

1. Запустите новую сессию с именем "`whoami`_part2". Все команды сохраняйте в файле \$USER_part.log, проследите за тем, чтобы он не содержал есс-последовательности форматирования и расцветки.

```
eltex-pg1-v6@eltex-2025-autumn:~$ screen -ls
There is a screen on:
      1584281.eltex-pg1-v6_part2      (10/15/2025 07:22:41 PM)      (Detached)
1 Socket in /run/screen/S-eltex-pg1-v6.
eltex-pg1-v6@eltex-2025-autumn:~$ screen -S $(whoami)_part2 -L -Logfile $USER_part.log
```

2. С помощью механизма дополнения имен команд выведите все команды, которые начинаются на «ls».

```
eltex-pg1-v6@eltex-2025-autumn:~$ bash
eltex-pg1-v6@eltex-2025-autumn:~$ ls
ls          lsattr      lsblk       lsblk_release  lscpu       lsfd        lsinitramfs  lsipc       lsirq       lslocks     lslogins    lsmem       lsmod       lsns       lsosf       lspci       lsusb
```

3. С помощью механизма дополнения имен переменных выведите все переменные, которые начинаются с «HIST».

```
eltex-pg1-v6@eltex-2025-autumn:~$ echo $HIST
$HISTCMD      $HISTCONTROL  $HISTFILE     $HISTFILESIZE $HISTSIZE
eltex-pg1-v6@eltex-2025-autumn:~$ echo $HIST
```

4. Узнайте, сколько команд может храниться в файле истории.

```
bash: ECHO: command not found
eltex-pg1-v6@eltex-2025-autumn:~$ echo $HISTFILESIZE
2000
```

5. Выведите имена файлов и каталогов из домашнего каталога, которые начинаются с «.».

```
eltex-pg1-v6@eltex-2025-autumn:~$ ls -a .
.      .bash_history  .bash_logout  .bashrc       .config        .log           _part.log     .profile      report        .report.swp    .viminfo
eltex-pg1-v6@eltex-2025-autumn:~$ ls -a ~
.      .bash_history  .bash_logout  .bashrc       .config        .log           _part.log     .profile      report        .report.swp    .viminfo
```

6. Настройте вывод даты выполнения команд, хранящихся в истории.

```

47 2025-10-15 19:22:41 screen -S top -d -m
48 2025-10-15 19:22:41 screen -r top
49 2025-10-15 19:22:41 screen -r 1753695
50 2025-10-15 19:22:41 ls -l
51 2025-10-15 19:22:41 screen -r 1753695
52 2025-10-15 19:22:41 ls -l
53 2025-10-15 19:22:41 vim report
54 2025-10-15 19:22:41 screen -ls
55 2025-10-15 19:22:41 screen -S 1753695 -X quit
56 2025-10-15 19:22:41 screen -S 1750918 -X quit
57 2025-10-15 19:22:41 screen -S 1750163 -X quit
58 2025-10-15 19:22:41 screen -S 4397 -X quit
59 2025-10-15 19:22:41 screen -S 3511 -X quit
60 2025-10-15 19:22:41 screen -S 3033 -X quit
61 2025-10-15 19:22:41 screen -ls
62 2025-10-15 19:22:41 script --flush --quiet --command bash --return
63 2025-10-15 19:22:41 screen -S 1527441 -X quit
64 2025-10-15 19:22:41 screen -ls
65 2025-10-15 19:22:41 screen -S 'whoami'_part2
66 2025-10-15 19:22:41 ls
67 2025-10-15 19:22:41 export TERM=dumb
68 2025-10-15 19:22:51 ls
69 2025-10-15 19:28:45 bash
70 2025-10-15 19:35:59 HISTFILESIZE
71 2025-10-15 19:37:16 ECHO $HISTFILESIZE
72 2025-10-15 19:37:24 echo $HISTFILESIZE
73 2025-10-15 19:38:09 ls -d ~/.* 2>/dev/null
74 2025-10-15 19:39:26 bash
75 2025-10-15 19:41:54 ls -a ~ | grep '^\. '
76 2025-10-15 19:46:32 echo $HIST
77 2025-10-15 19:47:56 echo $HISTFILESIZE
78 2025-10-15 19:50:48 ls -a .
79 2025-10-15 19:52:02 ls -a ~
80 2025-10-15 19:59:19 export HISTTIMEFORMAT='%F %T '
81 2025-10-15 20:01:26 export HISTCONTROL=ignoredups:erasedups
82 2025-10-15 20:01:28 history
eltex-pg1-v6@eltex-2025-autumn: $

```

7. Настройте автоматическое сохранение набираемых команд в файле истории:

- введите любую команду, например, команду date;

```

eltex-pg1-v6@eltex-2025-autumn: $ date
Wed Oct 15 08:06:55 PM +07 2025

```

- проверьте, есть ли эта команда в кэше и файле истории команд;

```

82 2025-10-15 20:06:55 date

```

```

exit
eltex-pg1-v6@eltex-2025-autumn: $ history -a
eltex-pg1-v6@eltex-2025-autumn: $ tail -5 ~/.bash_history
date
#1760534116
tail -5 ~/.bash_history
#1760534220
history -a

```

- определить переменную PROMPT_COMMAND так, чтобы кэш истории сохранялся в файле истории;

```
eltex-pg1-v6@eltex-2025-autumn: $ export PROMPT_COMMAND='date'
Wed Oct 15 08:19:31 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ echo $PROMPT_COMMAND
date
Wed Oct 15 08:19:34 PM +07 2025
```

- ввести любую команду и проверить, появилась ли эта команда в кэше и файле истории.

```
eltex-pg1-v6@eltex-2025-autumn: $ ls
_part.log report
Wed Oct 15 08:26:17 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ tail -5 ~/.bash_history
history -a
#1760534695
tail -5 ~/.bash_history && history 5
#1760534735
history -a
Wed Oct 15 08:26:19 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ history -a
Wed Oct 15 08:26:27 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ tail -5 ~/.bash_history
ls
#1760534779
tail -5 ~/.bash_history
#1760534787
history -a
Wed Oct 15 08:26:28 PM +07 2025
```

8. Создайте переменную DATE, в которую запишите текущую дату. Проверьте содержимое переменной.

```
eltex-pg1-v6@eltex-2025-autumn: $ DATE=$(date)
Wed Oct 15 08:29:33 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ echo $DATE
Wed Oct 15 08:29:33 PM +07 2025
Wed Oct 15 08:30:13 PM +07 2025
```

9. Создайте переменную TIME, в которую запишите текущее время. Проверьте содержимое переменной.

```

eltex-pg1-v6@eltex-2025-autumn: $ TIME=$(date +%T)
Wed Oct 15 08:31:42 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ echo $TIME
20:31:42
Wed Oct 15 08:31:45 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $

```

10. Создайте переменную DATE_TIME в которую поместите значения из переменных DATE и TIME, разделенных пробелом. Проверьте содержимое переменной.

```

eltex-pg1-v6@eltex-2025-autumn: $ DATE_TIME="$DATE $TIME"
Wed Oct 15 08:32:17 PM +07 2025
eltex-pg1-v6@eltex-2025-autumn: $ echo $DATE_TIME
Wed Oct 15 08:29:33 PM +07 2025 20:31:42
Wed Oct 15 08:32:21 PM +07 2025

```

11. Выведите имена файлов, содержащие хотя бы одну цифру, из каталогов /bin и /sbin.

```

x86_64-linux-gnu-gcc-nm
x86_64-linux-gnu-gcc-nm-12
x86_64-linux-gnu-gcc-ranlib
x86_64-linux-gnu-gcc-ranlib-12
x86_64-linux-gnu-gcov
x86_64-linux-gnu-gcov-12
x86_64-linux-gnu-gcov-dump
x86_64-linux-gnu-gcov-dump-12
x86_64-linux-gnu-gcov-tool
x86_64-linux-gnu-gcov-tool-12
x86_64-linux-gnu-gold
x86_64-linux-gnu-gp-archive
x86_64-linux-gnu-gp-collect-app
x86_64-linux-gnu-gp-display-html
x86_64-linux-gnu-gp-display-src
x86_64-linux-gnu-gp-display-text
x86_64-linux-gnu-gprof
x86_64-linux-gnu-gprofng
x86_64-linux-gnu-ld
x86_64-linux-gnu-ld.bfd
x86_64-linux-gnu-ld.gold
x86_64-linux-gnu-lto-dump
x86_64-linux-gnu-lto-dump-12
x86_64-linux-gnu-nm
x86_64-linux-gnu-objcopy
x86_64-linux-gnu-objdump
x86_64-linux-gnu-ranlib
x86_64-linux-gnu-readelf
x86_64-linux-gnu-size
x86_64-linux-gnu-strings
x86_64-linux-gnu-strip
dumpe2fs
e2freefrag
e2fsck
e2image
e2label
e2mmpstatus
e2scrub
e2scrub_all
e2undo
e4crypt
e4defrag
fsck.ext2
fsck.ext3
fsck.ext4
killall5
mke2fs
mkfs.ext2
mkfs.ext3
mkfs.ext4
resize2fs
tune2fs
update-grub2

```

12. Измените приглашение так, чтобы выводились имя хоста, имя пользователя и время: `имя_пользователя@имя_хоста-НН:ММ>` (Используйте переменные `bash` и команду `date`)

```
eltex-pg1-v6@eltex-2025-autumn: $ PS1="\u@\h-\$(date +%H:%M)> "  
Wed Oct 15 08:36:31 PM +07 2025  
eltex-pg1-v6@eltex-2025-autumn-20:36>
```

13. Сделайте так, чтобы в запускаемом интерпретаторе `bash` выводилось приглашение, установленное в родительском интерпретаторе `bash`.

```
eltex-pg1-v6@eltex-2025-autumn-20:39> source ~/.bashrc  
eltex-pg1-v6@eltex-2025-autumn:~$ export PS1='\u@\h-\$(date +%H:%M)> '  
eltex-pg1-v6@eltex-2025-autumn-20:59> source ~/.bashrc  
eltex-pg1-v6@eltex-2025-autumn:~$
```

14. Одной командной строкой создайте в домашнем каталоге подкаталоги для каждого месяца текущего года вида `YYYY-MM`(год реализуйте с помощью команды `date` и командной подстановки).

```
eltex-pg1-v6@eltex-2025-autumn:~$ mkdir -p ~/${for m in {01..12}; do echo "$(date +%Y)-$m"; done}  
eltex-pg1-v6@eltex-2025-autumn:~$ ls -ld ~/${for m in {01..12}; do echo "$(date +%Y)-$m"; done}  
2025-01 2025-02 2025-03 2025-04 2025-05 2025-06 2025-07 2025-08 2025-09 2025-10 2025-11 2025-12 /home/eltex-pg1-v6/2025-01  
eltex-pg1-v6@eltex-2025-autumn:~$
```