Exercise 1.

Show different commands to make your home folder to your current folder.

cd
cd ~
cd /home/ik0v - (ik0v – my urser name)
cd /home/$(whoami)
cd /home/$USER

Exercise 2.

Show several commands to print out your current folder.

pwd
/bin/pwd
pwd -P
echo “$PWD”

Exercise 3.

Command ls -i show inode number and corresponding file name. Write a command showing on screen inode nr. and file names for files from your home directory, sorted by inode number ascending. (Hint: use piping and sort command).

:~$ ls -i | sort
:~$ ls -il | sort -k 1 - same command, files listed out in format here.

Exercise 4.

df command shows all mounted partitions with detailed info about each of them. Use df command and answer following questions:

a) **How many partitions do you have on your Linux?**
   :~$ df -h | tail -n +2 | wc -l
   There are 9 partitions totally on my Linux

b) **What partition your home folder is located in and which filesystem it use?**

   :~$ df -T /home | awk '{print $1,$2}'
   /dev/nvme0n1p5 ext4

c) Assume that answer to a) is to or more partitions. Try following: create a hard link from one partition to another. Did you get an error? Why?

:~$ ln /home/ik0v/file1.txt /media/ik0v/3CFC-E22C/
ln: failed to create hard link … : Invalid cross-device link
We got error because hard links can’t point to a file in another partition.

Try to create a symbolic link now. Does it work?
:~$ ln -s ~/file1.txt /media/ik0v/TRK_34/file1_sl
Symbolic links works fine in different partitions.

However, I got an error:
ln: failed to create symbolic link … : Operation not permitted
That can be explained with different file systems on these partitions. Home folder is located in main partition of type ext4. Usb drive (TRK_34) got file extension of type vfat.

d) Is it possible to differ between a file and a hard link to that file?

There is no difference between a file and a hard link to that. Hard link takes same place and exist on its own. However we can determine original file by its creation time. Hard links will always be created after file.

Excercise 5.

a) **How many user accounts are on your Linux?**

:~$ getent passwd {1000..2000} | wc -l
8

b) **How many different login shells are used now?**

:~$ who | wc -l
:~$ users | wc -w

c) Second column in passwd file is a star. What does it mean? What if there is another character instead of star?

Star means that user doen't have password..
Otherwise it is x char instead of *.

Exercise 6.

What is PID number?

PID stands for process identity nr. Each active process has unique PID number.

Exercise 7.

Explain error message what u get in next lines:
$> ls
abc abd abe abf abg abh
$> rm abc ab*
rm: abc: No such file or directory

Running ls command after given code, I can see that files are deleted despite error.
:~$ rm ab* - command removes all files starting with (ab) without error.
:~$ rm abc – command removes abc fiel withour error.
Now I can make a conclusion that error comes because we try to delete abc file twice.

Exercise 8.

Assume that you have permissions to write to a file but not to delete it.

a) **What permissions does this file have?**

--w-------

To be able to write to a file, we need minimum write permissions. However, we cant see content of that file in this case.
-rw-------

It is a better option, now we can see content of that file and edit it

b) **Provide a command to empty a file without using editor.**

$> > 'file_name'

Exercise 9.

You want to set mask 037 for permissions.

a) **Which command will you use for that?**

umask 037

b) **What is purpose of doing this?**

It gives a stricter permissions when we create a new file or folder.

c) **What are standard permissions for a file after this command?**

:~$ touch file1.txt
:~$ ls -l file1.txt

-rw-r-----
It means read- and write perm. for owner and read permission for group.

d) **.... same for folder?**

:~$ mkdir folder1
:~$ ls -ld folder1

drwxr-----
It means no restriction for owner, Read permission for group and no perm. for other.

e) **Where would you add this command to have a permanent effect?**

umask shall be edited in file ~/.profile in this case.

f) **How can u find out current value of umask?**

:~$ umask

Exercise 10.

Write a command that shows size of all .conf files on your system together.

:~$ cd /
:/$ du -csh -B1 . *.conf

Exercise 11.

Show a command to find a file in your home directory that contains umask command?

:~$ grep -rnw '/home/ik0v' -e 'umask' 2> /dev/null
/home/ik0v/.profile:…

:~$ less ~/.profile | grep umask

Exercise 12.

Using up- and down arrow u operate on commands user earlier in command history.
a) How many such command can be saved on your system?
b) How and where you can change this value?

a) Google says that default number either 500 or 1000 for Debian.
   :~$ echo $HISTSIZE
   1000

b) HISTSIZE = (type a number for a new value)
   * found info that history size is “endless” if you type a number below 0 here.

Exercise 13.

a) How many files do you have in your home directory? (Including subfolders)
b) .... if u take also folders and hidden files?

a) :~$ find ~ -type f | wc -l - 3371 files
b) :~$ find | wc -l - 3649 files

Exercise 14.

a) A person trying to send files to your pc can’t contact it.
Your first step to find an error?

At a first I would check if pc is online via ping command:
:~$ ping <machine_ip>
example:
ping 127.0.0.1

b) Find all files i /etc folder (and subfolders) containing word monitor

:~$ grep -rnw '/etc' -e 'monitor'

c) Did you get quite a long result? Show how can you see one output one by one page.

:~$ grep -rnw '/etc' -e 'monitor' | less

d) Operation in b) gives quite a lot error messages. How can you get rid of them?

:~$ grep -rnw '/etc' -e 'monitor' 2> /dev/null

e) Can you send these errors to another terminal instead?

:~$ grep -rnw '/etc' -e 'monitor' 2> /dev/pts/1

Exercise 15.

Assume that you forgot a command to remove files. How can you find this command from a shell?

:~$ man -k ‘remove’

Exercise 16.

Answer with Linux command to each following question:

a) **Copy directory ~/old with all files and subfolders to a directory ~/new**

:~$ mkdir ~/old
:~$ mkdir ~/old/folder{1..5}
:~$ touch old/folder{1..5}/{a..g}
:~$ ls -R ~/old

:~$ cp -r ~/old ~/new
:~$ ls -R ~/new
Both folders have same content.

b) **List all file name (and folders) from /usr/bin starting with "b".**

:~$ find /usr/bin -name ‘b*’

c) **Find out what is a full path to a program ls.**

:~$ which ls
/usr/bin/ls

d) **Add a text "Last line" to existing file myfile.txt.**

:~$ cat >> myfile.txt
Last line
Ctrl^c

e) **Files file1 and file2 are in your current folder. Print out all lines from file1 and file2 containing word "exam" to a screen.**

:~$ cat file[12] | grep exam

f) **Create a symbolic link in a current folder to a folder /var/spool/mail.**

:~$ ln -s /var/spool/mail ./shortcut

Exercise 17.

You run a following commands:
:~$ PATH=""
:~$ ls
What is result from ls command? Explain.

PATH is a way to a place where terminal looks for a given commands.
When we set PATH to empty value, terminal has no idea where to look for a given command.

It is best to call a new bash before that, in order not to harm PATH variable in original shell.
:~$ ls
Result from a command line:
bash: ls: No such file or directory
ls command is located in /usr/bin folder (it can be checked with - which ls).

Exercise 18.

You have a text file readme.txt (add some text) and run following commands:
:~$ cat readme.txt > /dev/null
:~$ cat readme.txt
Explain a difference between these commands and what are they doing.

:~$ cat readme.txt – reads a file readme.txt and prints it on the screen.
:~$ cat readme.txt > /dev/null – does the same but sends output to “nothing” (device null).

Exercise 19.

Send result from date command to a file. Check file’s content. Add a result from ls command to the same file. Check that file again.

:~$ date > mydate.txt
:~$ less mydate.txt
Here comes output for date/time command.
ls >> mydate.txt (>> are used to add a new content to existing in a file).
:~$ less mydate.txt
Result from ls command is saved after date now.

Exercise 20.

How many processes are running on your Linux now?
How many of them belong to you?

:~$ ps -A --no-headers | wc -l - 137
:~$ ps -U $USER --no-headers | wc -l - 32

Exercise 21.

On which partition is located your home folder? How many partitions are totally on your pc? How did you find it? How much space does Linux itself take? What is a kernel version of your Linux?

:~$ df /home
/dev/nvme0n1p5

df -h | tail -n +2 | wc -l
There are 9 partitions on my pc.

:~$ df -h /home

Size Used Avail Use% Mounted on
53G 5.3G 45G 11% /
My homefolder contains just some text files.

:~$ uname -r
4.19.0-10-amd64 - kernel version

Exercise 22.

Go to your home folder and create a subfolder myetc. Go to that folder. Copy all files from /etc folder to myetc folder. Check that all files were copied by counting files in both folders. Remove files from myetc folder and check this operation. Copy files from /etc again, now with option following along each file being copied.

:~$ mkdir minetc
:~$ sudo cp -r /etc/* ~/myetc/
:~$ find /etc -type f | wc -l
944 files
:~$ sudo find ~/minetc -type f | wc -l
944 files

Exercise 23.

Use find command and show file names for all files located under /home folder (including subfolders). What can you do to avoid errors on your screen? Can you send these errors to another terminal? Use find command to find all files bigger than 10 Mbyte on your system. Extend same command to make sure these files are 10 Mbyte or more.

```
:~$ find /home -type f
:~$ find /home -type f 2> /dev/null
:~$ find /home -type f 2> /dev/pts/1
:~$ find /home -type f –size +10M
:~$ find /home -type f –size +10M –exec ls –lh {} \;
```