ls -a -l -t -h

all files

long listing

order by date of last alteration

human-readable Filesizes

ls -alth

ls -l

grep "abc" targetfile

grep ---file= patterns.txt targetfile

File containing patterns to search (one pattern by line)

targetfile

File in which to search

short form:

grep -f patterns.txt targetfile

Common "pure text" editors:

vim

emacs

nano

gedit

~~Libreoffice~~

~~MS Word~~

echo *

echo paul

file[12][0-9]

↳ picks everything from file10 to file29

Bash expansions

file[0-9]

↳ picks everything from file0 to file9 (incl. file1, file2, ...)

In Bash (NOT in regexps):

file[12]*

picks all files starting with "file1" or "file2"

careful with redirections:

$ echo "hello" > outfile
↳ erases the previous content of outfile, if it existed

$ echo "hello" >> outfile
↳ appends instead of erasing

$ grep > file.fasta ⚠2

↳ 1. empty file.fasta
2. print error message "Usage: ..."

correct would be:
grep '>' file.fasta

$ ztb-+, > outfile

$ ztb-+,

first thing done here is to prepare the standard output using the redirection, erasing any existing content

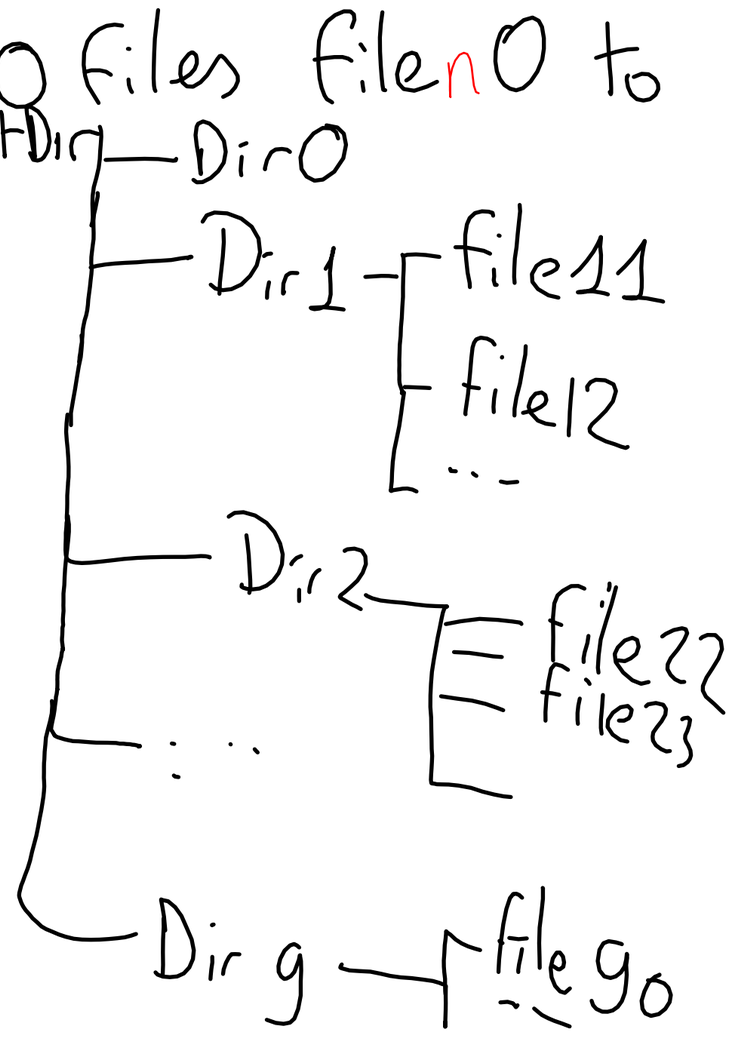
FOR LOOPS

```
for num in 1 2 3 4 5 6
do
    touch file${num}.txt
done
```

OR

```
for num in {1..6}; do touch file${num}.txt; done
```

1. create the TestDir directory
2. Inside TestDir, create Dir0 to Dir9 directories
3. Inside each Dirn directory, create 10 files filen0 to filen9
4. Combine all commands in one script that erases everything first, before rebuilding the whole TestDir data structure



```
for num in {0..9}
do
  for z in {10..19}
    do
       touch file${num}
  done
done
```

inner loop is just a dummy repetition of 10 times the same

```
rm -rf TestDir # very dangerous recursive deletion
for major in {0..9}
do
   mkdir -p TestDir/Dir${major}
   for minor in {0..9}
   do
      touch TestDir/Dir${major}/file${major}${minor}
   done
done
```

① exercise: split the contents of the file Spreadsheet.csv into as many chunks as necessary (chunk1.csv, chunk2.csv, etc) so that each one contains 10 rows from the input file

② Do same but in addition, preserve the header line in each of the output chunks.