NB!

For these exercises, it is required that you have an installation of Ubuntu. It is recommended that are using a virtual machine for the training purpose. But, it would also be possible to use a version of Ubuntu that is installed directly on your computer - (But this is at your own risk!)

Users and permissions:

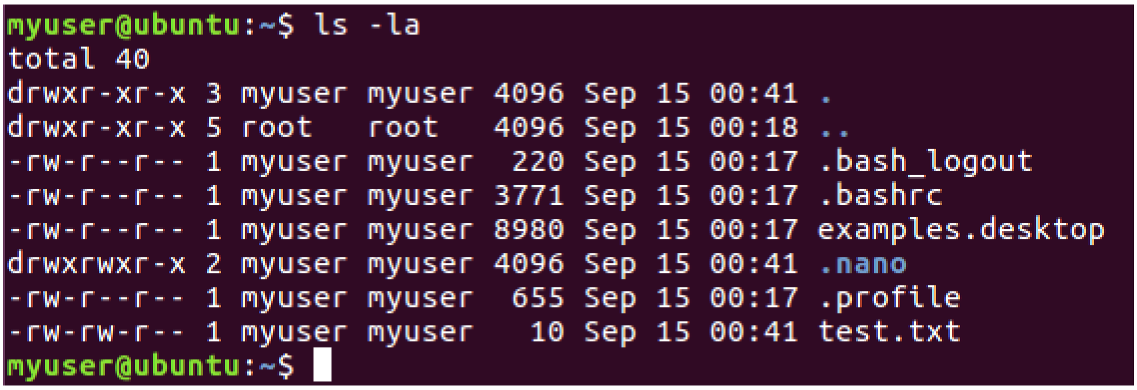
For all these exercises, it is expected that you use the Terminal.

In an operating system, files and folders can have different user permissions. This is to ensure that users won't be able to access and modify stuff in the system, which they shouldn't be able to. In these exercises, we will be looking into how this works.

Use access rights:

1) For the first task, create a new user named: "myuser".
2) Change the user, so that you are using the "myuser" account.
3) Go to the "myuser" home directory, and make a file by the name "test.txt" that contains the text: "Hello, World!"
4) Change the user back to your default user (or any other user that is part of the "sudo" group) and create a new user called "myuser2".
5) From "myuser2", try to change the text in the file "test.txt" that was created by "myuser". (You will get a "permission denied" when trying to save the changes.)

If we look at the permission for the "test.txt" file, we can see that the file owner ("myuser") can read and write to the file. The group ("myuser") can read and write to the file and everyone else can only read the file.

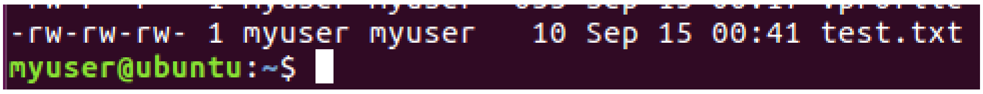


```
myuser@ubuntu:~$ ls -la
total 40
drwxr-xr-x 3 myuser myuser 4096 Sep 15 00:41 .
drwxr-xr-x 5 root   root   4096 Sep 15 00:18 ..
-rw-r--r-- 1 myuser myuser  220 Sep 15 00:17 .bash_logout
-rw-r--r-- 1 myuser myuser 3771 Sep 15 00:17 .bashrc
-rw-r--r-- 1 myuser myuser 8980 Sep 15 00:17 examples.desktop
drwxrwxr-x 2 myuser myuser 4096 Sep 15 00:41 .nano
-rw-r--r-- 1 myuser myuser  655 Sep 15 00:17 .profile
-rw-rw-r-- 1 myuser myuser   10 Sep 15 00:41 test.txt
myuser@ubuntu:~$
```

Let's try to change those settings.

6) Change to user "myuser" and modify the permission for the "test.txt" file, by typing in the command "chmod 666 test.txt".
7) Change to user "myuser2" and try to edit the file.

This time we could save the changes to the file, and if we look at the permission for the file, we can see that everyone can now edit the file.



```
-rw-rw-rw- 1 myuser myuser   10 Sep 15 00:41 test.txt
myuser@ubuntu:~$
```

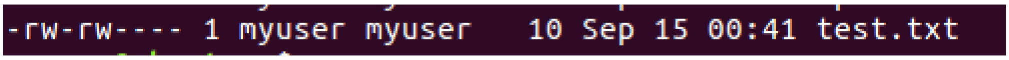
Group permissions:

Now we wish to look a bit into how to group our users. This is because we sometimes wish that only some specific users can access a file or folder.

1) Change to the default user (or any user that is part of the sudo group)
2) Now add the user “myuser2” to be a part of the “myuser” group.

```
sudo usermod -a -G myuser myuser2
```

3) Get a list of all the groups that “myuser2” is part of by typing in: “groups myuser2”
4) Change to the user “myuser” and change the permission rights for the “test.txt” file so that only the owner and the group owners can read and write to the file. (chmod 660 test.txt)
5) Change to the user “myuser2” and try to edit the file. (This will work since the “myuser2” is a part of the group that has permissions to do so).
6) Change to the default user (Or any user that is not a part of the “myuser” group), try to access the file. (This will not be possible since only the owner and the owning group can do this now.)



```
-rw-rw---- 1 myuser myuser   10 Sep 15 00:41 test.txt
```

7) Try to access the file from the default user (or any user that is part of the sudo group) with sudo rights. (This will work, since sudo rights overrule other settings!)

Installing, updating and executing programs:

For all these exercises, it is expected that you use the Terminal.

Application installation:

It is very common, that Linux systems comes with a package manager. In Debian based systems, there will be a package manager called “apt”. We will now try to use that to install different application on our Ubuntu system.

1) Make sure that the “apt” package manager has the newest updated references. (sudo apt-get update)
2) Install the application “sl” with the “apt” package manager.
3) Locate where the application “sl” have been installed on the system and run the application afterwards.

This was just a small example of how to install an application with the package manager. For some application, there might be a bit of configuration to it afterwards.

4) Install a MySQL server on the Ubuntu system by follow this guide: https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-16-04

Application update:

When there is a new version of an application, it is often a very good idea to get the newest version, for a variety of reasons. So let's look at how to update the installed applications in a Ubuntu system.

1) Make sure that the "apt" package manager has the newest updated references. (sudo apt-get update)
2) Use the "apt" package manager to update all the installed packages for the Ubuntu system. (sudo apt-get upgrade)

Mono (.NET library for Linux):

In this exercise we wish to write a small application in C# and then make it run in a Linux environment. Since C# is a part of the .Net framework that is a Microsoft Windows thing, we need to install a Linux version of the framework, so that we can run the program in Linux.

1) Search for the correct package name in "apt", for the "Mono" package to run .Net libraries in Linux. (apt-cache search [search word])
2) Install the complete package from the "apt" package manager.
3) Write a small C# console application ("Hello, World!" would be very fine), compile it, and put it on the Ubuntu system.
4) Use "mono" to run the C# application.

Useful links:

How to Add and Delete Users on Ubuntu 16.04

- https://www.digitalocean.com/community/tutorials/how-to-add-and-delete-users-on-ubuntu-16-04

An Introduction to Linux Permissions

- https://www.digitalocean.com/community/tutorials/an-introduction-to-linux-permissions