Cheatography

PYTHON PANDAS Cheat Sheet

by sanjeev95 via cheatography.com/111141/cs/21621/

install and import

installing pandas

```
pip install pandas
```

pip install pandas

```
import pandas as pd
```

Reading and describing

pd -> pandas
df-> dataframe

to read a file into a dataframe

```
df= pd.rea d_c sv( 'fi len ame')
```

look at the first 5 lines

```
df.head()
```

to describe df

```
df.des cribe()
```

```
df.info()
```

to print all the column names

```
teleco m_d ata.co lumns
```

to get the dimension of df

```
df.shape
```

Sorting and filtering

Rows and columns

to delete a row - [axis=0 means rows]

```
new_df = df.dro p([ 2,3 ],axis = 0)
```

to delete a column- [axis=1 means columns]

```
new_df = df.dro p([ 'co l1' ,'c ol2
 col2
```

Df manipulation

create or edit a new column

```
df['ne w_c oln ame'] = 5   #this creates a new new column with all valu
```

create a new column

```
df['ne w_c oln ame'] = [list of values]   #this creates a new column wi
ow
```

NOTE : `df['ne w_c oln ame'] = [list of values]` throws an error if the no of items in

create or edit a new row

```
df.loc [in dex _of _row] = [list of items]
```

NOTE : `df.loc [in dex _of _row] = [list of items]` throws an error if the no of items in

Selection

`df[col]`	Returns column with label col as Series
`df[[col1, col2]]`	Returns multiple columns as a new DataFrame

	Country	Capital	Population
1	Belgium	Brussels	11190846
2	India	New Delhi	1303171035
3	Brazil	Brasilia	207847528

`df.ilo c([0], [0])` --> 'Belgium' | s.iloc[0] | Selection by position (0th position on row and column)

`df.loc ([0], ['Coun try'])` --> 'Belgium'

`df.ix[2]` -->

Country	Brazil
Capital	Brasilia
Population	207847528

`df.ix[1, 'Capital']` --> 'New Delhi'

df.iloc[0,:] | select First row

Data Cleaning

df.set_index('column_-one')	Change the index with a new column
`df.columns =['new_col_-name1','new_col_n-ame2','new_col_-name3']`	Rename columns
pd.isnull()	Checks for null Values, Returns Boolean Arrray
pd.notnull()	Opposite of pd.isnull()
df.dropna()	Drop all rows that contain null values
df.dropna(axis=1)	Drop all columns that contain null values
df.dropna(axis=1,thre-sh=n)	Drop all rows have have less than n non null
df.fillna(x)	Replace all null values with x

JOIN/COMBINE

sort

sorting can be done column wise - default is ascending

```
df.sor t_v alu es( by= 'Total day charge')
```

`df.sor t_v alu es( col1)`	Sort values by col1 in ascending order (use ascending =False for descending sort)
`df.sor t_v alu es( [co l1, col 2], asc end ing =[T rue ,Fa lse])`	Sort values by col1 in ascending order then col2 in descending order

Filtering

`df[con dition]` `#eg: df[df[ 'co l'] >5]`

`df[df[ 'col'] > 0.5]`	Rows where the column col is greater than 0.5
`df[(df [col] > 0.5) & (df[col] < 0.7)]`	Rows where 0.7 > col > 0.5

Inplace

NOTE

`df.mer ge(df2)` gives you a copy of df merged with df2. you may save it to a new variable. ex `df3=df.me rge (df2)`

if you want to merge df2 to df right away use inplace. `df.mer ge( df2 ,in pla ce= True)`

```
df1.ap pen d(df2)
```

```
pd.con cat ([df1, df2],a xis=1)
```

```
df1.jo in( df2 ,on =co l1, how ='i nner')
```

left = takes the index of left df
right =takes the index of left
outer = union of both keys
inner = intersection of both keys

By **sanjeev95**
cheatography.com/sanjeev95/

Published 20th January, 2020.
Last updated 22nd January, 2020.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com