南](#)

当然 pandoc 也非常重要，可以看看

[Pandoc](#)



### 2.4.1 HTML

```
[36]: # HTML
      !jupyter nbconvert --to html zsh_basics.ipynb
```

```
[NbConvertApp] Converting notebook zsh_basics.ipynb to html
[NbConvertApp] Writing 595565 bytes to zsh_basics.html
```

### 2.4.2 PDF

PDF 转化为 pdf 倒是挺费功夫的，不信的话直接转化看看，应该会报错，比如不支持中文，因此我们需要去弄一个 template

这个文件所在路径为/opt/anaconda3/share/jupyter/nbconvert/templates/latex

```
[65]: !open /opt/anaconda3/share/jupyter/nbconvert/templates/latex
```

```
[79]: !jupyter nbconvert --to pdf zsh_basics.ipynb
```

```
[NbConvertApp] Converting notebook zsh_basics.ipynb to pdf
[NbConvertApp] Writing 119076 bytes to zsh_basics.pdf
```

```
[72]: !which jupyter
```

```
/opt/anaconda3/bin/jupyter
```

```
[68]: !jupyter nbconvert --to pdf --template dbzh_chapter.tplx zsh_basics.ipynb
```

```
[NbConvertApp] Converting notebook zsh_basics.ipynb to pdf
Traceback (most recent call last):
  File "/opt/anaconda3/bin/jupyter-nbconvert", line 11, in <module>
    sys.exit(main())
  File "/opt/anaconda3/lib/python3.9/site-packages/jupyter_core/application.py",
line 264, in launch_instance
    return super(JupyterApp, cls).launch_instance(argv=argv, **kwargs)
  File "/opt/anaconda3/lib/python3.9/site-
packages/traitlets/config/application.py", line 846, in launch_instance
    app.start()
  File "/opt/anaconda3/lib/python3.9/site-packages/nbconvert/nbconvertapp.py",
line 369, in start
    self.convert_notebooks()
  File "/opt/anaconda3/lib/python3.9/site-packages/nbconvert/nbconvertapp.py",
```

```

line 541, in convert_notebooks
    self.convert_single_notebook(notebook_filename)
File "/opt/anaconda3/lib/python3.9/site-packages/nbconvert/nbconvertapp.py",
line 506, in convert_single_notebook
    output, resources = self.export_single_notebook(notebook_filename,
resources, input_buffer=input_buffer)
File "/opt/anaconda3/lib/python3.9/site-packages/nbconvert/nbconvertapp.py",
line 435, in export_single_notebook
    output, resources = self.exporter.from_filename(notebook_filename,
resources=resources)
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/exporter.py", line 190, in from_filename
    return self.from_file(f, resources=resources, **kw)
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/exporter.py", line 208, in from_file
    return self.from_notebook_node(nbformat.read(file_stream, as_version=4),
resources=resources, **kw)
File "/opt/anaconda3/lib/python3.9/site-packages/nbconvert/exporters/pdf.py",
line 168, in from_notebook_node
    latex, resources = super().from_notebook_node(
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/latex.py", line 72, in from_notebook_node
    return super().from_notebook_node(nb, resources, **kw)
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/templateexporter.py", line 392, in
from_notebook_node
    output = self.template.render(nb=nb_copy, resources=resources)
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/templateexporter.py", line 151, in template
    self._template_cached = self._load_template()
File "/opt/anaconda3/lib/python3.9/site-
packages/nbconvert/exporters/templateexporter.py", line 362, in _load_template
    return self.environment.get_template(template_file)
File "/opt/anaconda3/lib/python3.9/site-packages/jinja2/environment.py", line
883, in get_template
    return self._load_template(name, self.make_globals(globals))
File "/opt/anaconda3/lib/python3.9/site-packages/jinja2/environment.py", line

```

```
857, in _load_template
    template = self.loader.load(self, name, globals)
File "/opt/anaconda3/lib/python3.9/site-packages/jinja2/loaders.py", line 429,
in load
    raise TemplateNotFound(name)
jinja2.exceptions.TemplateNotFound: index.tex.j2
```

```
[74]: zsh_in_jupyter.opendir()
```

### 2.4.3 Python

```
[26]: !jupyter nbconvert --to python zsh_basics.ipynb
```

```
[NbConvertApp] Converting notebook zsh_basics.ipynb to python
[NbConvertApp] Writing 4870 bytes to zsh_basics.py
```

```
[98]: zsh()
```

### 2.4.4 Latex

```
[96]: !jupyter nbconvert --to latex zsh_basics.ipynb
```

```
[NbConvertApp] Converting notebook zsh_basics.ipynb to latex
[NbConvertApp] Writing 31753 bytes to zsh_basics.tex
```

```
修改 documentclass[11pt]{article}
为
documentclass{article}
usepackage{ctex}
```

```
[1]: import zsh_in_jupyter
```

```
[5]: # 查看模组的说明
print(zsh_in_jupyter.__doc__)
```

用于调用 xxx

```
[6]: zsh_in_jupyter.opendir()
```

```
[ ]: !open
[ ]:
[ ]:
```

## 3 Work with Files and Directories

似乎用的不太多，但还是写一点

### 3.1 Create a File

Command **touch** is used for creating a file. (md macos 简直了，右键创建文件夹都得付费)

```
[19]: !touch test.sh
[24]: zsh_in_jupyter.opendir()
```

### 3.2 Create a Directory

```
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
[ ]:
```