

```
In [1]: import pandas as pd
import numpy as np
import os
import glob
```

```
In [2]: print('pandas version : ',pd.__version__)
print('numpy version : ',np.__version__)

pandas version : 1.1.4
numpy version : 1.18.2
```

ترکیب فایل های امتیازات لیگ انگلیس از سال 2000 تا 2019

از لینک زیر امکان دانلود امتیازات لیگ های مختلف رو دارید که در این مثال لیگ برتر انگلیس استفاده شده

```
In [3]: # www.football-data.co.uk/data
```

```
In [4]: #Merging csv files

# خواندن و ذخیره آدرس فایل ما
path=os.getcwd()
files=glob.glob(os.path.join(path, '*.csv'))

# انتخاب ستون های مورد نظر
column_names=['Date', 'HomeTeam', 'AwayTeam', 'FTHG', 'FTAG', 'FTR', 'HTHG', 'HTAG', 'HTR', 'HY', 'AY', 'HR', 'AR']
datalist=[]

# اضافه کردن اطلاعات هر فایل به لیست بالا
از نام فایل ما به عنوان متمایز کننده فصل ما استفاده می کنیم
for f in files:
    data=pd.read_csv(f,header=0,usecols=column_names)
    data['season']=os.path.basename(f).split('.')[0]
    datalist.append(data)
```

```
In [5]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7509 entries, 0 to 7508
Data columns (total 14 columns):
# Column Non-Null Count Dtype
---
0 Date 7508 non-null object
1 HomeTeam 7508 non-null object
2 AwayTeam 7508 non-null object
3 FTHG 7508 non-null float64
4 FTAG 7508 non-null float64
5 FTR 7508 non-null object
6 HTHG 7508 non-null float64
7 HTAG 7508 non-null float64
8 HTR 7508 non-null object
9 HY 7508 non-null float64
10 AY 7508 non-null float64
11 HR 7508 non-null float64
12 AR 7508 non-null float64
13 season 7509 non-null object
dtypes: float64(8), object(6)
memory usage: 821.4+ KB
```

معانی از نحوه عملکرد melt قبل از اعمال تغییرات

```
In [6]: sample = pd.DataFrame({'HomeTeam': {0: 'A', 1: 'B', 2: 'A'},
                              'AwayTeam': {0: 'B', 1: 'C', 2: 'C'},
                              'Value1': {0: 1, 1: 3, 2: 5},
                              'Value2': {0: 2, 1: 4, 2: 6},
                              })

print(sample)
print('-----')
print('After melting :')
print('-----')
print(sample.melt(id_vars=['Value1', 'Value2'], value_vars=['HomeTeam', 'AwayTeam']))

HomeTeam AwayTeam Value1 Value2
0 A B 1 2
1 B C 3 4
2 A C 5 6

-----
After melting :
-----
Value1 Value2 variable value
0 1 2 HomeTeam A
1 3 4 HomeTeam B
2 5 6 HomeTeam A
3 1 2 AwayTeam B
4 3 4 AwayTeam C
5 5 6 AwayTeam C
```

```
In [7]: df.head()
```

Out[7]:

	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HY	AY	HR	AR	season
0	19/08/00	Charlton	Man City	4.0	0.0	H	2.0	0.0	H	1.0	2.0	0.0	0.0	20002001
1	19/08/00	Chelsea	West Ham	4.0	2.0	H	1.0	0.0	H	1.0	2.0	0.0	0.0	20002001
2	19/08/00	Coventry	Middlesbrough	1.0	3.0	A	1.0	1.0	D	5.0	3.0	1.0	0.0	20002001
3	19/08/00	Derby	Southampton	2.0	2.0	D	1.0	2.0	A	1.0	1.0	0.0	0.0	20002001
4	19/08/00	Leeds	Everton	2.0	0.0	H	2.0	0.0	H	1.0	3.0	0.0	0.0	20002001

کپی رکورد ما با استفاده از melt و امتیازدهی

```
In [8]: # Copying (melting) records to have 'Away Team' column result in rows too

# ایجاد یک رکورد
df['Home']=df['HomeTeam']
df['Away']=df['AwayTeam']
df=df.melt(id_vars=['Date', 'HomeTeam', 'AwayTeam', 'FTHG', 'FTAG', 'FTR', 'HTHG', 'HTAG', 'HTR', 'HY', 'AY', 'HR', 'AR', 'season'],
           value_vars=['Home', 'Away'],
           var_name='HomeAway',
           value_name='Team')

# محاسبه امتیاز مسابقه
@np.vectorize
def result(team, home_team, home_goal, away_goal):
    if team==home_team:
        if home_goal>away_goal:
            return 3
        elif home_goal<away_goal:
            return 0
        else:
            return 1
    else:
        if home_goal>away_goal:
            return 0
        elif home_goal<away_goal:
            return 3
        else:
            return 1

df['MatchResult']=result(df['Team'],df['HomeTeam'],df['FTHG'],df['FTAG'])
df['home_goal']=np.where(df['Team']==df['HomeTeam'],df['FTHG'],df['FTAG'])
df['away_goal']=np.where(df['Team']!=df['HomeTeam'],df['FTHG'],df['FTAG'])
print(df.shape)
df.sort_values(['Date']).head(20)

(15018, 19)
```

Out[8]:

	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HY	AY	HR	AR	season	HomeAway
7715	01/01/01	Chelsea	Aston Villa	1.0	0.0	H	1.0	0.0	H	0.0	1.0	0.0	0.0	20002001	Away
213	01/01/01	Sunderland	Ipswich	4.0	1.0	H	1.0	1.0	D	1.0	5.0	0.0	0.0	20002001	Home
210	01/01/01	Leicester	Bradford	1.0	2.0	A	1.0	2.0	A	0.0	3.0	0.0	0.0	20002001	Home
209	01/01/01	Leeds	Middlesbrough	1.0	1.0	D	0.0	1.0	A	2.0	3.0	0.0	0.0	20002001	Home
208	01/01/01	Derby	Everton	1.0	0.0	H	1.0	0.0	H	2.0	3.0	0.0	0.0	20002001	Home
207	01/01/01	Coventry	Man City	1.0	1.0	D	0.0	0.0	D	1.0	2.0	0.0	0.0	20002001	Home
206	01/01/01	Chelsea	Aston Villa	1.0	0.0	H	1.0	0.0	H	0.0	1.0	0.0	0.0	20002001	Home
205	01/01/01	Charlton	Arsenal	1.0	0.0	H	1.0	0.0	H	1.0	1.0	0.0	0.0	20002001	Home
212	01/01/01	Man United	West Ham	3.0	1.0	H	2.0	0.0	H	1.0	2.0	0.0	0.0	20002001	Home
7714	01/01/01	Charlton	Arsenal	1.0	0.0	H	1.0	0.0	H	1.0	1.0	0.0	0.0	20002001	Away
7716	01/01/01	Coventry	Man City	1.0	1.0	D	0.0	0.0	D	1.0	2.0	0.0	0.0	20002001	Away
7717	01/01/01	Derby	Everton	1.0	0.0	H	1.0	0.0	H	2.0	3.0	0.0	0.0	20002001	Away
7718	01/01/01	Leeds	Middlesbrough	1.0	1.0	D	0.0	1.0	A	2.0	3.0	0.0	0.0	20002001	Away
7719	01/01/01	Leicester	Bradford	1.0	2.0	A	1.0	2.0	A	0.0	3.0	0.0	0.0	20002001	Away
7720	01/01/01	Liverpool	Southampton	2.0	1.0	H	1.0	1.0	D	0.0	1.0	0.0	0.0	20002001	Away
7721	01/01/01	Man United	West Ham	3.0	1.0	H	2.0	0.0	H	1.0	2.0	0.0	0.0	20002001	Away
7722	01/01/01	Sunderland	Ipswich	4.0	1.0	H	1.0	1.0	D	1.0	5.0	0.0	0.0	20002001	Away
211	01/01/01	Liverpool	Southampton	2.0	1.0	H	1.0	1.0	D	0.0	1.0	0.0	0.0	20002001	Home
8091	01/01/02	Middlesbrough	Everton	1.0	0.0	H	0.0	0.0	D	1.0	2.0	0.0	0.0	20012002	Away
8092	01/01/02	Sunderland	Aston Villa	1.0	1.0	D	0.0	0.0	D	1.0	3.0	0.0	0.0	20012002	Away

```
In [9]: # محاسبه لیگ مورد نظر
league=df.loc[df['season']=='20182019']

# ساخت جدول
def table_league(frame):
    return pd.DataFrame({'Played':np.size(frame['Team']),
                        'Won':np.sum(frame['MatchResult']==3),
                        'Drawn':np.sum(frame['MatchResult']==1),
                        'Lost':np.sum(frame['MatchResult']==0),
                        'GF':np.sum(frame['home_goal']),
                        'GA':np.sum(frame['away_goal']),
                        'GD':np.sum(frame['home_goal'])-np.sum(frame['away_goal']),
                        'Points':np.sum(frame['MatchResult'])
                        },index=frame['Team'].unique())

cal_league=league.groupby(['Team'])
result_table=cal_league.apply(table_league)
result_table.sort_values(['Points'],ascending=False)
```

Out[9]:

Team	Played	Won	Drawn	Lost	GF	GA	GD	Points	
Man City	Man City	38	32	2	4	95.0	23.0	72.0	98
Liverpool	Liverpool	38	30	7	1	89.0	22.0	67.0	97
Chelsea	Chelsea	38	21	9	8	63.0	39.0	24.0	72
Tottenham	Tottenham	38	23	2	13	67.0	39.0	28.0	71
Arsenal	Arsenal	38	21	7	10	73.0	51.0	22.0	70
Man United	Man United	38	19	9	10	65.0	54.0	11.0	66
Wolves	Wolves	38	16	9	13	47.0	46.0	1.0	57
Everton	Everton	38	15	9	14	54.0	46.0	8.0	54
West Ham	West Ham	38	15	7	16	52.0	55.0	-3.0	52
Leicester	Leicester	38	15	7	16	51.0	48.0	3.0	52
Watford	Watford	38	14	8	16	52.0	59.0	-7.0	50
Crystal Palace	Crystal Palace	38	14	7	17	51.0	53.0	-2.0	49
Bournemouth	Bournemouth	38	13	6	19	56.0	70.0	-14.0	45
Newcastle	Newcastle	38	12	9	17	42.0	48.0	-6.0	45
Burnley	Burnley	38	11	7	20	45.0	68.0	-23.0	40
Southampton	Southampton	38	9	12	17	45.0	65.0	-20.0	39
Brighton	Brighton	38	9	9	20	35.0	60.0	-25.0	36
Cardiff	Cardiff	38	10	4	24	34.0	69.0	-35.0	34
Fulham	Fulham	38	7	5	26	34.0	81.0	-47.0	26
Huddersfield	Huddersfield	38	3	7	28	22.0	76.0	-54.0	16

<https://www.datasense.ir/>