

# A Collection of Python Examples

Fan Wang

2020-05-09



# Contents

<b>Preface</b>	<b>5</b>
<b>1 Array, Matrix, Dataframe</b>	<b>7</b>
1.1 Array . . . . .	7
<b>2 System and Support</b>	<b>9</b>
2.1 File In and Out . . . . .	9
<b>A Index and Code Links</b>	<b>15</b>
A.1 Array, Matrix, Dataframe links . . . . .	15
A.2 System and Support links . . . . .	15



# Preface

This is a work-in-progress [website](#) consisting of python tutorials and examples to accomplish. Files are written with [RMD](#) (Allaire et al., 2020). Materials are gathered from various [projects](#) in which python code is used for research and paper-administrative tasks. Files are from [Fan](#)'s [pyfan](#) repository which has an associated [package](#). The package functionalize various tasks tested out in the Rmd files. In addition, the [pyecon](#) repository and the associated [package](#) contain functions and rmd files related explicitly to solving economic models. Bullet points show which python packages/functions are used to achieve various objectives.

From [Fan](#)'s other repositories: For dynamic borrowing and savings problems, see [Dynamic Asset Repository \(Matlab\)](#); For code examples, see also [Matlab Example Code](#), [R Example Code](#), and [Stata Example Code](#); For intro econ with Matlab, see [Intro Mathematics for Economists](#), and for intro stat with R, see [Intro Statistics for Undergraduates](#). See [here](#) for all of [Fan](#)'s public repositories.

The site is built using [Bookdown](#) (Xie, 2020).

Please contact [FanWangEcon](#) for issues or problems.



# Chapter 1

## Array, Matrix, Dataframe

### 1.1 Array

#### 1.1.1 Strings

Go to the [RMD](#), [PDF](#), or [HTML](#) version of this file. Go back to [fan's Python Code Examples Repository](#) ([bookdown site](#)).

##### 1.1.1.1 Replace a Set of Strings in String

Replace terms in string

```
# define string
st_full = """
abc is a great efg, probably xyz. Yes, xyz is great, like efg.
eft good, EFG capitalized, efg good again.
A B C or abc or ABC. Interesting xyz.
"""
```

```
# define new and old
ls_st_old = ['abc', 'efg', 'xyz']
ls_st_new = ['123', '456', '789']

# zip and loop and replace
for old, new in zip(ls_st_old, ls_st_new):
    st_full = st_full.replace(old, new)
```

```
# print
print(st_full)
```

```
##
## 123 is a great 456, probably 789. Yes, 789 is great, like 456.
## eft good, EFG capitalized, 456 good again.
## A B C or 123 or ABC. Interesting 789.
```





## Chapter 2

# System and Support

### 2.1 File In and Out

#### 2.1.1 Reading, Editing, Compile and Convert Tex with Pandoc

Go to the [RMD](#), [PDF](#), or [HTML](#) version of this file. Go back to [fan's Python Code Examples Repository](#) ([bookdown site](#)).

- python create a text file
- python write file from paragraphs

##### 2.1.1.1 Generate a tex file

Will a bare-bone tex file with some texts inside, save inside the `*_file*` subfolder.

First, create the text text string, note the the linebreaks utomatically generate linebreaks, note that slash need double slash:

```
# Create the Tex Text
# Note that trible quotes begin first and end last lines
stf_tex_contents = """\\documentclass[12pt,english]{article}
\\usepackage[bottom]{footmisc}
\\usepackage[urlcolor=blue]{hyperref}
\\begin{document}
\\title{A Latex Testing File}
\\author{\\href{http://fanwangecon.github.io/}{Fan Wang} \\thanks{See information \\href{https://fan
\\maketitle
Ipsum information dolor sit amet, consectetur adipiscing elit. Integer Latex placerat nunc orci.
\\paragraph{\\href{https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3140132}{Data}}
Village closure information is taken from a village head survey.\\footnote{Generally students went t
\\end{document}"""
# Print
print(stf_tex_contents)
```

```
## \documentclass[12pt,english]{article}
## \usepackage[bottom]{footmisc}
## \usepackage[urlcolor=blue]{hyperref}
## \begin{document}
## \title{A Latex Testing File}
## \author{\href{http://fanwangecon.github.io/}{Fan Wang} \thanks{See information \href{https://fan
## \maketitle
## Ipsum information dolor sit amet, consectetur adipiscing elit. Integer Latex placerat nunc orci.
## \paragraph{\href{https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3140132}{Data}}
## Village closure information is taken from a village head survey.\footnote{Generally students went
## \end{document}
```

Second, write the contents of the file to a new tex file stored inside the `*_file*` subfolder of the directory:

```
# Relative file name
srt_file_tex = "_file/"
sna_file_tex = "test_fan"
srn_file_tex = srt_file_tex + sna_file_tex + ".tex"
# Open new file
fl_tex_contents = open(srn_file_tex, 'w')
# Write to File
fl_tex_contents.write(stf_tex_contents)
# print
```

```
## 617
```

```
fl_tex_contents.close()
```

### 2.1.1.2 Replace Strings in a tex file

Replace a set of strings in the file just generated by a set of alternative strings.

```
# Open file Get text
fl_tex_contents = open(srn_file_tex)
stf_tex_contents = fl_tex_contents.read()
print(srn_file_tex)
```

```
# define new and old
```

```
## _file/test_fan.tex
```

```
ls_st_old = ['information', 'Latex']
ls_st_new = ['INFOREPLACE', 'LATEX']
```

```
# zip and loop and replace
```

```
for old, new in zip(ls_st_old, ls_st_new):
    stf_tex_contents = stf_tex_contents.replace(old, new)
print(stf_tex_contents)
```

```
# write to file with replacements
```

```
## \documentclass[12pt,english]{article}
## \usepackage[bottom]{footmisc}
## \usepackage[urlcolor=blue]{hyperref}
## \begin{document}
## \title{A LATEX Testing File}
## \author{\href{http://fanwangecon.github.io/}{Fan Wang} \thanks{See INFOREPLACE \href{https://fanwangecon.github.io/}{https://fanwangecon.github.io/}}}
## \maketitle
## Ipsum INFOREPLACE dolor sit amet, consectetur adipiscing elit. Integer LATEX placerat nunc orci.
## \paragraph{\href{https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3140132}{Data}}
## Village closure INFOREPLACE is taken from a village head survey.\footnote{Generally students went to the village head survey}
## \end{document}
```

```
sna_file_edited_tex = "test_fan_edited"
srn_file_edited_tex = srt_file_tex + sna_file_edited_tex + ".tex"
fl_tex_ed_contents = open(srn_file_edited_tex, 'w')
fl_tex_ed_contents.write(stf_tex_contents)
```

```
## 617
```

```
fl_tex_ed_contents.close()
```

### 2.1.1.3 Convert Tex File to Pandoc and Compile Latex

Compile tex file to pdf and clean up the extraneous pdf outputs.

```

import subprocess
import os

# Convert tex to pdf
subprocess.call(['C:/texlive/2019/bin/win32/xelatex.exe', '-output-directory',
                srt_file_tex, srn_file_edited_tex], shell=False)
# Clean pdf extraneous output

## 0
ls_st_remove_suffix = ['aux', 'log', 'out', 'bbl', 'blg']
for st_suffix in ls_st_remove_suffix:
    srn_cur_file = srt_file_tex + sna_file_edited_tex + "." + st_suffix
    if (os.path.isfile(srn_cur_file)):
        os.remove(srt_file_tex + sna_file_edited_tex + "." + st_suffix)

```

Use pandoc to convert tex file

```

import subprocess

# md file name
srn_file_md = srt_file_tex + "test_fan_edited.md"
# Convert tex to md
subprocess.call(['pandoc', '-s', srt_file_tex, '-o', srn_file_md])
# Open md file

## 0
fl_md_contents = open(srn_file_md)
print(fl_md_contents.read())

## ---
## author:
## - '[Fan Wang](http://fanwangecon.github.io/) [^1]'
## title: A Latex Testing File
## ---
##
## \maketitle
## Ipsum information dolor sit amet, consectetur adipiscing elit. Integer
## Latex placerat nunc orci.
##
## ##### [Data](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3140132)
##
## Village closure information is taken from a village head survey.[^2]
##
## [^1]: See information
##      [Tex4Econ](https://fanwangecon.github.io/Tex4Econ/) for more.
##
## [^2]: Generally students went to schools.

```

#### 2.1.1.4 Search for Files with Suffix in Several Folders

- python search all files in folders with suffix

Search for files in several directories that have a particular suffix. Then decompose directory into sub-components.

```

from pathlib import Path

# directories to search in
ls_spt_srh = ["C:/Users/fan/R4Econ/amto/array/",

```

```

"C:/Users/fan/R4Econ/amto/list"]

# get file names
ls_spn_found = [spn_file for spt_srh in ls_spt_srh
                 for spn_file in Path(spt_srh).rglob('*.R')]
for spn_found in ls_spn_found:
    drive, path_and_file = os.path.splitdrive(spn_found)
    path_no_suffix = os.path.splitext(spn_found)[0]
    path_no_file, file = os.path.split(spn_found)
    file_no_suffix = Path(spn_found).stem
    print('file:', file, '\ndrive:', drive,
          '\nfile no suffix:', file_no_suffix,
          '\nfull path:', spn_found,
          '\npt no file:', path_no_file,
          '\npt no suf:', path_no_suffix, '\n')

## file: fs_ary_basics.R
## drive: C:
## file no suffix: fs_ary_basics
## full path: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_basics.R
## pt no file: C:\Users\fan\R4Econ\amto\array\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_basics
##
## file: fs_ary_generate.R
## drive: C:
## file no suffix: fs_ary_generate
## full path: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_generate.R
## pt no file: C:\Users\fan\R4Econ\amto\array\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_generate
##
## file: fs_ary_mesh.R
## drive: C:
## file no suffix: fs_ary_mesh
## full path: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_mesh.R
## pt no file: C:\Users\fan\R4Econ\amto\array\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_mesh
##
## file: fs_ary_string.R
## drive: C:
## file no suffix: fs_ary_string
## full path: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_string.R
## pt no file: C:\Users\fan\R4Econ\amto\array\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_ary_string
##
## file: fs_meshr.R
## drive: C:
## file no suffix: fs_meshr
## full path: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_meshr.R
## pt no file: C:\Users\fan\R4Econ\amto\array\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\array\htmlpdf\fs_meshr
##
## file: fs_listr.R
## drive: C:
## file no suffix: fs_listr
## full path: C:\Users\fan\R4Econ\amto\list\htmlpdf\fs_listr.R
## pt no file: C:\Users\fan\R4Econ\amto\list\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\list\htmlpdf\fs_listr
##

```

```
## file: fs_lst_basics.R
## drive: C:
## file no suffix: fs_lst_basics
## full path: C:\Users\fan\R4Econ\amto\list\htmlpdf\fs_lst_basics.R
## pt no file: C:\Users\fan\R4Econ\amto\list\htmlpdf
## pt no suf: C:\Users\fan\R4Econ\amto\list\htmlpdf\fs_lst_basics
```



# Appendix A

## Index and Code Links

### A.1 Array, Matrix, Dataframe links

#### A.1.1 **Section 1.1 Array** links

1. [Python String Manipulation Examples: rmd](#) | [r](#) | [pdf](#) | [html](#)
  - Various string manipulations
  - **py**: `zip()`

### A.2 System and Support links

#### A.2.1 **Section 2.1 File In and Out** links

1. [Python Reading and Writing to File Examples: rmd](#) | [r](#) | [pdf](#) | [html](#)
  - Reading from file and replace strings in file.
  - **py**: `open()` + `write()` + `replace()` + `[c for b in [[1,2],[2,3]] for c in b]`
  - **subprocess**: `read()`
  - **pathlib**: `Path().rglob()` + `Path().stem`
  - **os**: `remove()` + `listdir()` + `path.isfile()` + `path.splitdrive()` + `os.path.splitext()` + `os.path.split()`





# Bibliography

Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J., Chang, W., and Iannone, R. (2020). *rmarkdown: Dynamic Documents for R*. R package version 2.1.

Xie, Y. (2020). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.18.