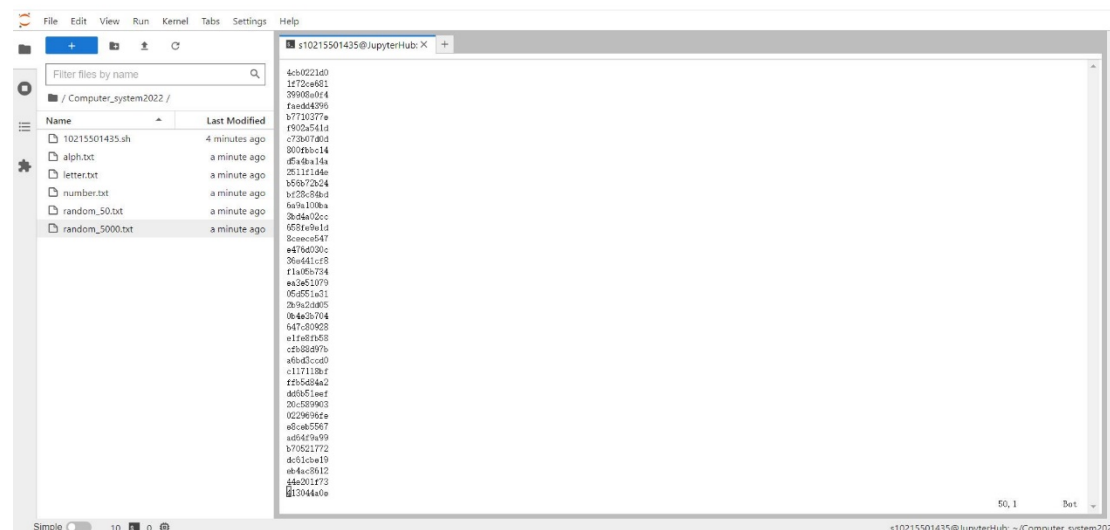


Shell 脚本：



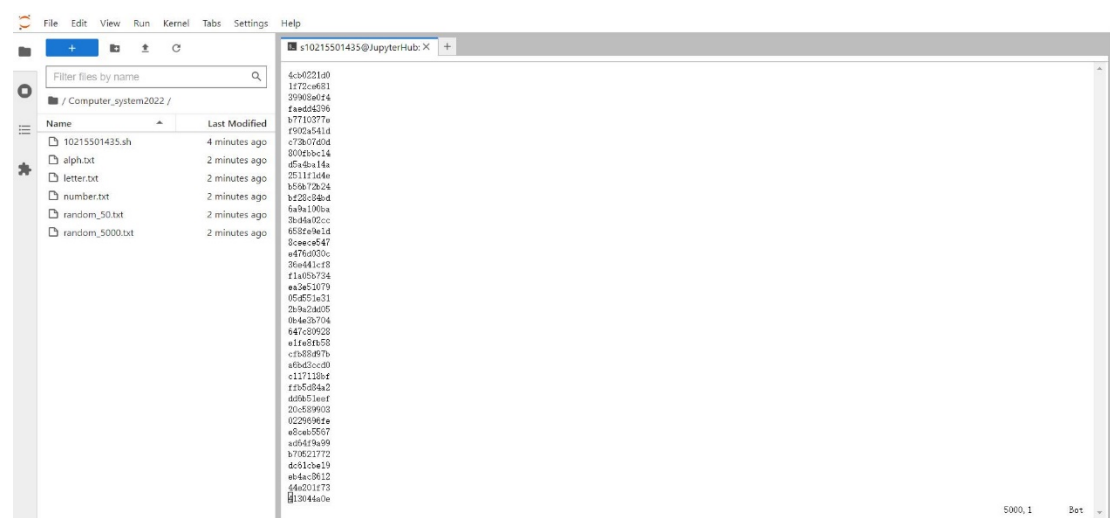
```
#!/bin/bash
#生成50行随机的字符串
for i in {1..50}
do
    echo $RANDOM | md5sum | cut -c 1-9 >> random_50.txt
done
done
#将上述50行随机的字符串循环100次，得到5000行有规律的字符串
times=1
while((times<=100))
do
    head -n 50 random_50.txt >> random_5000.txt
    times=$((times + 1))
done
#以覆盖写的方式，字母排序，输出unique行
sort -u random_5000.txt > alph.txt
#以覆盖写的方式，数字排序，输出unique行
sort -n random_5000.txt > number.txt
#以追加写的方式，字母排序，输出unique行
sort -u random_5000.txt >> alph.txt
#以追加写的方式，数字排序，输出unique行
sort -n random_5000.txt >> number.txt
#已经生成一个含有60个Computer的letter文件
crp -c "Computer" letter.txt
```

第一句先生成 50 行随机的字符串，结果如下：



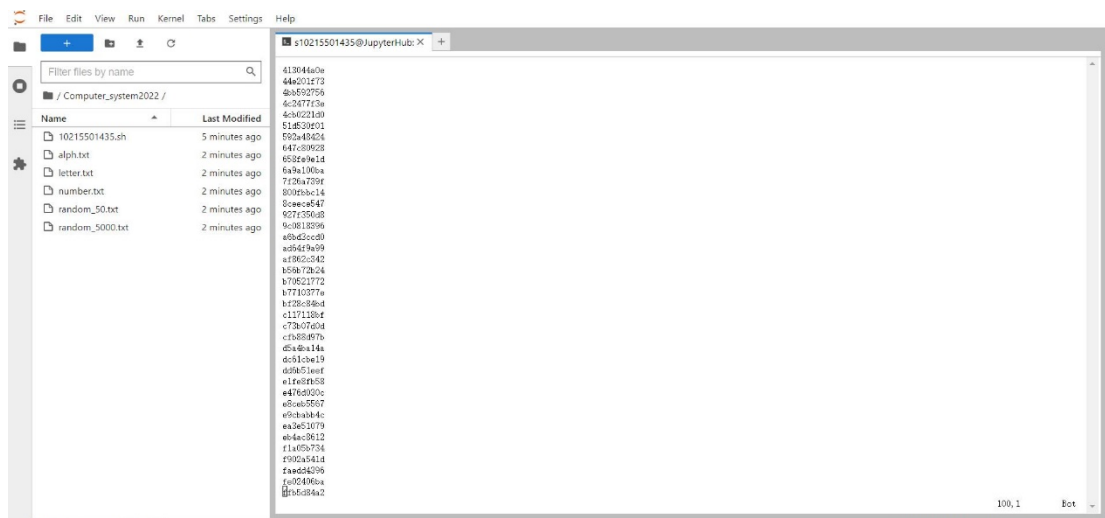
```
4e30221a9
1f72e681
39908a04
faed4596
b7710377e
f902a541d
c7207d0d
8079bc14
d549a14a
2511f164e
b59b7b24
b22c59d4
6a9a100ba
3b44a0cc
658f6b1d
8c9ec6547
e476d030c
36a41c18
f1a05b734
ea3e51079
05d51e31
2b9c2d805
0b4e3b704
647c89520
e1f68f658
c7b88497b
a6bd30cd9
c117118bf
f6b4544a2
d09651eef
20c589902
0229696fa
e8cab5567
a0d4f9a99
b70521772
dc81cbe19
eb4ac9612
44a201f73
11304460e
```

第二步：把上述过程循环执行 100 次，生成有 5000 行的字符串

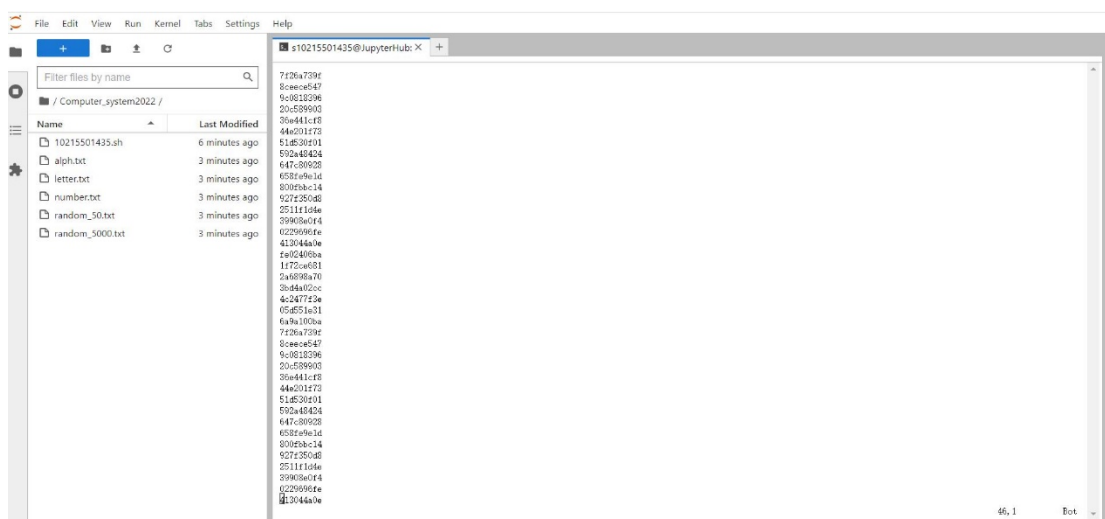


```
4e30221a9
1f72e681
39908a04
faed4596
b7710377e
f902a541d
c7207d0d
8079bc14
d549a14a
2511f164e
b59b7b24
b22c59d4
6a9a100ba
3b44a0cc
658f6b1d
8c9ec6547
e476d030c
36a41c18
f1a05b734
ea3e51079
05d51e31
2b9c2d805
0b4e3b704
647c89520
e1f68f658
c7b88497b
a6bd30cd9
c117118bf
f6b4544a2
d09651eef
20c589902
0229696fa
e8cab5567
a0d4f9a99
b70521772
dc81cbe19
eb4ac9612
44a201f73
11304460e
```

按字母排序，输出 unique 行，在同一个文件里先覆盖写再追加写，所以最终的结果有 100 行（其中有 50 行是相等的）



按数字排序，在同一个文件里先覆盖写再追加写，最终结果有 46 行，值得注意的是其中有 23 行是相等的，也印证先覆盖写再追加写的结果是正确的。



最后我生成了一个含有 Computer 的文本，并将这段文本循环了六十次，最后查找，结果是

```
s10215501435@JupyterHub: ~/Computer_system2022$ bash 10215501435.sh
60
```

60 次，结果正确。