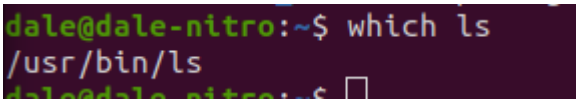
Damir Nabiullin - Lab 3

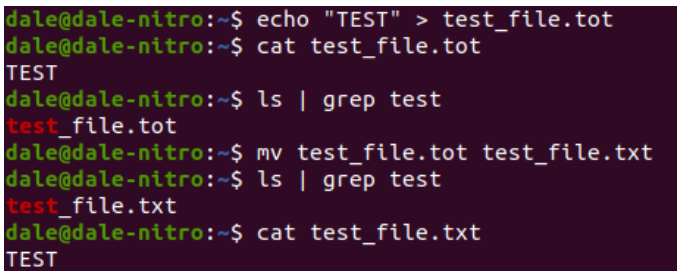
Answers:

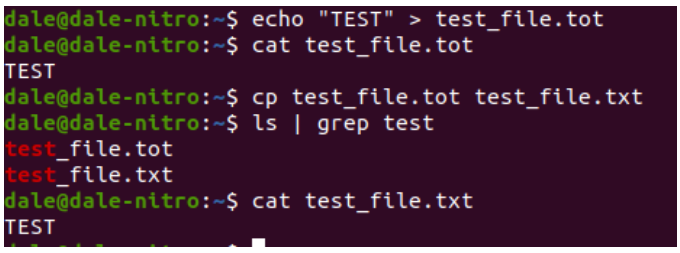
1. Pipe in this command is used to put text from `/etc/apt/sources.list` to `less` command.

2. Section 5 mean **File formats and conventions**. I found it by `man man` command. To find it we can use such command: man 5 `/etc/passwd`. But some pages have all of these sections and some does not have.
3. On my machine the full path for the ls command is /usr/bin/ls. I found it by using `which` command.



4. I used such two commands: `mv test_file.tot test_file.txt` and `cp test_file.tot test_file.txt`





5. I created such command: `echo -e $my_string | sort | uniq -u > savefile.txt && echo "$USER" >> savefile.txt`

```
dale@dale-nitro:~$ my_string='The location of hundreds of crab pots\nLittle Red Riding Hood\nThe location o
f hundreds of crab pots\nThe location of hundreds of crab pots\nThe sound of thunder\nEight hours in a row\
nAll aboard\nEight hours in a row'
dale@dale-nitro:~$ echo -e $my_string | sort | uniq -u > savefile.txt && echo "$USER" >> savefile.txt
dale@dale-nitro:~$ cat savefile.txt
All aboard
Little Red Riding Hood
The sound of thunder
dale
```

6. To suppress the output and error of the `ping 127.0.0.1` command I decided to specify files for stdout and stderr. I put the stdout data of this command to ping.log and stderr to the same place as stdout: `ping 127.0.0.1 1> ping.log 2>&1`

```
dale@dale-nitro:~$ ping 127.0.0.1 1> ping.log 2>&1
^Cdale@dale-nitro:~$ cat ping.log
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.052 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.039 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.045 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.044 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3076ms
rtt min/avg/max/mdev = 0.039/0.045/0.052/0.004 ms
dale@dale-nitro:~$ ping 127.0.0. 1> ping.log 2>&1
dale@dale-nitro:~$ cat ping.log
ping: 127.0.0.: Name or service not known
```

7. I decided to write a command that read lines before *end* command, sorts given lines, and saves them to *sorted.txt* file `sort << end | nl --b a > sorted.txt`:

```
dale@dale-nitro:~$ sort << end | nl --b a > sorted.txt
> abc
>
> def
>
>
> zwx
>
> klm
> end
dale@dale-nitro:~$ cat sorted.txt
     1
     2
     3
     4
     5  abc
     6  def
     7  klm
     8  zwx
```

8. `cd /home/$USER/testdir`, `cd ../../home/$USER/testdir`, `cd ~/testdir`

```
dale@dale-nitro:~$ cd /usr/share/
dale@dale-nitro:/usr/share$ mkdir /home/$USER/testdir
dale@dale-nitro:/usr/share$ ls /home/$USER/ | grep test
testdir
dale@dale-nitro:/usr/share$ cd /home/$USER/testdir
dale@dale-nitro:~/testdir$ cd /usr/share/
dale@dale-nitro:/usr/share$ cd ../../home/$USER/testdir
dale@dale-nitro:~/testdir$ cd /usr/share/
dale@dale-nitro:/usr/share$ cd ../../home/$USER^Cestdir
dale@dale-nitro:/usr/share$ cd ~/$USER/testdir
bash: cd: /home/dale/dale/testdir: No such file or directory
dale@dale-nitro:/usr/share$ cd ~/testdir
```

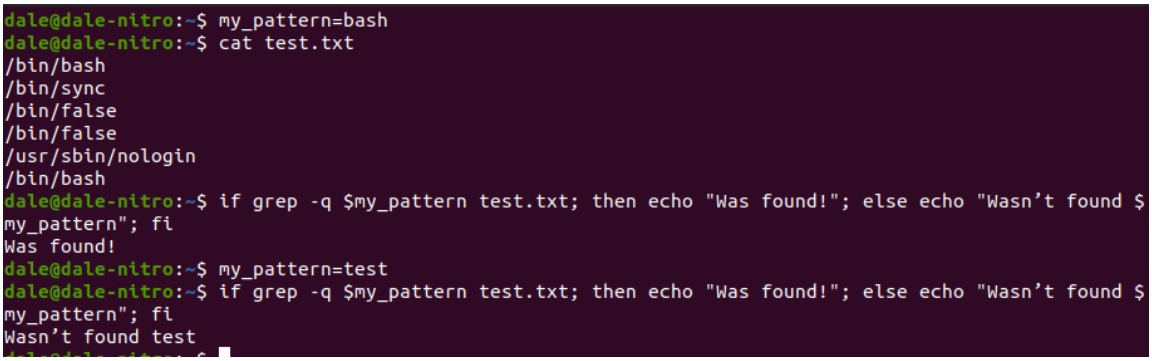
9. I decided to write such command: `cut -d: -f7 /etc/passwd | sort | uniq`. This command gets last column of file `/etc/passwd`, sorts it, and gets unique commands.

```
dale@dale-nitro:~$ cut -d: -f7 /etc/passwd | sort | uniq
/bin/bash
/bin/false
/bin/sync
/usr/sbin/nologin
```

10. To find all man pages with specific words we can use such command: `man -wK <word>`. In our case this command looks like: `man -wK malloc`

```
dale@dale-nitro:~$ man -wK malloc
/usr/share/man/man1/x86_64-linux-gnu-gcc-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-gcc-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-gcc-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-gcc-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-gcc-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-g++-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-g++-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-g++-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-g++-9.1.gz
/usr/share/man/man1/x86_64-linux-gnu-g++-9.1.gz
/usr/share/man/man1/ltrace.1.gz
/usr/share/man/man1/memusage.1.gz
/usr/share/man/man1/mtrace.1.gz
/usr/share/man/man1/top.1.gz
/usr/share/man/man1/git-fast-import.1.gz
/usr/share/man/man1/ld.bfd.1.gz
/usr/share/man/man1/ld.bfd.1.gz
/usr/share/man/man1/ld.bfd.1.gz
/usr/share/man/man1/ld.bfd.1.gz
/usr/share/man/man1/python3.8.1.gz
/usr/share/man/man1/python3.8.1.gz
/usr/share/man/man1/python3.9.1.gz
/usr/share/man/man8/ld.so.8.gz
/usr/share/man/man8/ld.so.8.gz
/usr/share/man/man8/ld.so.8.gz
/usr/share/man/man3/libmaxminddb.3.gz
/usr/share/man/man3/libmaxminddb.3.gz
/usr/share/man/man3/readline.3readline.gz
```

11. I made such command: `if grep -q $my_pattern test.txt; then echo "Was found!"; else echo "Wasn't found $my_pattern"; fi` to check if `test.txt` file has `$my_pattern`.


dale@dale-nitro:~$ my_pattern=bash
dale@dale-nitro:~$ cat test.txt
/bin/bash
/bin/sync
/bin/false
/bin/false
/usr/sbin/nologin
/bin/bash
dale@dale-nitro:~$ if grep -q $my_pattern test.txt; then echo "Was found!"; else echo "Wasn't found $
my_pattern"; fi
Was found!
dale@dale-nitro:~$ my_pattern=test
dale@dale-nitro:~$ if grep -q $my_pattern test.txt; then echo "Was found!"; else echo "Wasn't found $
my_pattern"; fi
Wasn't found test