# Python实验 数据预处理

班级:2017211314 学号:2017213508 学生:蒋雪枫

**链家网爬虫：**

1. 核心代码：

我们在爬虫的时候就进行数据预处理，采用正则表达式的方法获取数据：

    def parse(*self*,*response*):

        item=MyItem()

*for* each in response.xpath("/html/body/div[4]/ul[2]/\*"):

            item['name']=each.xpath("a/@title").extract()

            item['location1']=each.xpath("div/div[2]/span[1]/text()").extract()

            item['location2'] = each.xpath("div/div[2]/span[2]/text()").extract()

            item['location3'] = each.xpath("div/div[2]/a/text()").extract()

            item['huxing']=each.xpath("div/a/span[1]/text()").extract()

            item['area']=each.xpath("div/div[3]/span/text()").extract()

            item['totality']=each.xpath("div/div[6]/div[2]/text()").extract();

            item['meanPrice']=each.xpath("div/div[6]/div[1]/span[1]/text()").extract()

*if*(item['area']):

                match1=re.search(r'[0-9]+',item['area'][0])

*if* match1:

                    item['area'][0]=eval(match1.group(0))

*if*(item['totality']):

                match1=re.search(r'[0-9]+',item['totality'][0])

*if* match1:

                    item['totality'][0]=eval(match1.group(0))

*if*(item['meanPrice']):

                match1=re.search(r'[0-9]+',item['meanPrice'][0])

*if* match1:

                    item['meanPrice'][0]=eval(match1.group(0))

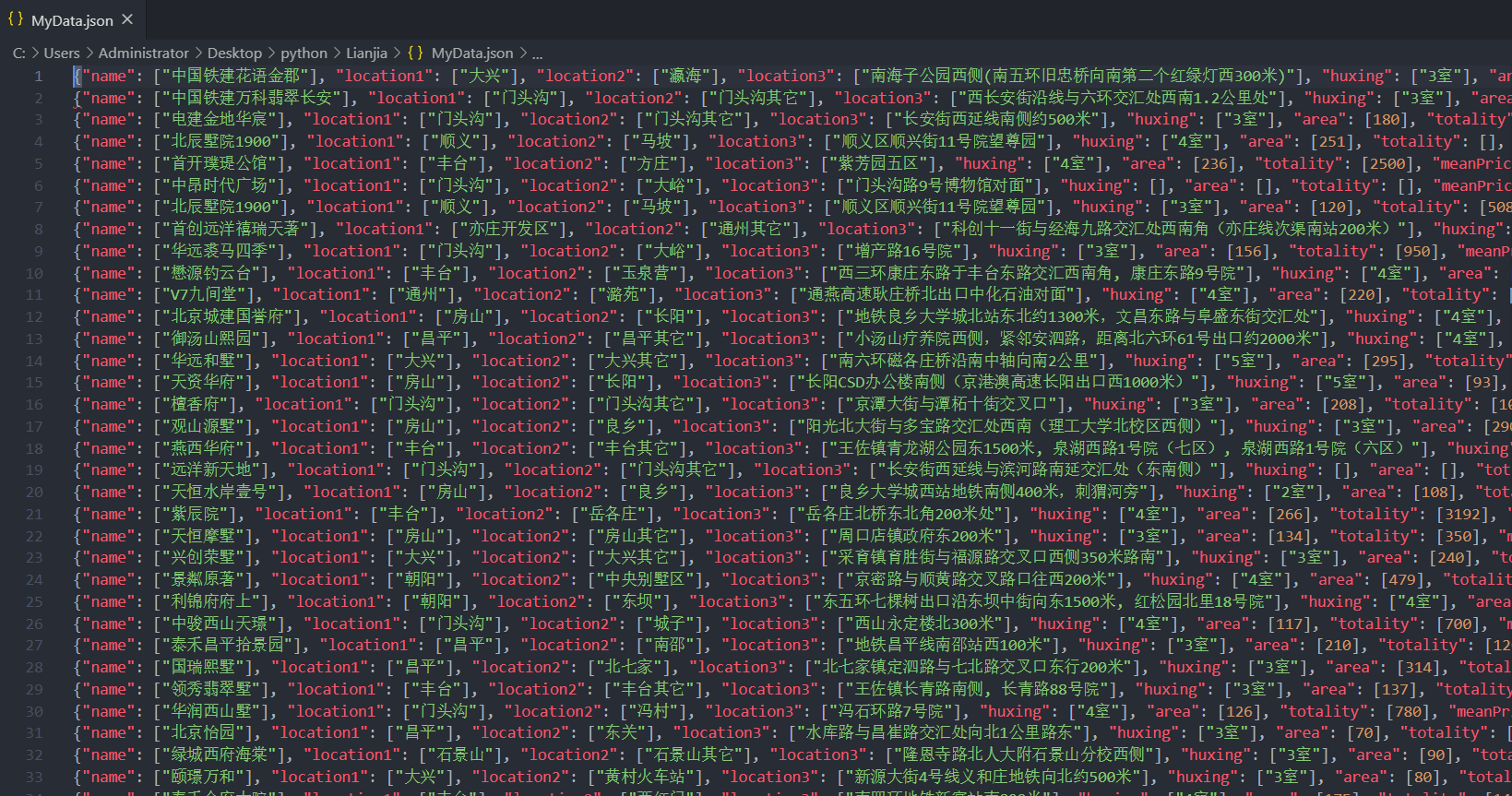
*if*(item['meanPrice'][0]<10000):

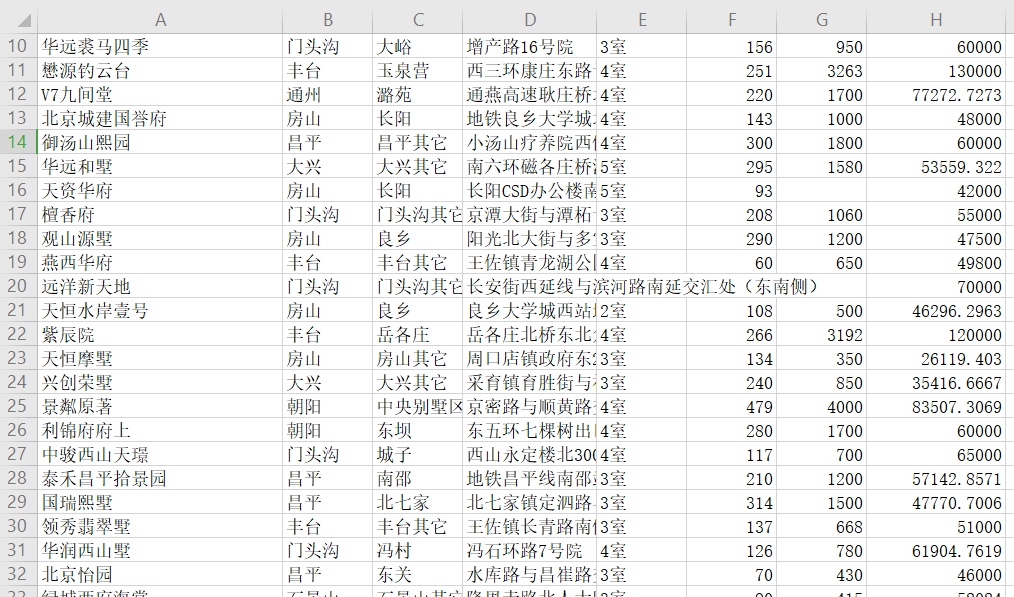
#在北京不可能有低于10000元/平的房子，如果有，那一定是单位搞错了

                    item['meanPrice'][0]=round(item['totality'][0]\*10000/item['area'][0],4)

*yield*(item)

1. 爬取后的csv文件截图：





**PM指数分析：**

源代码：

*import* numpy *as* np

*import* pandas *as* pd

*import* time

*import* matplotlib.pyplot *as* plt

*#1.打开CSV文件*

fileNameStr = 'BeijingPM20100101\_20151231.csv'

df = pd.read\_csv(fileNameStr,*encoding*='utf-8',*dtype*=str)

*#2010 0101 - 2015 1231*

*#6年 12个月*

print("info============================================================================")

print(df.info())

print("=======================================")

start = time.time()

account=[[0 *for* i in range(13)] *for* j in range(6)]

counters=[[0 *for* i in range(13)] *for* j in range(6)]

*for* i in range(52584):

*if*(eval(df['year'][i])==2010):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[0][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[0][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[0][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[0][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[0][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[0][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[0][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[0][eval(df['month'][i])]+=1

*elif*(eval(df['year'][i])==2011):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[1][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[1][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[1][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[1][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[1][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[1][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[1][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[1][eval(df['month'][i])]+=1

*elif*(eval(df['year'][i])==2012):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[2][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[2][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[2][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[2][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[2][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[2][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[2][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[2][eval(df['month'][i])]+=1

*elif*(eval(df['year'][i])==2013):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[3][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[3][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[3][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[3][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[3][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[3][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[3][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[3][eval(df['month'][i])]+=1

*elif*(eval(df['year'][i])==2014):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[4][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[4][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[4][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[4][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[4][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[4][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[4][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[4][eval(df['month'][i])]+=1

*elif*(eval(df['year'][i])==2015):

*if* not(df['PM\_Dongsi'][i] is np.nan):

            account[5][eval(df['month'][i])]+=int(df['PM\_Dongsi'][i])

            counters[5][eval(df['month'][i])]+=1

*if* not (df['PM\_Dongsihuan'][i] is np.nan):

            account[5][eval(df['month'][i])]+=int(df['PM\_Dongsihuan'][i])

            counters[5][eval(df['month'][i])]+=1

*if* not(df['PM\_Nongzhanguan'][i] is np.nan):

            account[5][eval(df['month'][i])]+=int(df['PM\_Nongzhanguan'][i])

            counters[5][eval(df['month'][i])]+=1

*if* not(df['PM\_US Post'][i] is np.nan):

            account[5][eval(df['month'][i])]+=int(df['PM\_US Post'][i])

            counters[5][eval(df['month'][i])]+=1

list\_year=[]

*for* i in range(6):

    sumt=0

*for* j in range(1,13):

        print("year:{},month:{},meanIndex={:.2f}".format(i+2010,j,account[i][j]/counters[i][j]))

        sumt+=account[i][j]/counters[i][j]

    print("year{}'s meanIndex={:.2f}".format(i+2010,sumt/12))

    list\_year.append(round(sumt,2)/12)

print("Full process Time:",time.time()-start)

year=[2010,2011,2012,2013,2014,2015]

*#调用plt.plot来画图,横轴纵轴两个参数即可*

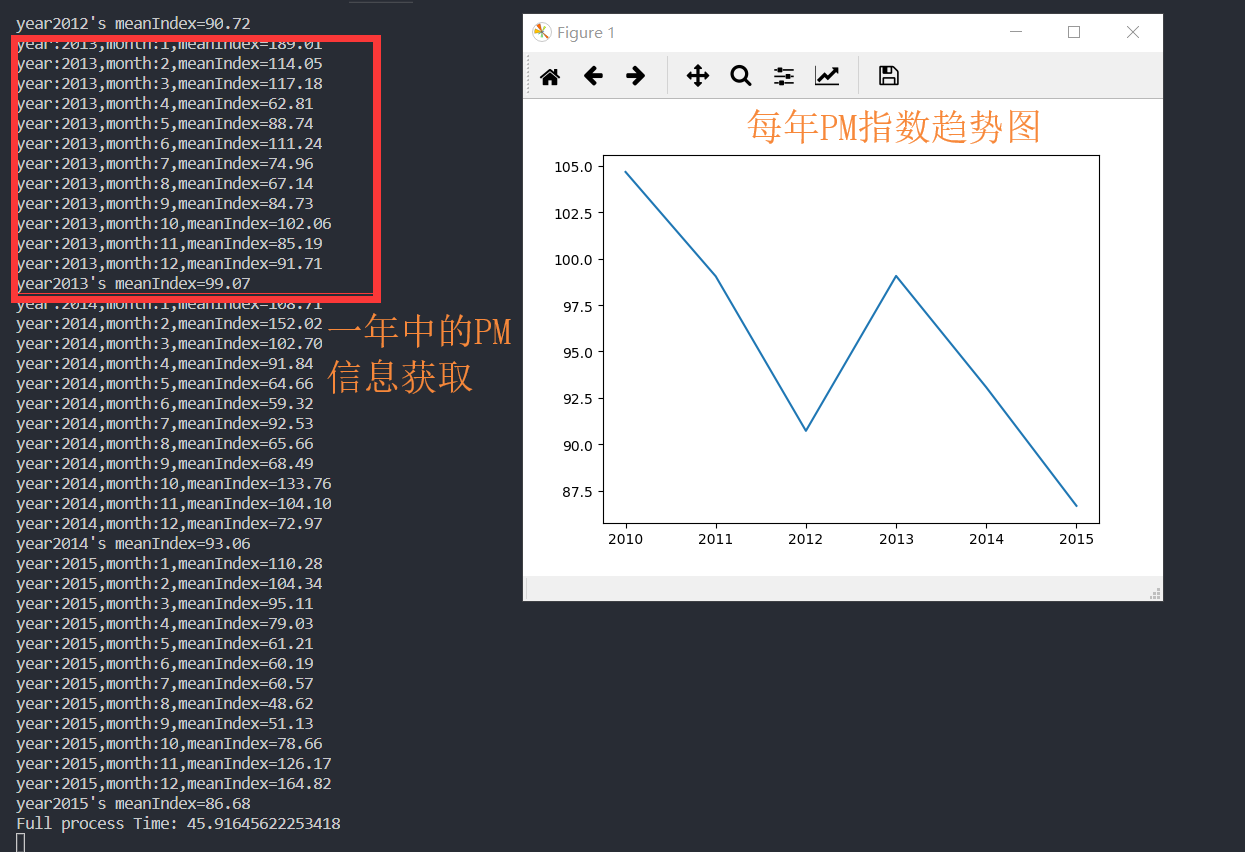
plt.plot(year,list\_year)

plt.title=("Year Trend Graph")

*# 用show展现出来图*

plt.show()

运行结果：



info============================================================================

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 52584 entries, 0 to 52583

Data columns (total 18 columns):

No 52584 non-null object

year 52584 non-null object

month 52584 non-null object

day 52584 non-null object

hour 52584 non-null object

season 52584 non-null object

PM\_Dongsi 25052 non-null object

PM\_Dongsihuan 20508 non-null object

PM\_Nongzhanguan 24931 non-null object

PM\_US Post 50387 non-null object

DEWP 52579 non-null object

HUMI 52245 non-null object

PRES 52245 non-null object

TEMP 52579 non-null object

cbwd 52579 non-null object

Iws 52579 non-null object

precipitation 52100 non-null object

Iprec 52100 non-null object

dtypes: object(18)

memory usage: 7.2+ MB

None

=======================================

year:2010,month:1,meanIndex=90.40

year:2010,month:2,meanIndex=97.24

year:2010,month:3,meanIndex=94.05

year:2010,month:4,meanIndex=80.07

year:2010,month:5,meanIndex=87.07

year:2010,month:6,meanIndex=109.04

year:2010,month:7,meanIndex=123.43

year:2010,month:8,meanIndex=97.68

year:2010,month:9,meanIndex=122.79

year:2010,month:10,meanIndex=118.78

year:2010,month:11,meanIndex=138.38

year:2010,month:12,meanIndex=97.12

year2010's meanIndex=104.67

year:2011,month:1,meanIndex=44.87

year:2011,month:2,meanIndex=150.29

year:2011,month:3,meanIndex=57.99

year:2011,month:4,meanIndex=91.72

year:2011,month:5,meanIndex=65.11

year:2011,month:6,meanIndex=108.79

year:2011,month:7,meanIndex=107.39

year:2011,month:8,meanIndex=103.73

year:2011,month:9,meanIndex=94.97

year:2011,month:10,meanIndex=145.56

year:2011,month:11,meanIndex=109.43

year:2011,month:12,meanIndex=108.72

year2011's meanIndex=99.05

year:2012,month:1,meanIndex=118.92

year:2012,month:2,meanIndex=84.44

year:2012,month:3,meanIndex=96.47

year:2012,month:4,meanIndex=87.84

year:2012,month:5,meanIndex=90.97

year:2012,month:6,meanIndex=96.63

year:2012,month:7,meanIndex=80.65

year:2012,month:8,meanIndex=81.17

year:2012,month:9,meanIndex=59.95

year:2012,month:10,meanIndex=94.95

year:2012,month:11,meanIndex=87.44

year:2012,month:12,meanIndex=109.19

year2012's meanIndex=90.72

year:2013,month:1,meanIndex=189.01

year:2013,month:2,meanIndex=114.05

year:2013,month:3,meanIndex=117.18

year:2013,month:4,meanIndex=62.81

year:2013,month:5,meanIndex=88.74

year:2013,month:6,meanIndex=111.24

year:2013,month:7,meanIndex=74.96

year:2013,month:8,meanIndex=67.14

year:2013,month:9,meanIndex=84.73

year:2013,month:10,meanIndex=102.06

year:2013,month:11,meanIndex=85.19

year:2013,month:12,meanIndex=91.71

year2013's meanIndex=99.07

year:2014,month:1,meanIndex=108.71

year:2014,month:2,meanIndex=152.02

year:2014,month:3,meanIndex=102.70

year:2014,month:4,meanIndex=91.84

year:2014,month:5,meanIndex=64.66

year:2014,month:6,meanIndex=59.32

year:2014,month:7,meanIndex=92.53

year:2014,month:8,meanIndex=65.66

year:2014,month:9,meanIndex=68.49

year:2014,month:10,meanIndex=133.76

year:2014,month:11,meanIndex=104.10

year:2014,month:12,meanIndex=72.97

year2014's meanIndex=93.06

year:2015,month:1,meanIndex=110.28

year:2015,month:2,meanIndex=104.34

year:2015,month:3,meanIndex=95.11

year:2015,month:4,meanIndex=79.03

year:2015,month:5,meanIndex=61.21

year:2015,month:6,meanIndex=60.19

year:2015,month:7,meanIndex=60.57

year:2015,month:8,meanIndex=48.62

year:2015,month:9,meanIndex=51.13

year:2015,month:10,meanIndex=78.66

year:2015,month:11,meanIndex=126.17

year:2015,month:12,meanIndex=164.82

year2015's meanIndex=86.68

Full process Time: 45.91645622253418