* **Index**

df.set\_index(‘Col name’,inplace=True)

df.sort\_index()

df.reset\_index(inplace=True)

* **Subset**

series=df[‘Col name’]

df\_new=df[[‘Col name1’,’Col name2’]] (dataframe)

df\_new=df.loc[‘Index name 1’:’Index name 2’,:]

series.loc[‘Index name’]

* **Filter**

>, <, >=, <=, ==, !=

filter = df[‘Col name’]>10

string\_filter=df[‘Col name’].str.contains(‘text’)

df\_filtered=df[filter]

Combine multiple conditions with (filter1) & (filter2), (filter1) | (filter2)

* **Count values in a series**

df[‘Col name’].value\_counts()

* **Grouping**

grp= df.groupby(‘Col name’)

grp[‘Col name1’,’Col name2’].count()

* **Aggregate functions**

.nunique(),.sum(),.mean(),.max(),.min()

grp[‘Col name1’,’Col name2’].agg([‘min’,’max’,’mean’])

* **Join**

df3=pd.merge(df1,df2,how='outer',on='Col name')

**including when and how to use a self-join**

when- you want to get a value from another row of same table

df.join(df.drop('m\_ids',1).set\_index('e\_ids'),on='m\_ids',rsuffix='e\_names')

* **Sort**

df.sort\_values(by=[‘Col name1’,’Col name2’],ascending=True, inplace=True)

sorted(“series”)

* **Find, Fill, & Drop NaN**

df[df.isna().any(axis=1)] #find missing values

df.dropna(axis=0,subset=['Col name'],inplace=True)

df['Col name'].fillna(df['Col name'].mean(),inplace=True)

* **Duplicates**

df[df[‘Col name’].duplicated(keep=False)==True] #identify duplicates

df.drop([row#,row#],axis=0,inplace=True) #drop specific rows

df.drop\_duplicates(subset=’Col name’,keep=’first’,inplace=True) #keep first/last

* **Create a unique key**

df[‘key name’]=df1[‘Col Name’].astype(str)+’-’+df1[‘Col Name’].astype(str)

* **Concat data sources**

df=pd.concat([df1,df2],ignore\_index=True)