**福布斯数据处理实训**

1. 导入数据
2. 更新列名
3. 找出'Sales','Profits'及'Assets'中非数字的内容
4. 用replace()方法替换“Sales”列中含有字母的内容
5. 查看“Assets”列中非数字的内容
6. 替换非数字的内容，以及替换千分位间隔符号
7. 发现“Profits”中有NaN值，需要先进行替换
8. 将NaN值填充为 0
9. 将“Profits”列中非数字的内容进行替换，并查看替换后的结果
10. 将sting类型的数字转换为数据类型，这里使用 pd.to\_numeric() 方法
11. 拆分"Company\_cn\_en"列，新生成两列，分别为公司英文名称和中文名称
12. 拆分"Country\_cn\_en"列，新生成两列，分别为国家中文名称和英文名称
13. 由于国家的英文名称中，最后有半个括号，需要去除，用 Series.str.slice()方法
14. 参数表示选取从开始到倒数第二个，即不要括号")"
15. 考虑的中国的企业有区分为中国大陆，中国香港，中国台湾
16. 对应的国家英文名称也需要修改下
17. 中国大陆：CN；中国香港：CN-HK；中国台湾：CN-TA
18. 将列名进行重新排序
19. 导入数据2007年的数据，原始数据的单位为十亿美元

df\_2007 = pd.read\_csv('./data/data\_forbes\_2007.csv', encoding='gbk', thousands=',')

print('the shape of DataFrame: ', df\_2007.shape)

print(df\_2007.dtypes)

df\_2007.head(3)

import pandas as pd

data = pd.read\_csv('C:\\Users\\yybwy\\Desktop\\data\_forbes\_2007.csv',sep=',',encoding='gbk')

data

1. 更新columns的命名

column\_update = ['Year', 'Rank', 'Company\_cn\_en', 'Country\_cn\_en',

'Industry\_cn', 'Sales', 'Profits', 'Assets', 'Market\_value']

df\_2007.columns = column\_update

df\_2007.head(3)

data1 = data.rename(columns={

"年份":"Year",

"排名(Rank)": "Rank",

"公司名称(Company)": "Company",

"所在国家或地区(Country)":"Country",

"所在行业(Industry)": "Industry",

"销售收入(Sales)":"Sales",

"利润(Profits)": "Profits",

"总资产(Assets)":"Assets",

"市值(Market Vaue)":"Market Vaue"

})

data1

1. 通过前面的分析可看出，只有“Market\_value”是数字类型，找出'Sales','Profits'及'Assets'中非数字的内容

df\_2007[df\_2007['Sales'].str.contains('.\*[A-Za-z]', regex=True)]

data1[data1.Sales.str.contains('.\*[A-Za-z]')]

data1[(data1.Profits.isnull()) | (data1.Profits.str.contains('.\*[A-Za-z]'))]

data1[data1.Assets.str.contains('.\*[A-Za-z]',regex=True)]

1. 更新columns的命名

df\_2007['Sales'] = df\_2007['Sales'].replace('([A-Za-z])', '', regex=True)

1. 查看替换后的结果

df\_2007.loc[[117,616,880], :]

1. 查看“Assets”列中非数字的内容

df\_2007[df\_2007['Assets'].str.contains('.\*[A-Za-z]', regex=True)]

1. 替换非数字的内容，以及替换千分位间隔符号

# 将数字后面的字母进行替换

df\_2007['Assets'] = df\_2007['Assets'].replace('([A-Za-z])', '', regex=True)

# 千分位数字的逗号被识别为string了，需要替换

df\_2007['Assets'] = df\_2007['Assets'].replace(',', '', regex=True)

df\_2007.loc[616, :]

1. 发现“Profits”中有NaN值，需要先进行替换

df\_2007[pd.isnull(df\_2007['Profits'])]

1. 将NaN值填充为 0

df\_2007['Profits'].fillna(0, inplace=True)

df\_2007.loc[[958,1440,1544,1912], :]

1. 将“Profits”列中非数字的内容进行替换，并查看替换后的结果

df\_2007['Profits'] = df\_2007['Profits'].replace('([A-Za-z])', '', regex=True)

df\_2007.loc[[117,616,880], :]

1. 将sting类型的数字转换为数据类型，这里使用 pd.to\_numeric() 方法

df\_2007['Sales'] = pd.to\_numeric(df\_2007['Sales'])

df\_2007['Profits'] = pd.to\_numeric(df\_2007['Profits'])

df\_2007['Assets'] = pd.to\_numeric(df\_2007['Assets'])

df\_2007.dtypes

1. 拆分"Company\_cn\_en"列，新生成两列，分别为公司英文名称和中文名称

df\_2007['Company\_en'],df\_2007['Company\_cn'] = df\_2007['Company\_cn\_en'].str.split('/', 1).str

print(df\_2007['Company\_en'][:5])

print(df\_2007['Company\_cn'] [-5:])

df\_2007.tail(3)

1. 拆分"Country\_cn\_en"列，新生成两列，分别为国家中文名称和英文名称

df\_2007['Country\_cn'],df\_2007['Country\_en'] = df\_2007['Country\_cn\_en'].str.split('(', 1).str

print(df\_2007['Country\_cn'][:5])

print(df\_2007['Country\_en'][-5:])

14. 由于国家的英文名称中，最后有半个括号，需要去除，用 Series.str.slice()方法

参数表示选取从开始到倒数第二个，即不要括号")"

df\_2007['Country\_en'] = df\_2007['Country\_en'].str.slice(0,-1)

df\_2007.head(3)

15. 考虑的中国的企业有区分为中国大陆，中国香港，中国台湾

对应的国家英文名称也需要修改下

中国大陆：CN；中国香港：CN-HK；中国台湾：CN-TA

df\_2007[df\_2007['Country\_cn'].str.contains('中国',regex=True)]

df\_2007['Country\_en'] = df\_2007['Country\_en'].replace(['HK.\*','TA'],['CN-HK', 'CN-TA'],regex=True)

df\_2007[df\_2007['Country\_en'].str.contains('CN',regex=True)]

df\_2007['Industry\_en'] = ''

df\_2007.tail(3)

16. 将列名进行重新排序

columns\_sort = ['Year', 'Rank', 'Company\_cn\_en','Company\_en',

'Company\_cn', 'Country\_cn\_en', 'Country\_cn',

'Country\_en', 'Industry\_cn', 'Industry\_en',

'Sales', 'Profits', 'Assets', 'Market\_value']

# 按指定list重新将columns进行排序

df\_2007 = df\_2007.reindex(columns=columns\_sort)

print(df\_2007.shape)

print(df\_2007.dtypes)

df\_2007.head(3)