```python
#201009_习题讲解1
# 请实现一个程序，实现如下需求点：
# 1.程序开始的时候提示用户输入学生年龄信息 格式如下：
# Jack Green ,   21  ;  Mike Mos, 9;
# 我们假设 用户输入 上面的信息，必定会遵守下面的规则：
#   学生信息之间用分号隔开（分号前后可能有不定数量的空格），
#   每个学生信息里的 姓名和 年龄之间用 逗号隔开（逗号前后可能有不定数量的空格）
#
# 2. 程序随后将输入的学生信息分行显示，格式如下
# Jack Green :   21;
# Mike Mos   :   09;
# 学生的姓名要求左对齐，宽度为20， 年龄信息右对齐，宽度为2位，不足前面补零
#第一步,拆分原始字符串,以分号作为分割符,分割每个字段Jack Green ,   21
#第二步,进一步拆分,将姓名和年龄分开
#第三步,按照题目要求格式化字符串并打印
# str1='Jack Green ,   21  ;  Mike Mos, 9;'  #原始字符串
# list1=str1.split(';')  #进行第一步切割,形成一个列表
# for one in list1:  #遍历这个列表
#     if one!='':  #只处理不为空的元素
#         # name = one.split(',')[0]
#         # age = one.split(',')[1]
#         name,age=one.split(',')  #用两个变量分别接收列表中的两个元素
#         name=name.strip()  #去掉前后的空格
#         age=int(age.strip())  #去掉前后的空格,并转化为int型
#         print(f'{name:<20}:   {age:>02};')

# 1.下面的log变量记录了云服务器上 当天上传的文件信息
# 其中第一列是文件名，第二列是文件大小
#
# 请编写一个程序，统计出不同类型的 文件的大小总和
# 比如：
# jpeg  9988999
# json   324324
# png   2423233

log = '''
f20180111014341/i_51a7hC3w.jpeg 169472  FrITJxleSP7wUD-MWw-phL_KP6Eu 15156063244230469   image/jpeg  0
f20180111014341/j_ROHpl4EG.json 1036    ForGzwzV3e-uR3_UzvppJs1VgfQG 15156064773253144   application/json    0
f20180111020739/i_OTDKsOrD.jpeg 169472  FrITJxleSP7wUD-MWw-phL_KP6Eu 15156076847077556   image/jpeg  0
f20180111020739/j_JFO6xiir.json 1040    FmUhTchdLOd7LBoE8OXzPLDKcW6O 15156077904192983   application/json    0
f20180111090619/i_1BwNksbL.jpg  49634   FtXBGmipcDha-67WQgGQR5shEBu2 15156329458714950   image/jpeg  0
f20180111090619/i_3BKlsRaZ.jpg  30152   FoWfMSuqz4TEQl5FT-FY5wqu5NGf 15156330575626044   image/jpeg  0
f20180111090619/i_5XboXSKh.jpg  40238   Fl84WaBWThHovIBsQaNFoIaPZcWh 15156329453409855   image/jpeg  0
f20180111090619/i_6DiYSBKp.jpg  74017   FrYG3icChRmFGnWQK6rYxa88KuQI 15156329461803290   image/jpeg  0
f20180111090619/i_76zaF2IM.jpg  38437   Fui8g5OrJhOGQqZzT9wtepfq99lJ 15156334738356648   image/jpeg  0
```

```
f20180111090619/i_B6TFYjks.jpg  37953   FleWqlK2W1ZmEgAatAEcm1gpR0kC
15156329464034474   image/jpeg  0
f20180111090619/i_N9eITqj3.jpg  38437   Fui8g5OrJh0GQqZzT9wtepfq99lJ
15156330419595764   image/jpeg  0
f20180111090619/i_QTSNWmA6.jpg  37953   FleWqlK2W1ZmEgAatAEcm1gpR0kC
15156333104224056   image/jpeg  0
f20180111090619/i_XdHcAfh1.jpg  56479   FjLQIQ3GxSEHDfu6tRcMylK1MZ05
15156334227270309   image/jpeg  0
f20180111090619/i_Xyy723MU.jpg  50076   FsfZpQzqu084RUw5NPYW9-Yfam_R
15156334229987458   image/jpeg  0
f20180111090619/i_d8GoOEOv.jpg  30152   FoWfMSuqz4TEQl5FT-FY5wqu5NGf
15156334736228515   image/jpeg  0
f20180111090619/i_diuHmX53.jpg  40591   FuTx1pw4idbKnV5MSvNGxCA5L470
15156333878320713   image/jpeg  0
f20180111090619/i_qQKzheSH.jpg  55858   FjOA3i8V7fzzOiPQFL79ao15hkN9
15156329456666591   image/jpeg  0
f20180111090619/i_rHL5SYk8.jpg  40238   Fl84WaBWThHovIBsQaNFoIaPZcWh
15156336509742181   image/jpeg  0
f20180111090619/i_xZmQxUbz.jpg  40238   Fl84WaBWThHovIBsQaNFoIaPZcWh
15156333240603466   image/jpeg  0
f20180111090619/i_zBDNgXDv.jpeg 73616   FlgNwq8lypgsxrWs_ksrS_x47SQV
15156334232887875   image/jpeg  0
f20180111090619/j_4mxbEiVh.json 2990    Fpq-3yl3Yr1CadNrJVSDnpeRhQtT
15156331445226898   application/json    0
f20180111090619/j_i1K74768.json 3042    Fl5PpDw1TsZXMuhoq1RUrOeGZ6br
15156335067090003   application/json    0
f20180111095839/i_Q7KMKeda.png  518522  Fl-yB1_ruL2uxZN9k7DjB62h9dYH
15156359599713253   image/png   0
f20180111095839/j_5DpqHolV.json 184 FoYvi7cmSrzuVjUgCRzW5kU95SVo
15156359719719064   application/json    0
f20180111100442/i_No8kToIV.jpg  48975   Fu1cw3f--5Vpz9kLGeJfvljhCtyZ
15156364349642377   image/jpeg  0
f20180111100442/i_P1bkvSeg.jpg  68200   FvYe8vi46TjUKhEy_UwDqLhO6ZsW
15156363800690634   image/jpeg  0
f20180111100442/i_T1AulkcD.jpg  52641   Fj2YzvdC1n_1sF93ZZgrhF3OzOeY
15156364021186365   image/jpeg  0
f20180111100442/i_X8d8BNO7.jpg  50770   FivwidMiHbogw77lqgkIKrgmF3eA
15156363969737156   image/jpeg  0
f20180111100442/i_gOwtOsCX.jpg  76656   FmtixxOmP9CAUTNosjLuYQHL6kOP
15156363448222155   image/jpeg  0
f20180111100442/i_h5OT9324.jpg  72672   FvbIqPLTh2cQHTIBv2akUfahZa_Z
15156364401354652   image/jpeg  0
f20180111100442/i_he8iLYI6.jpg  49399   FjeJvwjwhU-hKZsq66UoBg9_tEJs
15156363907932480   image/jpeg  0
f20180111100442/i_kg29t7Pp.jpg  76293   FuYj__sSeEN7AsXMbxO24Z8Suh8d
15156364156384686   image/jpeg  0
f20180111100442/i_oz1YoBI1.jpg  75620   FkY3xsUMwOI01zgoH1iXXgiQeq6I
15156364089112904   image/jpeg  0
f20180111100442/i_xrOT98on.jpg  50021   Fql7ookM1Rc6V7VairKAfnKe-o9w
15156363856357316   image/jpeg  0
f20180111135114/i_Zqt8Tmoe.png  161629  FlELw59_mV3VqDBLyu1BKN4fIWnx
15156500155209863   image/png   0
f20180111135114/j_uhHoMXKq.json 159 FrypljwAr2LgoLAePBNTUYTUAgDt
15156500200488238   application/json    0
f20180111142119/i_s83iZ2GR.png  92278   Fns8tdh3JCkRmfE_COYEu4o8w03E
15156517082371259   image/png   0
f20180111142119/j_Og45JRth.json 159 Fq1rFwdRguYRXrp61nGZ5TsUG1V-
15156517143375596   application/json    0
```

```
f20180111144306/i_yE5TC84E.png  139230  Fjf61ymabEnEvnr5ZMHFjXGCrYlP  15156530038824150   image/png   0
f20180111144306/j_OF4WVtSH.json 159 FqwkKcxfo8jdOjFUyuH4X2CrnE9q  15156530083419530   application/json    0
f20180111150230/i_KtnER4g3.png  120044  FuwOWdrqzcr2-UScem-LzEMgMezs  15156541734892258   image/png   0
f20180111150230/j_xMSUEejY.json 158 FjJr_4deMqFphGaptm-2Pa6wwRP2  15156541771989216   application/json    0
f20180111151741/i_JuSWztB3.jpg  92506   FrIjRevHSi6xv4-NQa2wrHu5a1zQ  15156550875370965   image/jpeg  0
f20180111153550/i_9wWzVenl.gif  769872  FvslKY9JUaCQm-lu02E34tvAP_oG  15156561674621628   image/gif   0
'''

#第一步,将文件类型提取出来
#1按行分割每一行,2以\t制表符作为分割符,取到第一段的值,3然后以.号作为分割符,取到列表的第二位,也就是文件类型
#第二步,将文件大小提取出来
#第三步,将相同类型的文件大小进行累加,不同类型的分开放

# dict1={}
# list1=log.split('\n')  #分割每一行
# for one in list1:  #遍历这个列表
#     if one !='':  #只处理非空行
#         one1=one.split('\t')[0]  #以\t为分割符进行切割,然后取得第0位
#         file_size = int(one.split('\t')[1])  #提取出文件大小,并转为int型
#         file_type=one1.split('.')[1]  #以.号为分割符再次进行切割,然后取得第1位
#         if file_type not in dict1:
#             dict1[file_type]=file_size
#         else:
#             dict1[file_type]+=file_size
# print(dict1)
# list2=[]
# def fun1(file_type,file_size):
#     for one in list2:
#         if one[0]==file_type:  #如果子列表中已经存在这个数据类型
#             one[1]+=file_size  #进行文件大小的累加
#             break
#     else:
#         list2.append([file_type,file_size])  #向list1中添加一个子列表,里面存放文件类型,文件大小
# # [[文件类型,文件大小],[文件类型,文件大小]]
#
# for one in list1:  #遍历这个列表
#     if one !='':  #只处理非空行
#         one1=one.split('\t')[0]  #以\t为分割符进行切割,然后取得第0位
#         file_size = int(one.split('\t')[1])  #提取出文件大小,并转为int型
#         file_type=one1.split('.')[1]  #以.号为分割符再次进行切割,然后取得第1位
#         fun1(file_type,file_size)
# print(list2)
```