JupyterTutorial_JL

October 11, 2020

0.0.1 Intro to Jupyter Notebooks and Quick Python Warmup

0.0.2 University of California, Santa Barbara

0.0.3 PSTAT 135/235: Big Data Analytics

0.0.4 Last Updated: May 30, 2020

Welcome to this short assignment where you will demonstrate basic Jupyter notebook knowledge and do a quick Python warmup! Total points: 8

1) (1 PT) First, rename this notebook to JupyterTutorial_[your_initials], where you will enter your initials in place of [your_initials].

Done! It is now called "JupyterTutorial_JL.ipynb"(JL = Justin Lee).

2) (1 PT) In the cell below, enter a list of data science topics you find interesting. Use the markdown style (you will need to change the style from the Code style).

- Machine Learning
- Data Mining
- Simulations

3) (1 PT) In the cell below, enter the following Python list:

some_vals = [1, 6, 10, 55]

You will use the Code style, and run the cell.

[1]:
```
some_vals = [1, 6, 10, 55]
```

4) (1 PT) Next, use a list comprehension to return a filtered list containing only the values greater than 6.
Call this list *some_vals_filtered* and print it. You can chain multiple commands on a single line like this:

[2]:
```
some_vals_filtered = list(filter(lambda x: (x > 6), some_vals))
print(some_vals_filtered)
```

```
[10, 55]
```

Next, a small pandas dataframe is constructed.

[14]:
```
import pandas as pd

df = pd.DataFrame({'first_name': ['Andy','Crystal'],
                   'domain_facebook' : [1,1],
                   'domain_foursquare' : [0,0],
                   'age' : [20, 32]})
df
```

[14]:

	first_name	domain_facebook	domain_foursquare	age
0	Andy	1	0	20
1	Crystal	1	0	32

5) (1 PT) In the cell below, write a list comprehension that returns the fields names in the dataframe `df` containing the string *domain*. Run the cell to verify the correct result.

[15]:
```
for col in df.columns:
    if "domain" in col:
        print(col)
```

```
domain_facebook
domain_foursquare
```

6) (1 PT) Use the list comprehension from (5) to index into `df` and show the data for columns containing *domain*

[18]:
```
for col in df.columns:
    if "domain" in col:
        print(df[col])
```

```
0    1
1    1
Name: domain_facebook, dtype: int64
0    0
1    0
Name: domain_foursquare, dtype: int64
```

7) (1 PT) In the cell below, print the *domain_facebook* column

[23]:
```
print(df['domain_facebook'])
```

```
0    1
1    1
Name: domain_facebook, dtype: int64
```

8) (1 PT) In the cell below, print the row with index 1.

[28]:
```
print(df.loc[[1]])
```

```
  first_name  domain_facebook  domain_foursquare  age
1    Crystal                1                  0   32
```

[29]:
```
# Save notebook as PDF document
!jupyter nbconvert --to pdf `pwd`/*.ipynb
```

```
[NbConvertApp] Converting notebook
/home/jovyan/assignments/M1_6/JupyterTutorial_JL.ipynb to pdf
[NbConvertApp] Writing 29710 bytes to ./notebook.tex
[NbConvertApp] Building PDF
[NbConvertApp] Running xelatex 3 times: ['xelatex', './notebook.tex', '-quiet']
[NbConvertApp] Running bibtex 1 time: ['bibtex', './notebook']
[NbConvertApp] WARNING | bibtex had problems, most likely because there were no
citations
[NbConvertApp] PDF successfully created
[NbConvertApp] Writing 42337 bytes to
/home/jovyan/assignments/M1_6/JupyterTutorial_JL.pdf
```

[ ]:
```

```