

Dersim Besiktas  
TP10 : Les commandes – Les principaux caractères du shell

---

**SOMMAIRE**

1. Les redirections (>, >>, 2>, 2>>, 2>&1 et  ).....	2
2. Utilisation des jokers.....	16
3. Utilisation des caractères de protection.....	18
4. Autres caractères.....	19

## 1. Les redirections (>, >>, 2>, 2>>, 2>&1 et |)

Je redirige le résultat de la commande **cal** dans un fichier qui se nomme cal.txt  
Puis j'affiche son contenu avec la commande **cat**

```
guest@DEB13Server:~$ cal > cal.txt
guest@DEB13Server:~$ cat cal.txt
  Décembre 2025
di lu ma me je ve sa
   1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

guest@DEB13Server:~$ _
```

Je redirige le résultat de la commande **date** dans le même fichier et l'ancien contenu est écrasé

```
guest@DEB13Server:~$ date > cal.txt
guest@DEB13Server:~$ cat cal.txt
mer. 17 déc. 2025 10:12:58 CET
guest@DEB13Server:~$ _
```

Je redirige le résultat de la commande **cal** dans un fichier qui se nomme histo.txt et à l'aide des deux chevrons >>, j'ajoute à la suite du contenu du fichier le résultat de la commande **date**

```
guest@DEB13Server:~$ cal > histo.txt
guest@DEB13Server:~$ date >> histo.txt
guest@DEB13Server:~$ more histo.txt
  Décembre 2025
di lu ma me je ve sa
   1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

mer. 17 déc. 2025 10:14:29 CET
guest@DEB13Server:~$
```

En mettant un chevron > j'écrase le contenu existant du fichier **histo.txt** et lorsque j'affiche le contenu du fichier avec la commande **cat** il n'y a plus rien

```
guest@DEB13Server:~$ > histo.txt
guest@DEB13Server:~$ cat histo.txt
guest@DEB13Server:~$ _
```

Je redirige les valeurs du premiers champs du fichier **notes.csv** dans un nouveau fichier nommé **elevés.txt**

```
root@DEB13Server:~$ cut -d , -f 1 notes.csv > elevés.txt
root@DEB13Server:~$ ls -l
total 16
-rw-r--r-- 1 root root 79 17 déc. 11:03 elevés.txt
-rw-r--r-- 1 root root 74 10 déc. 17:44 étudiants.txt
-rw-r--r-- 1 root root 218 10 déc. 18:06 notes.csv
-rw-r--r-- 1 root root 80 10 déc. 18:07 prenom_tries
root@DEB13Server:~$ cat elevés.txt
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier

root@DEB13Server:~#
```

Je refais la même opération en mettant un deuxième chevron pour ne pas écraser le contenu du fichier, puis j'affiche le contenu avec la commande **ls**

```
root@DEB13Server: ~# cut -d , -f 1 notes.csv >> eleves.txt
root@DEB13Server: ~# cat eleves.txt
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier

Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier

root@DEB13Server: ~#_
```

La commande **sort** permet de trier le contenu du fichier et je redirige le résultat vers un nouveau fichier qui se nomme **avecdoublons.txt**

```
root@DEB13Server: ~# sort eleves.txt > avecdoublons
root@DEB13Server: ~# cat avecdoublons

Antoine
Antoine
Azer
Azer
Cedric
Cedric
David
David
Denis
Denis
Fabien
Fabien
Nicolas
Nicolas
Souphiane
Souphiane
Tiphaine
Tiphaine
Xavier
Xavier
root@DEB13Server: ~# _
```

La commande **uniq** permet de ne pas afficher deux fois les doublons

```
root@DEB13Server: ~# uniq avecdoublons > sansdoublons
root@DEB13Server: ~# cat sansdoublons

Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier
root@DEB13Server: ~#
```

```
root@DEB13Server: ~# uniq avecdoublons pasdedoublons
root@DEB13Server: ~#
```

```
root@DEB13Server: ~# cut -d , -f 1 fichier_inexistant.csv > eleves.txt
cut: fichier_inexistant.csv: Aucun fichier ou dossier de ce nom
root@DEB13Server: ~# cat eleves.txt
root@DEB13Server: ~#_
```

Je redirige les sortie d'erreurs dans un fichier à part qui se nomme erreurs.log

```
root@DEB13Server: ~#cut -d , -f 1 fichier_inexistant.csv > eleves.txt 2> erreurs.log
root@DEB13Server: ~#ls -l
total 28
-rw-r--r-- 1 root root 158 17 déc. 14:16 avecdoublons
-rw-r--r-- 1 root root  0 17 déc. 14:23 eleves.txt
-rw-r--r-- 1 root root 64 17 déc. 14:23 erreurs.log
-rw-r--r-- 1 root root 74 10 déc. 17:44 etudiants.txt
-rw-r--r-- 1 root root 218 10 déc. 18:06 notes.csv
-rw-r--r-- 1 root root 74 17 déc. 14:19 pasdedoublons
-rw-r--r-- 1 root root 80 10 déc. 18:07 prenom_tries
-rw-r--r-- 1 root root 74 17 déc. 14:18 sansdoublons
root@DEB13Server: ~#_
```

```
root@DEB13Server: ~#cut -d , -f 1 fichier_inexistant.csv > sio1.txt 2>&1
root@DEB13Server: ~#cat sio1.txt
cut: fichier_inexistant.csv: Aucun fichier ou dossier de ce nom
root@DEB13Server: ~#_
```

```
root@DEB13Server: ~#cut -d , -f 1 fichier_inexistant.csv >> sio1.txt 2>&1
root@DEB13Server: ~#cat sio1.txt
cut: fichier_inexistant.csv: Aucun fichier ou dossier de ce nom
cut: fichier_inexistant.csv: Aucun fichier ou dossier de ce nom
root@DEB13Server: ~#
```

La commande **cal 2025** permet d'afficher le calendrier de l'année 2025, le **pipe** permet d'exécuter deux commandes différentes

```
root@DEB13Server: #cal 2025 | more
2025
  Janvier          Février          Mars
di lu ma me je ve sa di lu ma me je ve sa di lu ma me je ve sa
      1  2  3  4                1                1
  5  6  7  8  9 10 11    2  3  4  5  6  7  8    2  3  4  5  6  7  8
12 13 14 15 16 17 18    9 10 11 12 13 14 15    9 10 11 12 13 14 15
19 20 21 22 23 24 25    16 17 18 19 20 21 22    16 17 18 19 20 21 22
26 27 28 29 30 31      23 24 25 26 27 28      23 24 25 26 27 28 29
                                     30 31

  Avril           Mai           Juin
di lu ma me je ve sa di lu ma me je ve sa di lu ma me je ve sa
      1  2  3  4  5                1  2  3    1  2  3  4  5  6  7
  6  7  8  9 10 11 12    4  5  6  7  8  9 10    8  9 10 11 12 13 14
13 14 15 16 17 18 19    11 12 13 14 15 16 17    15 16 17 18 19 20 21
20 21 22 23 24 25 26    18 19 20 21 22 23 24    22 23 24 25 26 27 28
27 28 29 30            25 26 27 28 29 30 31    29 30

  Juillet         Août          Septembre
di lu ma me je ve sa di lu ma me je ve sa di lu ma me je ve sa
      1  2  3  4  5                1  2          1  2  3  4  5  6
  6  7  8  9 10 11 12    3  4  5  6  7  8  9    7  8  9 10 11 12 13
13 14 15 16 17 18 19    10 11 12 13 14 15 16    14 15 16 17 18 19 20
20 21 22 23 24 25 26    17 18 19 20 21 22 23    21 22 23 24 25 26 27
27 28 29 30 31          24 25 26 27 28 29 30    28 29 30
                          31

  Octobre         Novembre        Décembre
di lu ma me je ve sa di lu ma me je ve sa di lu ma me je ve sa
      1  2  3  4                1                1  2  3  4  5  6
  5  6  7  8  9 10 11    2  3  4  5  6  7  8    7  8  9 10 11 12 13
12 13 14 15 16 17 18    9 10 11 12 13 14 15    14 15 16 17 18 19 20
19 20 21 22 23 24 25    16 17 18 19 20 21 22    21 22 23 24 25 26 27
26 27 28 29 30 31      23 24 25 26 27 28 29    28 29 30 31
                          30

root@DEB13Server: ~#
```

Les commandes **sort** et **tail -3** permettent de trier le fichier et d'afficher les 3 dernières lignes

```
guest@DEB13Server:~$ sort /etc/services | tail -3
zope          9673/tcp      # zope server
zope-ftp     8021/tcp      # zope management by ftp
zserv        346/tcp       # Zebra server
guest@DEB13Server:~$ _
```

La commande **grep** permet d'extraire toutes les lignes d'un fichier qui ne commencent pas par #

```
guest@DEB13Server:~$ grep -v # /etc/services | sort | more_
```

```
acr-nema      104/tcp       dicom          # Digital Imag. & Comm. 300
afpovertcp   548/tcp       # AFP over TCP
afs3-bos     7007/udp      # basic overseer process
afs3-callback 7001/udp      # callbacks to cache managers
afs3-fileserver 7000/udp
afs3-kaserver 7004/udp      # AFS/Kerberos authentication
afs3-prserver 7002/udp      # users & groups database
afs3-rmtsys  7009/udp      # remote cache manager service
afs3-update  7008/udp      # server-to-server updater
afs3-vlserver 7003/udp      # volume location database
afs3-volser  7005/udp      # volume managment server
amanda       10080/tcp     # amanda backup services
amandaidx   10082/tcp     # amanda backup services
amidxtape    10083/tcp     # amanda backup services
amqp         5672/sctp     # AMQP protocol over TLS/SSL
amqp         5672/tcp
amqps        5671/tcp
asf-rmcp     623/udp       # ASF Remote Management and Control Protocol
asp          27374/tcp     # Address Search Protocol
asp          27374/udp
auth         113/tcp       authentication tap ident
babel        6696/udp      # Babel Routing Protocol
bacula-dir   9101/tcp      # Bacula Director
bacula-fd    9102/tcp      # Bacula File Daemon
bacula-sd    9103/tcp      # Bacula Storage Daemon
bbs          7000/tcp
bgp          179/tcp       # Border Gateway Protocol
bgpd         2605/tcp     # bgpd vty (zebra)
biff         512/udp      comsat
binkp        24554/tcp    # binkp fidonet protocol
bootpc      68/udp
bootps      67/udp
canna        5680/tcp     # cannaserver
cfengine     5308/tcp
chargen     19/tcp       ttytst source
chargen     19/udp       ttytst source
cisco-sccp  2000/tcp     # Cisco SCCP
clic-build-daemon 8990/tcp # Common lisp build daemon
clearcase   371/udp      Clearcase
cmip-agent   164/tcp
cmip-agent   164/udp
cmip-man     163/tcp     # ISO mgmt over IP (CMOT)
cmip-man     163/udp
--Plus--
```

Je trie par ordre alphabétique les prénoms qui sont dans le premier champ du fichier **notes.csv**

```
root@DEB13Server: ~# cut -d , -f 1 notes.csv | sort
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier
root@DEB13Server: ~#_
```

J'enregistre la liste dans un fichier qui se nomme **prenoms\_tries**

```
root@DEB13Server: ~# cut -d , -f 1 notes.csv | sort > prenomns_tries.txt
root@DEB13Server: ~# cat prenomns_tries.txt
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier
root@DEB13Server: ~#_
```

J'affiche la liste des utilisateurs du système local à l'aide de la commande **cat** et je les affiche page par page avec la commande **more**

```
root@DEB13Server: ~#cat /etc/passwd | more
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534:./nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:./usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon:/usr/lib/dhcpcd:/bin/false
systemd-timesync:x:991:991:systemd Time Synchronization:./usr/sbin/nologin
messagebus:x:990:990:System Message Bus:/nonexistent:/usr/sbin/nologin
sio:x:1000:1000:sio,,,:/home/sio:/bin/bash
vboxadd:x:989:1:./var/run/vboxadd:/bin/false
sshd:x:987:65534:sshd user:/run/sshd:/usr/sbin/nologin
guest:x:1001:1001:./home/guest:/bin/bash
root@DEB13Server: ~#_
```

J'affiche la liste des **GID** à l'aide de la commande **cut -d : -f 4 /etc/passwd | sort -n | uniq**

```
root@DEB13Server: ~# cut -d : -f 4 /etc/passwd | sort -n | uniq
0
1
2
3
7
8
9
10
12
13
33
34
38
39
60
990
991
998
1000
1001
65534
root@DEB13Server: ~#
```

j'affiche le nombre de ligne et de caractères qui sont contenu dans la liste des fichiers dans le répertoire personnel de **root**

```
root@DEB13Server: ~# ls -l | wc
  11    92   567
root@DEB13Server: ~# ls -l
total 36
-rw-r--r-- 1 root root 158 17 déc. 14:16 avecdoublons
-rw-r--r-- 1 root root  0 17 déc. 14:23 eleves.txt
-rw-r--r-- 1 root root  64 17 déc. 14:23 erreurs.log
-rw-r--r-- 1 root root  74 10 déc. 17:44 etudiants.txt
-rw-r--r-- 1 root root 218 10 déc. 18:06 notes.csv
-rw-r--r-- 1 root root  74 17 déc. 14:19 pasdedoublons
-rw-r--r-- 1 root root  80 10 déc. 18:07 prenom_tries
-rw-r--r-- 1 root root  79 17 déc. 14:49 prenom_tries.txt
-rw-r--r-- 1 root root  74 17 déc. 14:18 sansdoublons
-rw-r--r-- 1 root root 128 17 déc. 14:26 sio1.txt
root@DEB13Server: ~#_
```

A l'aide de la commande **grep** j'extrait tout les fichiers qui ont l'extension **.txt**

```
root@DEB13Server: ~# ls -l | grep .txt
-rw-r--r-- 1 root root  0 17 dec. 14:23 eleves.txt
-rw-r--r-- 1 root root  74 10 déc. 17:44 etudiants.txt
-rw-r--r-- 1 root root  79 17 déc. 14:49 prenom_tries.txt
-rw-r--r-- 1 root root 128 17 déc. 14:26 sio1.txt
root@DEB13Server: ~#
```

J'affiche la ligne du fichier **notes.csv** concernant l'étudiant **Azer**

```
root@DEB13Server: ~# cat notes.csv | grep -i azer
Azer, SISR, 13, AB
root@DEB13Server: ~#_
```

## 2. Utilisation des jokers

Je me login en tant que **guest** et je me déplace dans le répertoire **/bin** à l'aide de la commande **cd** puis avec la commande **ls r\*** j'affiche toutes les commandes qui commencent par la lettre **r**

```
guest@DEB13Server:/bin$ ls r*
ranlib rdma readlink rename.ul reportbug reset resizepart rgrep rmdir routel rtstat runcon run-with-aspell
rbash readelf realpath renice report-hw resizecons rev rm rnano rpcgen run0 run-parts rview
guest@DEB13Server:/bin$
```

J'affiche les commandes composées de 5 caractères

```
guest@DEB13Server:/bin$ ls ?????
bz2sum chage chown dmesg flock gzexe ld.so lsmem lzcat mount nstat pgrep pkill reset sdiff split tsort which zdiff
bzcat chcon cksum egrep gmake iconv login lsmem lzcmp namei pager pidof pl2pm rgrep shred strip uname xargs zdump
bzcmp chgrp clear eject gprof ijoin lsblk lsmem mandb nohup partx ping4 prove rmdir skill tload users xauth zgrep
bzexe chmod colrm false groff ipcmk lscpu lspci mkdir nproc paste ping6 pzstd rnano sleep touch watch xzcat zless
bzzip2 chown diff3 fgrep grops ipcrm lsipc lsusb mknod nroff patch pinky rbash rview snice troff wdctl xzcmp zmore
guest@DEB13Server:/bin$
```

J'affiche les commandes de 2 caractères commençant par la lettre **e**

```
guest@DEB13Server:/bin$ ls e?
ex
guest@DEB13Server:/bin$
```

J'affiche les commandes commençant par la lettre **w,x,y ou z**

```

guest@DEB13Server:/bin$ ls [wxyz]*
w      word-list-compress      x86_64-linux-gnu-gcc-ar      x86_64-linux-gnu-ld.bfd      xz      zdump
waitpid  wtmpdb                  x86_64-linux-gnu-gcc-ar-14   x86_64-linux-gnu-lto-dump    xzcat   zegrep
wall     x86_64                  x86_64-linux-gnu-gcc-nm      x86_64-linux-gnu-lto-dump-14 xzcmp   zfgrep
watch    x86_64-linux-gnu-addr2line x86_64-linux-gnu-gcc-nm-14   x86_64-linux-gnu-nm          xzdiff  zforce
wc       x86_64-linux-gnu-ar     x86_64-linux-gnu-gcc-ranlib  x86_64-linux-gnu-objcopy     xzegrep zgrep
wdctl   x86_64-linux-gnu-as     x86_64-linux-gnu-gcc-ranlib-14 x86_64-linux-gnu-objdump    xzfgrep zipdetails
wget    x86_64-linux-gnu-c++filt x86_64-linux-gnu-gcov        x86_64-linux-gnu-ranlib      xzgrep  zless
whatis  x86_64-linux-gnu-cpp    x86_64-linux-gnu-gcov-14     x86_64-linux-gnu-readelf    xzless  zmore
whereis x86_64-linux-gnu-cpp-14 x86_64-linux-gnu-gcov-dump   x86_64-linux-gnu-size       xzmore  znew
which   x86_64-linux-gnu-elfedit x86_64-linux-gnu-gcov-dump-14 x86_64-linux-gnu-strings    yes      zstd
which.debianutils x86_64-linux-gnu-g++ x86_64-linux-gnu-gcov-tool   x86_64-linux-gnu-strip      ydomainname zstdcat
whiptail x86_64-linux-gnu-g++-14 x86_64-linux-gnu-gcov-tool-14 xargs                          zcat     zstdgrep
who     x86_64-linux-gnu-gcc    x86_64-linux-gnu-gprof      xauth                          zcmp     zstdless
whoami  x86_64-linux-gnu-gcc-14 x86_64-linux-gnu-ld         xsubpp                          zdiff    zstdmt
guest@DEB13Server:/bin$

```

J'affiche les commandes qui ne commencent pas par les lettres qui sont entre **a et v**

```

guest@DEB13Server:/bin$ ls [!a-v]*
['
VBoxAudioTest  whiptail      x86_64-linux-gnu-g++-14      x86_64-linux-gnu-gprof      xsubpp   zdump
VBoxClient     who           x86_64-linux-gnu-gcc         x86_64-linux-gnu-ld         xz        zegrep
VBoxControl    whoami       x86_64-linux-gnu-gcc-14     x86_64-linux-gnu-ld.bfd    xzcat    zfgrep
VBoxDRMClient  word-list-compress x86_64-linux-gnu-gcc-ar     x86_64-linux-gnu-lto-dump  xzcmp    zforce
w              wtmpdb       x86_64-linux-gnu-gcc-nm     x86_64-linux-gnu-lto-dump-14 xzdiff   zgrep
waitpid       x86_64      x86_64-linux-gnu-gcc-nm-14   x86_64-linux-gnu-nm        xzegrep  zipdetails
wall         x86_64-linux-gnu-addr2line x86_64-linux-gnu-gcc-ranlib x86_64-linux-gnu-objcopy   xzfgrep  zless
watch        x86_64-linux-gnu-ar     x86_64-linux-gnu-gcc-ranlib-14 x86_64-linux-gnu-objdump  xzgrep   zmore
wc           x86_64-linux-gnu-as     x86_64-linux-gnu-gcov       x86_64-linux-gnu-ranlib    xzless   znew
wdctl       x86_64-linux-gnu-c++filt x86_64-linux-gnu-gcov-14    x86_64-linux-gnu-readelf  xzmore   zstd
wget        x86_64-linux-gnu-cpp    x86_64-linux-gnu-gcov-dump   x86_64-linux-gnu-size     yes       zstdcat
whatis      x86_64-linux-gnu-cpp-14 x86_64-linux-gnu-gcov-dump-14 x86_64-linux-gnu-strings  ydomainname zstdgrep
whereis     x86_64-linux-gnu-elfedit x86_64-linux-gnu-gcov-tool-14 x86_64-linux-gnu-strip    zcat     zstdless
which       x86_64-linux-gnu-g++    x86_64-linux-gnu-gcov-tool  xargs                        zcmp     zstdmt
            x86_64-linux-gnu-gcc    x86_64-linux-gnu-ld         xauth
guest@DEB13Server:/bin$

```

### 3. Utilisation des caractères de protection

Le contre slash permet de ne pas prendre en compte le #

```
guest@DEB13Server:~$ touch #commentaire
touch: op rande de fichier manquant
Saisissez « touch --help » pour plus d'informations.
guest@DEB13Server:~$ touch \#commentaire
guest@DEB13Server:~$ ls -l *comme*
-rw-rw-r-- 1 guest guest 0 17 d c. 15:24 '#commentaire'
guest@DEB13Server:~$
```

Les **cotes simples** permettent de prot ger un ensemble de caract re

```
guest@DEB13Server:~$ touch 'Fichier dont le nom contient des espaces'
guest@DEB13Server:~$ ls -l *espace*
-rw-rw-r-- 1 guest guest 0 17 d c. 21:17 'Fichier dont le nom contient des espaces'
guest@DEB13Server:~$
```

Les  **doubles cotes**  fonctionnent comme les **cotes simples**   part que le caract re \$ interpr te les variables

```
guest@DEB13Server:~$ echo "Mon r pertoire de connexion : $HOME"
Mon r pertoire de connexion : /home/guest
guest@DEB13Server:~$ _
```

## 4. Autres caractères

Le **point-virgule** permet de séparer les commandes en mettant à la ligne

```
guest@DEB13Server:~$ echo "voici la date :" ; date
voici la date :
mer. 17 déc. 2025 15:27:33 CET
guest@DEB13Server:~$ _
```

Les **anti-cotes** permettent d'interpréter une commande dans une commande

```
guest@DEB13Server:~$ echo "Voici la date : `date`"
Voici la date : mer. 17 dec. 2025 15:28:40 CET
guest@DEB13Server:~$
```