jamk.fi

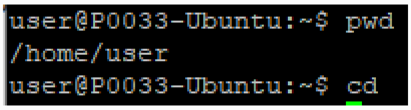
Linux perusteet [TTC1040]

harjoitus T4

Maarit Salo

15.9.2021

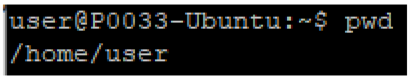
1. **What is the path for your home directory? How do you change your current working directory to your home directory?**



```
user@P0033-Ubuntu:~$ pwd
/home/user
user@P0033-Ubuntu:~$ cd
```

cd changes the directory to my home directory. Cd ~ also works.

2. **How can you find out your current working directory?**

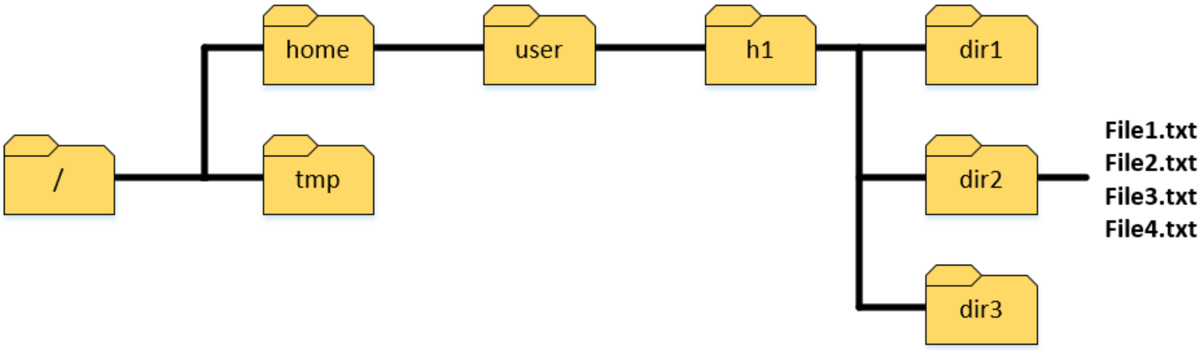


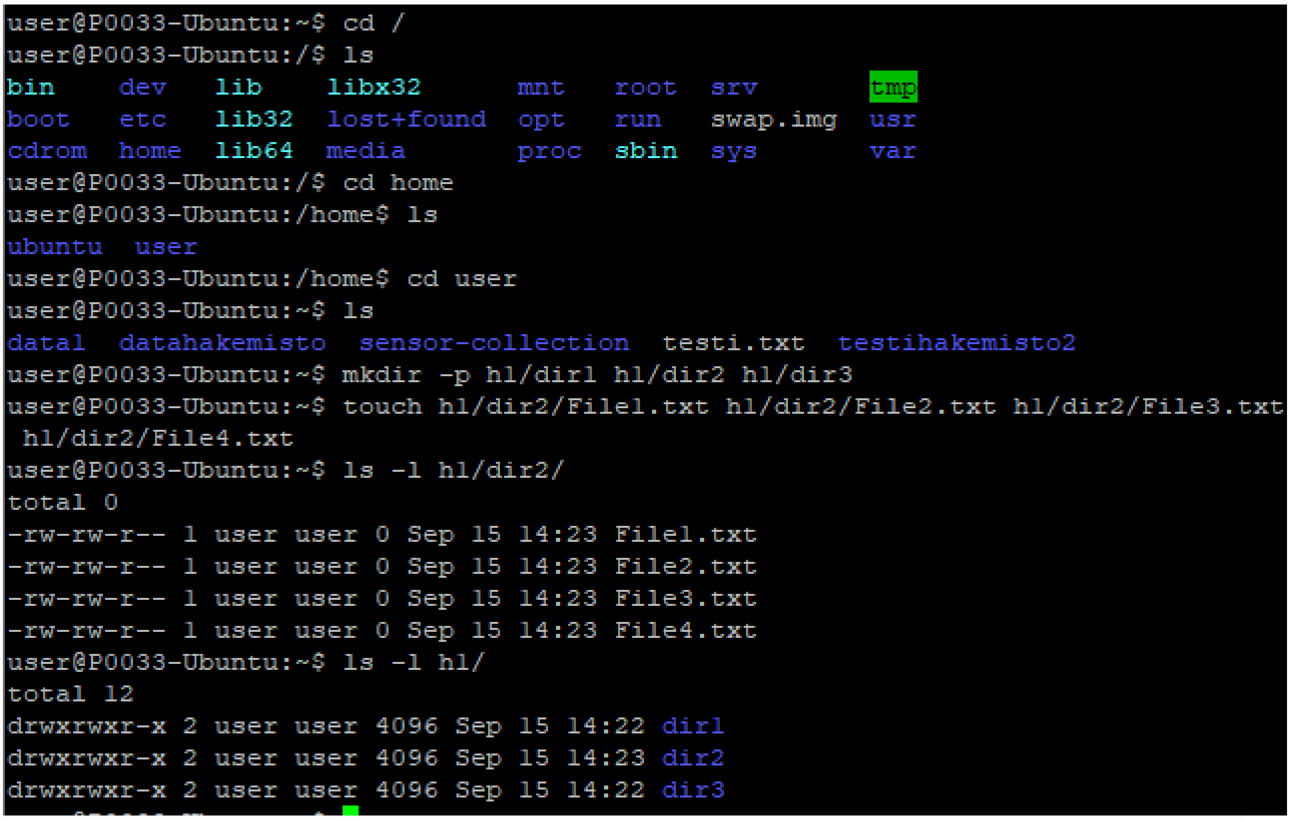
```
user@P0033-Ubuntu:~$ pwd
/home/user
```

3. **Explain briefly what option -p does when used with mkdir command. In addition, find out what version number of mkdir command you have.**

Normally mkdir creates a child directory if one uses paths upon creation, such as dir1/dir2/child_directory. If one wishes to create parent directories at the same time as creating the child directory, one uses -p option. It creates the parent directories along with the child directory. For example, if I want to create the directory data3 inside data2 and both of them inside data1, I can write mkdir -p data1/data2/data3, so that data1 and data2 are created at the same time. Without -p, we would get an error because the assumption would be that data1 and data2 exist already, and data3 would be the only directory being created.

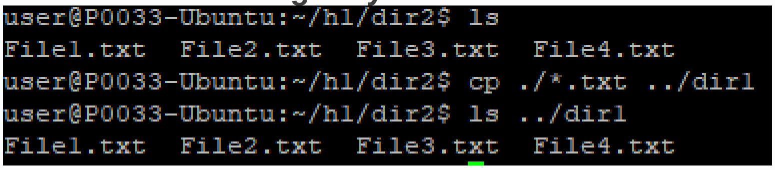
4. **Create the following directory structure including the files to your home directory using your Linux shell. Files can be empty or containing text.**





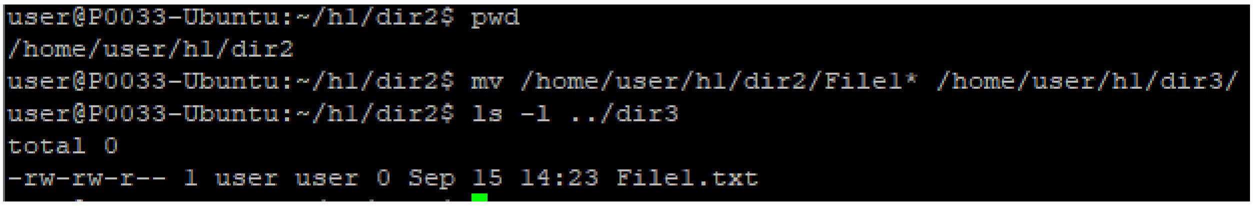
```
user@P0033-Ubuntu:~$ cd /
user@P0033-Ubuntu:/$ ls
bin     dev    lib     libx32      mnt    root   srv        tmp
boot    etc    lib32   lost+found  opt    run    swap.img   usr
cdrom   home   lib64   media       proc   sbin   sys        var
user@P0033-Ubuntu:/$ cd home
user@P0033-Ubuntu:/home$ ls
ubuntu  user
user@P0033-Ubuntu:/home$ cd user
user@P0033-Ubuntu:~$ ls
data1  datahakemisto  sensor-collection  testi.txt  testihakemisto2
user@P0033-Ubuntu:~$ mkdir -p h1/dir1 h1/dir2 h1/dir3
user@P0033-Ubuntu:~$ touch h1/dir2/File1.txt h1/dir2/File2.txt h1/dir2/File3.txt
 h1/dir2/File4.txt
user@P0033-Ubuntu:~$ ls -l h1/dir2/
total 0
-rw-rw-r-- 1 user user 0 Sep 15 14:23 File1.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:23 File2.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:23 File3.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:23 File4.txt
user@P0033-Ubuntu:~$ ls -l h1/
total 12
drwxrwxr-x 2 user user 4096 Sep 15 14:22 dir1
drwxrwxr-x 2 user user 4096 Sep 15 14:23 dir2
drwxrwxr-x 2 user user 4096 Sep 15 14:22 dir3
```

5. **Copy all files with .txt extension from dir2 to dir1 using relative path for directories using only one command.**



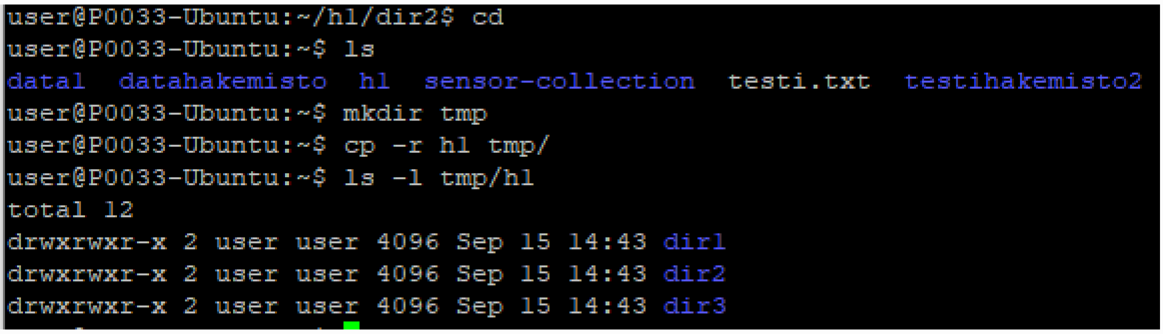
```
user@P0033-Ubuntu:~/h1/dir2$ ls
File1.txt  File2.txt  File3.txt  File4.txt
user@P0033-Ubuntu:~/h1/dir2$ cp ./*.txt ../dir1
user@P0033-Ubuntu:~/h1/dir2$ ls ../dir1
File1.txt  File2.txt  File3.txt  File4.txt
```

6. **Move all files starting with string File1 from dir2 to dir3 using absolute path for directories using only one command.**



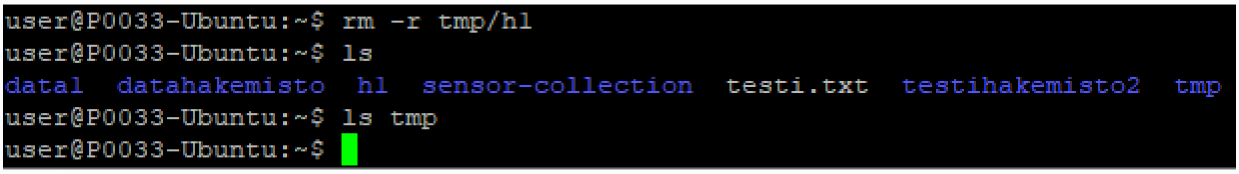
```
user@P0033-Ubuntu:~/h1/dir2$ pwd
/home/user/h1/dir2
user@P0033-Ubuntu:~/h1/dir2$ mv /home/user/h1/dir2/File1* /home/user/h1/dir3/
user@P0033-Ubuntu:~/h1/dir2$ ls -l ../dir3
total 0
-rw-rw-r-- 1 user user 0 Sep 15 14:23 File1.txt
```

7. **Create tmp directory inside your home directory. Copy directory structure created in fourth part starting from directory h1 into the tmp directory in one command. The final directory tree should then look like this: /home/your_user/tmp/h1...**



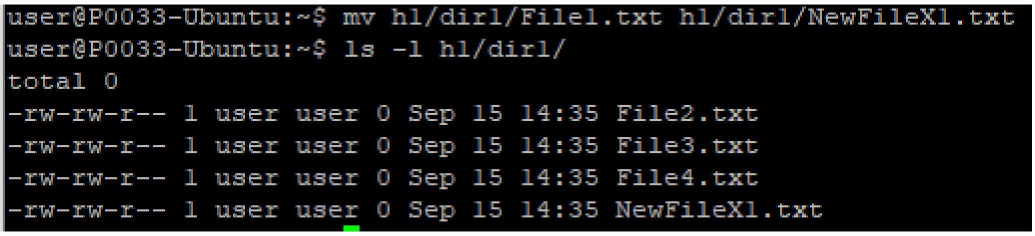
```
user@P0033-Ubuntu:~/h1/dir2$ cd
user@P0033-Ubuntu:~$ ls
data1  datahakemisto  h1  sensor-collection  testi.txt  testihakemisto2
user@P0033-Ubuntu:~$ mkdir tmp
user@P0033-Ubuntu:~$ cp -r h1 tmp/
user@P0033-Ubuntu:~$ ls -l tmp/h1
total 12
drwxrwxr-x 2 user user 4096 Sep 15 14:43 dir1
drwxrwxr-x 2 user user 4096 Sep 15 14:43 dir2
drwxrwxr-x 2 user user 4096 Sep 15 14:43 dir3
```

8. **Remove tmp/h1 directory structure from your home directory using only one command.**



```
user@P0033-Ubuntu:~$ rm -r tmp/h1
user@P0033-Ubuntu:~$ ls
data1  datahakemisto  h1  sensor-collection  testi.txt  testihakemisto2  tmp
user@P0033-Ubuntu:~$ ls tmp
user@P0033-Ubuntu:~$
```

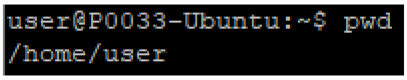
9. **Rename File1.txt file under dir1 in a way it begins with string NewFileX1 (File1.txt → NewFileX1.txt)**



```
user@P0033-Ubuntu:~$ mv h1/dir1/File1.txt h1/dir1/NewFileX1.txt
user@P0033-Ubuntu:~$ ls -l h1/dir1/
total 0
-rw-rw-r-- 1 user user 0 Sep 15 14:35 File2.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:35 File3.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:35 File4.txt
-rw-rw-r-- 1 user user 0 Sep 15 14:35 NewFileX1.txt
```

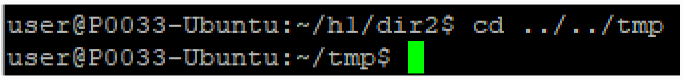
10. **How do you distinguish the following two paths: relative and absolute? What do these terms mean? Give examples from both paths.**

An absolute path is always determined starting from the root directory, whereas the relative path is determined from the current working directory.



```
user@P0033-Ubuntu:~$ pwd
/home/user
```

Example for the absolute path: Using the pwd command, we print our current working directory. It gives us the absolute path from the root directory to our current directory. In the above screenshot, /home/user, / refers to root directory.



```
user@P0033-Ubuntu:~/h1/dir2$ cd ../../tmp
user@P0033-Ubuntu:~/tmp$
```

Example for the relative path: In the screenshot above, we were in the directory /home/user/h1/dir2, and then used the command cd ../../tmp in order to climb 2 levels upwards, back to our home directory, and then descend to the directory tmp.