Summary

Grouping

Grouping involves combining data from several columns and grouping them based on a value of another column.

Item	year	Sale
tea	2010	1200
coffee	2010	1050
sugar	2010	500
tea	2011	1500
coffee	2011	1200
sugar	2011	1000
tea	2012	1230
coffee	2012	1300
sugar	2012	1420


```
coffee:
     item  year  sales
1  coffee  2010   1050
4  coffee  2011   1200
7  coffee  2012   1300

sugar:
    item  year  sales
2  sugar  2010    500
5  sugar  2011   1000
8  sugar  2012   1420

tea:
  item  year  sales
0  tea  2010   1200
3  tea  2011   1500
6  tea  2012   1230
```

Grouping by is useful for trying to find relationships between data and gathering insights and it is usually used with aggregate functions (mean ,max, median, count, … etc.).

How to group by:

```
In [5]: zoo.groupby('animal').count()
```

Out[5]:

animal	uniq_id	water_need
elephant	3	3
kangaroo	3	3
lion	4	4
tiger	5	5
zebra	7	7

Grouping by could also be done on multiple columns by passing in a list of column names instead.

Useful methods:

df_group.agg(): This method runs aggregate function(s) on all columns by default. When wanting to apply multiple functions, then they are passed as a list. When wanting to apply to specific columns the columns are passed in a dictionary as the keys where the values are the function(s) that should be applied to them.

```
In [35]: data.groupby('month', as_index=False).agg({"duration": "sum"})
```

Out[35]:

	month	duration
0	2014-11	26639.441
1	2014-12	14641.870
2	2015-01	18223.299
3	2015-02	15522.299
4	2015-03	22750.441

df_group.apply(function):

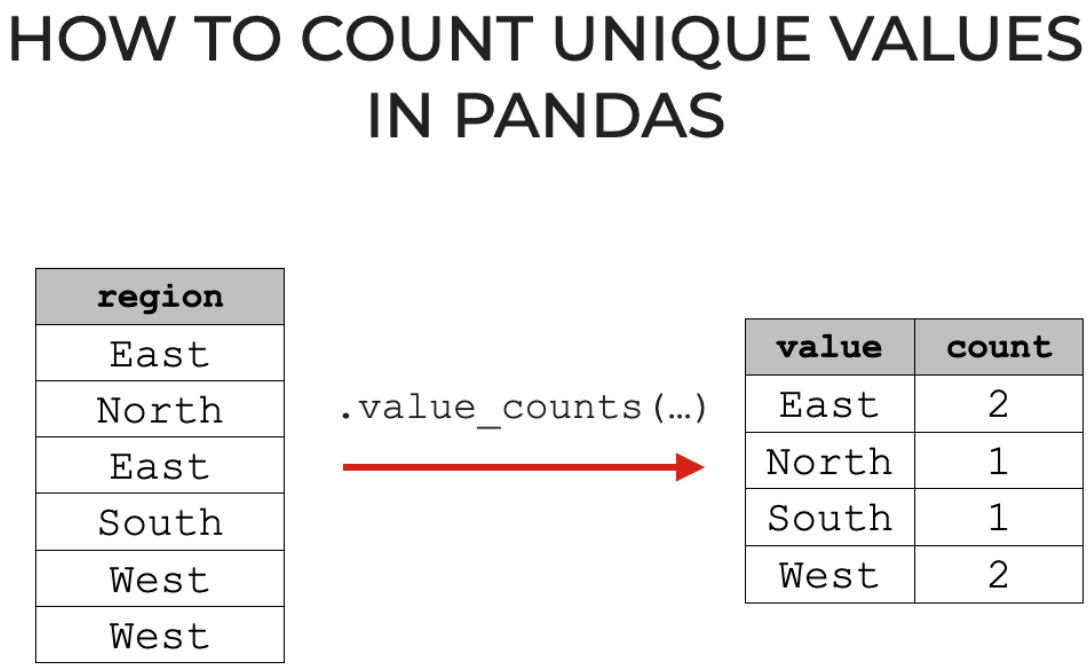
This function applies a given function to every column in a group and returns a series or dataframe.

```
#find relative frequency of each team name in DataFrame
df.groupby('team').apply(lambda x: x['team'].count() / df.shape[0])

team
A    0.428571
B    0.571429
dtype: float64
```

df_group.value_counts():

This method is useful when trying to count unique values occurrence in each group and a normalize keyword argument may be set to true to get the ratio of the occurrence of a value to the whole.



Date time

Date time data has a data type in pandas that supports operations that would be otherwise hard to implement like (date1 > date2) or date > 2020. Date time data could be interpreted in loading by passing additional arguments to read_csv() or after loading by converting data using pd.to_datetime() . A data parser or a format string may be required to add to the list of arguments if pandas couldn't figure out how the date and time are formatted. df.resample() resamples the data if the date and time where the index for the date frame and resampling could be done by day month, year , … etc.