Check my RTTY Skimmer Spots

I saved my Rtty Skimmer spots, and copied them to my data analysis machine.

There is nothing special or difficult about this task.

My Skimmer was initially set to minimal, but after a few hours I moved it to normal.

```python
with open('./Wae Rtty/WAERTTYSpots.txt') as spotfile:
    myspots=spotfile.read().split('\n')
```

I would now like to check we have the data correctly.

```python
# Just check we are looking ok.... And we are
for n in range (0,3):
    print(myspots[n])
```

And I see

```
2021-11-12 02:13:55Z    14091.5  HL2DCM        12-Nov-2021 0213Z     9 dB  45 BPS  DE RTTY          <N0N
2021-11-12 02:14:09Z    14091.5  HL2DCM        12-Nov-2021 0214Z     9 dB  45 BPS  CQ RTTY          <N0N
2021-11-12 05:26:28Z    28081.8  DU3TW         12-Nov-2021 0526Z    15 dB  45 BPS  CQ RTTY          <N0N
```

Check Call and Operation

```python
# Lets grab 2 items  - who I think their call is ... and the DE/CQ field
# I am assuming that this is a fix format output
for n in range (0,3):
    call =myspots[n][32:44]
    doing=myspots[n][79:81]
    print(f"<{call}> <{doing}>")
```

Again that looks reasonable (Ignore the spaces.. I will strip them out before using this data).

```
<HL2DCM       > <DE>
<HL2DCM       > <CQ>
<DU3TW        > <CQ>
```

Time for a list of all callsigns my skimmer spotted

Lets extract all the calls - and make them unique

```
calls=[]
for s in myspots:
    call =s[32:44].strip()
    if (len(call)):
        calls.append(call)
calls=set(calls)
print(f"My Skimmer spotted {len(calls)} unique called in WAE Rtty")
```

And this outputs

```
My Skimmer spotted 283 unique called in WAE Rtty
```

So I have no excuse about not working 283 stations .... Oh dear.... ha ha

SCP Time - data

With the skimmer data looking useable, it is time to turn out attention to SCP.

This file was the latest, and was obtained from http://www.supercheckpartial.com/MASTER.SCP I will open it and strip all comments out ... (apologies Stu Phillips, K6TU)

```
with open('Wae Rtty/MASTER.SCP') as scp_file:
    scp=[a for a in scp_file.read().split('\n') if not a.startswith('#')]
```

Yes it is really that easy to read in.... It is just a big list.

But the big question is .... **AM I IN IT ?**

Lets check

```
'DU3TW' in scp
```

And it returns

```
True
```

Phew, I breath a sigh of relief.

Checking My Skimmer against SCP

I hope this part has not been too drawn out...

```
good=0
unmatched=0
ucalls=[]
for c in calls:
    if c in scp:
        good +=1
    else:
        unmatched +=1
        ucalls.append(c)

ucalls.sort()
print("These are the unmatched calls ")
for uc in ucalls:
    print(f"{uc}")
print(f"Overall match rate is {100.0*good/len(calls)}")
```

And I see

```
These are the unmatched calls
8N1OME/1
9M2U
9W2FHG
AA0ZK
B0RI
BD3GNI
BG4QNE
DG5RC
DK2O
DR5Z
DV9I
DV9IH
EA5U
G5LG
HS6VW
IK4CLF
IK7GRU
JA1AY
JA3XJF
JE7JZT
JI3CWI
JN1VXL
JR1NH
JS6UEY
KE1AC
LG5LG
N6HPX
PA1C
S53P
SP4NK
YB1BRS
YB3EDD
YB9G
YC0SCL
YC4SIZ
YC9AOS
YD1EQX
YD3ASV
YD3TSJ
YE8DWC
YE8RA
YG8RAG
Overall match rate is 85.15901060070671
```

85% match ... That's pretty good I would say.

QRZ ?

Sadly I do not have a XML query account on QRZ ... so I can not automate a lookup... But of the calls I picked ... in this "un-matched" list... Most seems to have QRZ pages.

I can see two locally busted calls

- DV9I
- DV9IH

This looks unlikely

- B0RI
- G5LG (maybe a G-Station contest call however)