

檢查並統計電廠的下載失敗紀錄

平常不定時檢查下載失敗的紀錄

檢查 11/26 抓取失敗的紀錄 (初次下載時)

OpenSSH SSH client

```
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606140002/output.log:364:抓取失敗: 興達 A2 2020/11/23 22:08:35
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606143602/output.log:191:抓取失敗: 南部 A3 2020/11/23 23:04:25
Administrator@WTGroup:~$ ~/bash_scripts/chk_windy_failure.sh | grep '11/26'
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:40:抓取失敗: 協和 B 2020/11/26 00:01:12
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:45:抓取失敗(2): 協和 B 2020/11/26 00:01:34
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:74:抓取失敗: 協和馬祖珠山 A2 2020/11/26 00:02:18
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:79:抓取失敗(2): 協和馬祖珠山 A2 2020/11/26 00:02:40
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:117:抓取失敗: 林口 A1 2020/11/26 00:03:38
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:122:抓取失敗(2): 林口 A1 2020/11/26 00:04:00
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:127:抓取失敗: 林口 A2 2020/11/26 00:04:16
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:176:抓取失敗: 南部 A1 2020/11/26 00:05:34
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:181:抓取失敗(2): 南部 A1 2020/11/26 00:05:55
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:206:抓取失敗: 南部 B 2020/11/26 00:06:40
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:235:抓取失敗: 通霄 A1 2020/11/26 00:07:30
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:240:抓取失敗(2): 通霄 A1 2020/11/26 00:07:51
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:255:抓取失敗: 通霄 A3 2020/11/26 00:08:22
Administrator@WTGroup:~$ ~/bash_scripts/chk_windy_failure.sh | grep '11/26' | grep '(2)' 11/26 當天最終丟失紀錄 (以下 5 筆)
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:45:抓取失敗(2): 協和 B 2020/11/26 00:01:34
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:79:抓取失敗(2): 協和馬祖珠山 A2 2020/11/26 00:02:40
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:122:抓取失敗(2): 林口 A1 2020/11/26 00:04:00
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:181:抓取失敗(2): 南部 A1 2020/11/26 00:05:55
/volumes/WTGroup_Data/tmp_python/synoscheduler/7/1606320002/output.log:240:抓取失敗(2): 通霄 A1 2020/11/26 00:07:51
Administrator@WTGroup:~$ ~/bash_scripts/chk_windy_failure.sh | grep '11/26' | wc -l
13
Administrator@WTGroup:~$
```

11/26 當天第一次抓資料時，共有 13 筆擷取失敗



Grouped bar chart with labels

This example shows a how to create a grouped bar chart and how to annotate bars with labels.

```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

labels = ['G1', 'G2', 'G3', 'G4', 'G5']
men_means = [20, 34, 30, 35, 27]
women_means = [25, 32, 34, 20, 25]

x = np.arange(len(labels)) # the label locations
width = 0.35 # the width of the bars

fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, men_means, width, label='Men')
rects2 = ax.bar(x + width/2, women_means, width, label='Women')

# Add some text for labels, title and custom x-axis tick labels, etc.
ax.set_ylabel('Scores')
ax.set_title('Scores by group and gender')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()
```

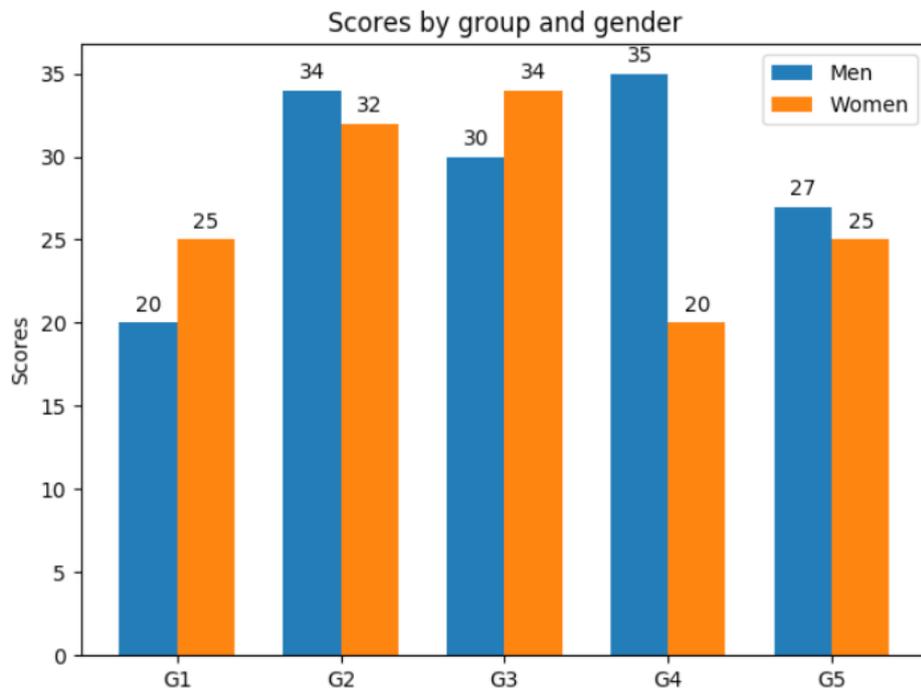
google 大神找到的範例
(matplotlib 的說明文件)

```
def autolabel(rects):
    """Attach a text label above each bar in *rects*, displaying its height."""
    for rect in rects:
        height = rect.get_height()
        ax.annotate('{}'.format(height),
                    xy=(rect.get_x() + rect.get_width() / 2, height),
                    xytext=(0, 3), # 3 points vertical offset
                    textcoords="offset points",
                    ha='center', va='bottom')

autolabel(rects1)
autolabel(rects2)

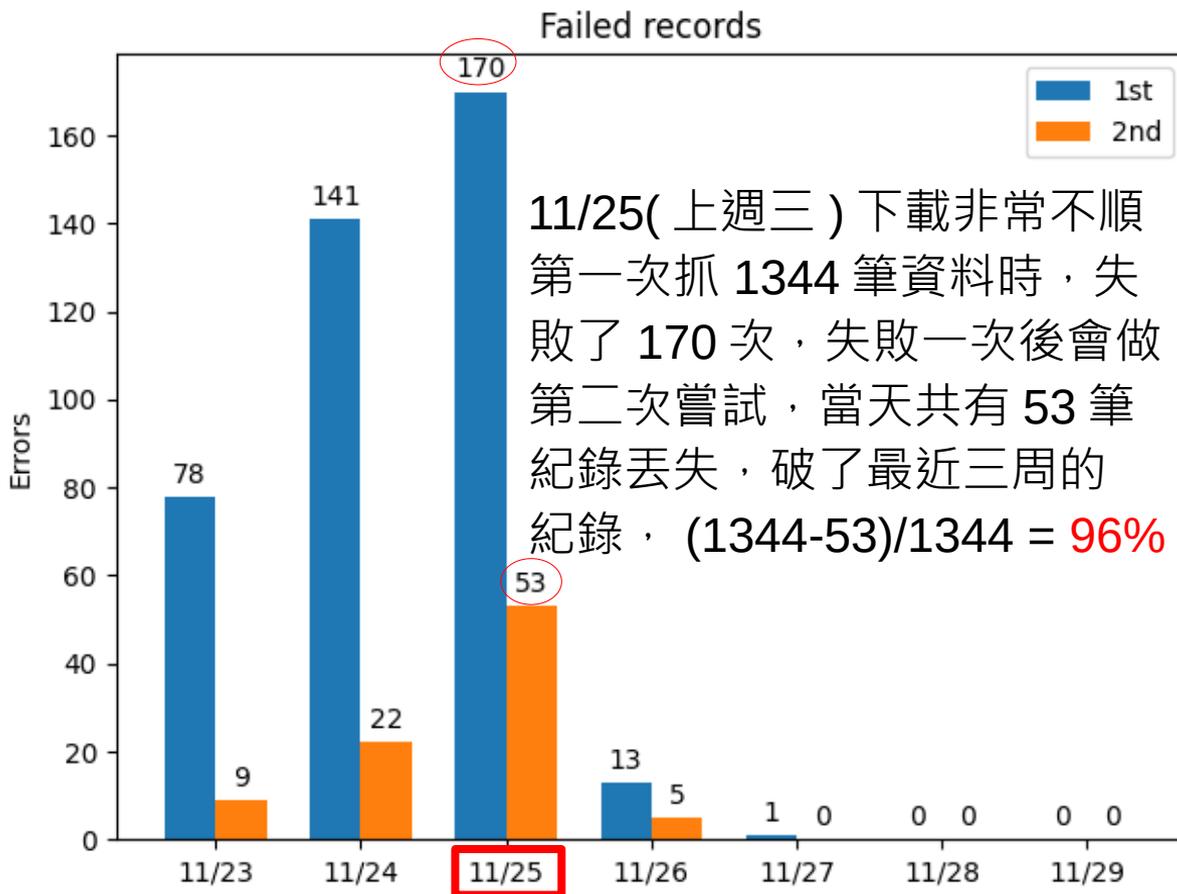
fig.tight_layout()

plt.show()
```



粗略統計下載失敗 (用 python 實現)

56 處 gps 左標
X
24 一天
=
1344 筆紀錄



```

1 import matplotlib
2 import matplotlib.pyplot as plt
3 import numpy as np
4
5
6 ...
7 days_name = ['11/23',
8             '11/24',
9             '11/25',
10            '11/26',
11            '11/27',
12            '11/28',
13            '11/29']
14
15 #'''
16 days_name = ['11/16',
17             '11/17',
18             '11/18',
19             '11/19',
20             '11/20',
21             '11/21',
22             '11/22',
23            '11/23',
24            '11/24',
25            '11/25',
26            '11/26',
27            '11/27',
28            '11/28',
29            '11/29']
30 #'''

```

一周

兩周

一旦程式寫好，平常我只要修改存放日期的表格，就可以看到結果了



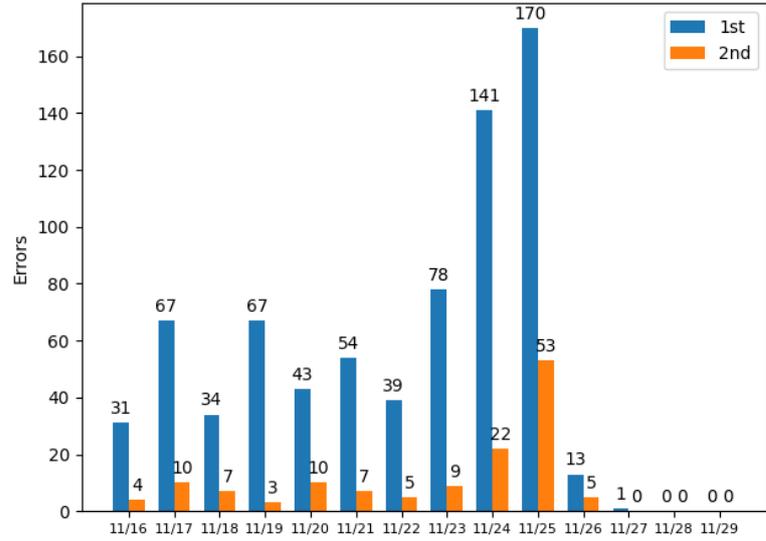
```

31 days_failed = [[] for i in days_name]
32
33 with open('windy powerplant 20201130.txt', mode='rt', encoding='utf8') as f:
34     for line in f.readlines():
35         for n, day in enumerate(days_name):
36             if day in line:
37                 days_failed[n].append(line)
38
39 labels = days_name
40 errors_1 = [len(day) for day in days_failed]
41
42 errors_2 = []
43 for day in days_failed:
44     count = 0
45     for error in day:
46         if '(2)' in error:
47             count = count + 1
48     errors_2.append(count)
49 #quit()
50
51 #labels = ['G1', 'G2', 'G3', 'G4', 'G5']
52 #errors_1 = [20, 34, 30, 35, 27]
53 #errors_2 = [25, 32, 34, 20, 25]
54
55 x = np.arange(len(labels)) # the label locations
56 width = 0.35 # the width of the bars

```

準備我的資料，取代這三行

Failed records



```

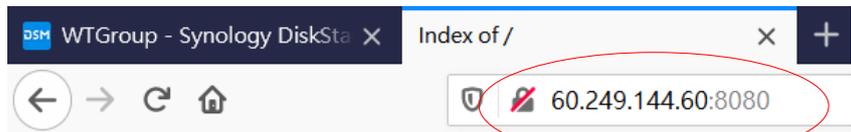
57 fig, ax = plt.subplots()
58 rects1 = ax.bar(x - width/2, errors_1, width, label='1st')
59 rects2 = ax.bar(x + width/2, errors_2, width, label='2nd')
60
61 # Add some text for labels, title and custom x-axis tick labels, etc.
62 ax.set_ylabel('Errors')
63 ax.set_title('Failed records')
64 ax.set_xticks(x)
65 ax.set_xticklabels(labels)
66 ax.legend()
67 plt.xticks(fontsize=8) # add for 2-weeks record
68
69
70 def autolabel(rects):
71     """Attach a text label above each bar in *rects*, displaying its height."""
72     for rect in rects:
73         height = rect.get_height()
74         ax.annotate('{}'.format(height),
75                 xy=(rect.get_x() + rect.get_width() / 2, height),
76                 xytext=(0, 3), # 3 points vertical offset
77                 textcoords="offset points",
78                 ha='center', va='bottom')
79
80
81 autolabel(rects1)
82 autolabel(rects2)
83
84 fig.tight_layout()
85
86 fig.savefig('test.png') # add
87
88 plt.show()

```

改成兩周後資料較密集，將 x 軸的字改小一點

另外存成圖檔

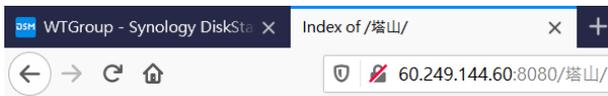
查看下載好的資料



Index of /

(1)

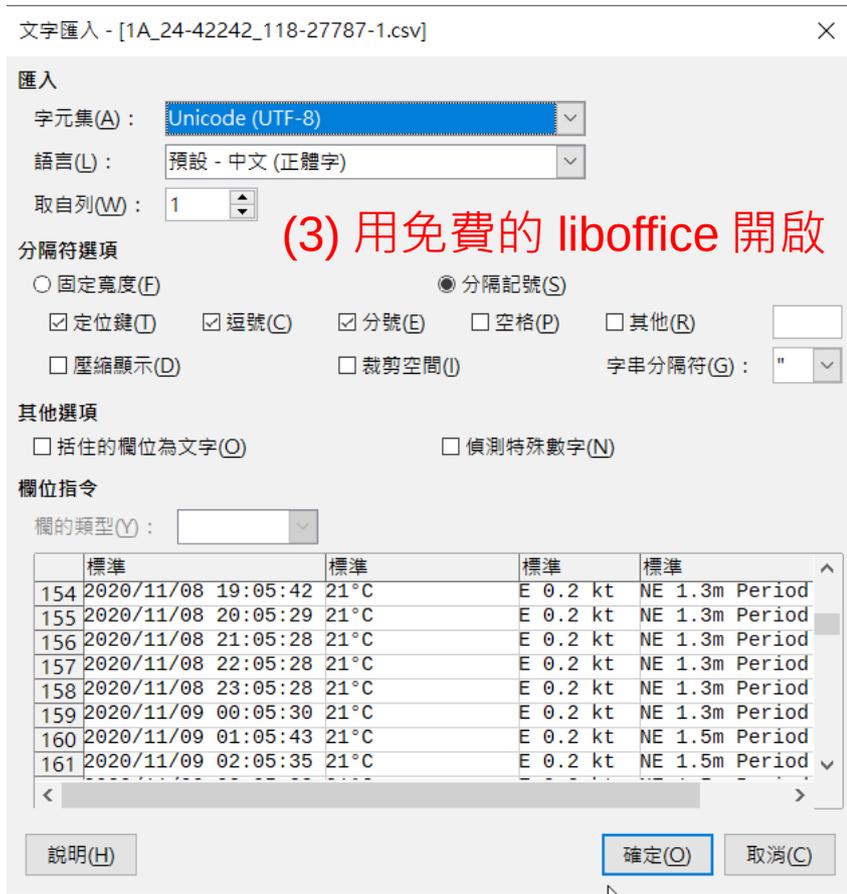
- (drwxrwxrwx) 協和/
- (drwxrwxrwx) 協和馬祖珠山/
- (drwxrwxrwx) 南部/
- (drwxrwxrwx) 台中/
- (drwxrwxrwx) 塔山/
- (drwxrwxrwx) 大林/
- (drwxrwxrwx) 大潭/
- (drwxrwxrwx) 尖山/
- (drwxrwxrwx) 林口/
- (drwxrwxrwx) 興達/
- (drwxrwxrwx) 通霄/
- (-rw-----) 32.1k 20201119_0859.txt
- (-rw-----) 233.0k nohup.out



Index of /塔山/

(2)

- (drwxr-xr-x) ./
- (-rwxrwxrwx) 38.8k 1A_24-42242_118-27787.csv
- (-rwxrwxrwx) 38.9k 1B_24-41955_118-28245.csv
- (-rwxrwxrwx) 38.9k 2A_24-4131_118-2714.csv
- (-rwxrwxrwx) 38.9k 2B_24-409767_118-276883.csv
- (-rwxrwxrwx) 39.0k 背景站_24-423286_118-267192.csv



(3) 用免費的 liboffice 開啟