pwd	Where are you ?
ls -la	show list files, directories
cd	change directory
cd ~	go to home directory
cd ..	go to up one level
cd ./	this directory
realpath /d1/d2/f	create right path

read, write, eXecute

cat fileName	read small file (to 1M)
cat file1 file2 > file3	create file3, add file1 and file2
more fileName	read file for part
tail fileName	read 10 last rows
tail –n 100 fileName	read 100 last rows
tail –f fileName	read real time
mkdir dirName	create directory
touch fileName	create file or edit file date
rmdir dirName	delete empty directory
rm -r dirName	delete directory
rm fileName	delete file
cp file1 file2	copy file1 to file2
cp –R dir1 dir2	copy directory1 to directory2
mv file1 file2	rename file1 to file2
mc	open total commander
mcedit fileName	edit file
nano fileName	edit file
vi fileName	edit file

df -h	show space on the disc
du -h	show files` size

stdin(0), stdout(1), stderr(2)

command > fileName	record to file with overwrite
command >> fileName	record to file with add to tail
command 2> fileName	record err to file

grep

ls –l	grep "text"	search "text" in ls -l result
grep –R "date" ./	search "date" in all files and directories
"[^0-9]+" ""	all !=number to null

rwx rwx rwx	chmod g+w fileName
user group other	chmod u+x fileName

script bash

which bash
mcedit script.sh
#!/usr/bin/env bash
//script text

NOW=$(date)
echo $NOW

```bash
if [ "$1" != "" ]
then
    echo "yes"
else
    echo "no"
fi

if [ "$a" = "$b" ]
if [ "$a" == "$b" ]
if [ "$a" != "$b" ]

if [ "$a" -eq "$b" ] equal number
if [ "$a" -ne "$b" ] not equal number
if [ "$a" -gt "$b" ] is greater number
if [ "$a" -ge "$b" ] is greater or equal number
if [ "$a" -lt "$b" ] is less number
if [ "$a" -le "$b" ] is less or equal number
```

```bash
DIRECTORIES="/test1 /test2 /test3"
for DIRNAME in $DIRECTORIES
do
    echo "Directory: $DIRNAME"
done

for i in {1..20}
do
    echo $i
done
```

```
cat service.2020-12-01.log | grep 'FILTER START' | awk '{print $9}' | sort | uniq -c | sort -nr  >
~/sp1201_public_api.txt
```

files to zip = tar -czvf fileName.gz fileList

files from zip = tar -xzvf fileName.gz

create a file

#!/bin/bush

_____

chmod +x log.sh

./log.sh

----

wc filleName => count lines, words, letters

```
cat service.2020-12-01.log | grep 'FILTER START' | awk '{print $9}' | sort -u > ~/sp_public_api.txt
```

```
cat service.2020-12-01.log | grep 'FILTER START' | awk '{print $9}' | sort | uniq -c | sort -nr  >
~/sp1201_public_api.txt
```

```
curl https://dictionary.openbudget.gov.ua/items/kfk | awk 'BEGIN{RS=","}{print}' | grep 'beginDoc' | sort |
uniq –c
```

```
cat dpsTax.txt | awk 'BEGIN{RS="},"}{print}' | grep '{' -c
```

рейтинг АЦСК

```
cat service.2020-12-* | grep 'checking' | awk '{print $8}' | sort | uniq -c | sort -nr >
~/sign_2020_12_all_verify.txt
```

Команда sort INPUTFILE | uniq -c | sort -nr выводит статистику встречаемости строк в файле INPUTFILE (ключ -nr, в команде sort, означает сортировку в порядке убывания). Этот шаблон может с успехом использоваться при анализе файлов системного журнала, словарей и везде, где необходимо проанализировать лексическую структуру документа.

Команды обработки текста
https://www.opennet.ru/docs/RUS/bash_scripting_guide/x7050.html

Регулярные выражения
http://mit.spbau.ru/files/regexp.pdf

Утилиты обработки текста в ОС Linux
http://library.voenmeh.ru/cnau/WPSkNoRefFySfVw.pdf

grep -v 'text' fileName = include text

ЧЕК-ЛИСТ
1. Количество файлов в периоде
2. Количество ошибок в периоде

-B 3 >> before 3 lines
-A 1 >> after 1 line
wc -l >> count lines

```
grep -B 5 -A 2 --color 'keyword' /path/to/file.log
cat service.2022-02-10.log | grep -B 3 -A 1 -i error
cat service.2022-02-10.log | grep -i ERROR | wc -l
zgrep -a -i error service.2022-02-03.log.gz
```

```
ls service.2022-02* | wc -l
```


***
RUN
```
chmod +x acskList.sh | sh acskList.sh
```


```
#!/bin/bash
cd ~/
cd /opt/verify-signature-service/
cat service.2022-02-*.log | grep 'checking' | awk '{print $8}' | sort | uniq -c | sort -nr > ~/acsk_list.txt
```


```
#!/bin/bash
cd ~/
cd /opt/ob-logs/ob-public-api/
for i in *2020-{0..12}*;do echo $i;done
```