1. pandas模块处理excel：
2. import pandas as pd
3. df = pd. read\_excel( ‘xxx.xls’[, sheet\_name = ‘student’]) 读取指定文件的表单[默认第一个]，返回DataFrame对象
4. df.to\_excel(‘xxx.xls’) 将Series或DataFrame对象输出到Excel文件
5. Series对象：由一组数据（values）和一组行索引（index）组成
6. DataFrame对象：表格型数据类型
   1. df = pd.DataFrame(data[ ,columns = [ ],index = [ ] ]) 创建DataFrame对象，columns、index分别为列、行索引
      1. data: 可用值为等长列表的字典或嵌套字典的形式创建。以下one，two为行索引；state，year为列索引
         1. 字典：data = { ‘state’ : [‘open’, ‘close’], ‘year’ : [2000, 2001], ‘pop’ : [1.5, 1.7]}
         2. 嵌套字典：data = {‘state’ : [‘one’ : ‘open’, two : ‘close’], ‘year’ : [‘two’ : 2001, ‘one’ : 2000]}
   2. df[‘state’] 获取列索引为state的数据，返回Series对象
   3. df[[‘state’, ‘date’]] 获取索引为state和date的数据，返回DataFrame对象
   4. df[‘year’] = 2012 修改列索引为year的数据为2012
      1. 若用Series对象赋值，会按指定行索引赋值
      2. 为不存在的列赋值，会自动创建新列
   5. df = df[df['业务日期'].notnull() & df['金额'] !=0] 去除无效数据
   6. 按日期统计：
      1. df[‘date‘] = pd.to\_datetime(df[‘date’]) 将date列的数据转换为日期类型
      2. df\_date = df.set\_index(‘date’, drop = True) 设置date列为索引，并删去原数据中的date列
      3. df\_date\_mouth = df\_date.resample(‘M’).sum().to\_period(‘M’) 返回按月统计的DataFrame对象
         1. .resample(‘M’).sum() 将同月的日期改为此月[最后一天]后合并。w：周，M：月，Q：季度，A：年
            1. MS, QS, AS 改为此月/季度/年的第一天
         2. .to\_period(‘M’) 按月显示数据（不显示几日） W：月，Q：季度，A：年
7. numpy:
   1. ndarray类：
8. matplotlib绘制数据模型：
9. import matplotlib.pyplot as plt
10. 绘制线性模型：
11. clf = linear\_model.LinearRegression(fit\_intercept=True, nomalize=False) 创建线性模型
12. x = pd.DataFrame(df\_date\_mouth[‘xx’]) 获取所需数据并转换为DataFrame对象
13. y = pd.DataFrame(df\_date\_mouth[‘aa’])
14. clf.fit(x, y) 拟合线性模型
15. k = clf.coef\_ 回归系数（斜率），numpy.ndarray类型
16. b = clf.intercept\_ 截距，numpy.ndarray类型
17. y\_pred = clf.predict(x) 获得回归线的y值
18. plt.rc(‘font’, family=’SimHei’, size=10) 设置字体
19. plt.figure(‘销售收入分析’) 设置标题

plt.scatter(x, y, color = ‘red’) 绘制散点图

plt.plot(x, y\_pred, color=’blue’, linewidth=1.5) 绘制回归线

plt.ylabel(u’y轴标签’)

plt.xlabel(u’x轴标签’)

plt.show()