

[Previous](#) - [Up](#) - [Next](#)

## Tartalomjegyzék

- [1 Linux permissions](#)
  - ◆ [1.1 chmod](#)
  - ◆ [1.2 More info](#)
- [2 Html](#)
  - ◆ [2.1 Creating your homepage](#)
  - ◆ [2.2 Important](#)
  - ◆ [2.3 Syntax](#)
  - ◆ [2.4 Tasks](#)
  - ◆ [2.5 Useful links](#)

## Linux permissions

- Permission groups
  - ◆ **superuser** (su), sys admins
  - ◆ **owner**, the user who made the file/directory, ex. *someone*
  - ◆ **group**, a subset of users with a distinct name, ex. *student*
  - ◆ **others**, everyone else, public, guest, untrusted users
- Types of permissions
  - ◆ **w**: Write, can modify
  - ◆ **r**: Read, can read
  - ◆ **x**: eXecute, run a program, or list a directory
  - ◆ **-**: none

It is specified who can do what, ex:

- owner can write, read and execute
- its group can read and execute, but not modify
- others cannot do anything

The superusers (su) can do anything at any time, only a superuser can make an other user super.

These can be expressed with a set of three alpha-numeric charactes:

- **u**: user, owner
- **g**: group
- **o**: others
- **a**: all of the above
- read: **r** or **4**
- write: **w** or **2**
- execute: **x** or **1**
- none: **-** or **0**

The **a** is not the same as **o**, because it is possible that a group does not have a permission, but someone outside of the groups has.

The permissions can be represented with a 10 character string:

1	2	3	4	5	6	7	8	9	10
type									
read	write	execute	read	write	execute	read	write	execute	

```
$ ls -l ~
drwxr-xr-x 8 borbely student 4096 Aug 30 23:24 Desktop
drwxr-xr-x 2 borbely student 4096 Mar 27 2012 Downloads
drwxr-xr-x 2 borbely student 4096 Oct 20 2009 Drives
drwx----- 2 borbely student 4096 Apr 20 10:42 mail
drwxr-xr-x 7 borbely student 4096 Sep 6 13:01 public_html
$ _
```

The `public_html` folder is owned by `borbely`, its group is `student`, permissions: `drwxr-xr-x`

- type: it is a **d**irectory
- my permissions **rw**x means that owner can do anything
- other students' permissions: **r-x**, they can read but not write
- others: **r-x**, they can still read but not write

Numerically, you can encode this in 3 digits. The three numbers are (from left to right): owner, group and others. The numbers 4: read, 2: write, 1: execute. They can be added. Example:

`/home/student/borbely/public_html` has permission: 755 meaning

- owner (`borbely`) 7=4+2+1: read, write, execute
- group (`student`) 5=4+1: read and execute
- other 5=4+1: read and execute

## chmod

You can change the permissions with `chmod` Examples:

- `chmod 700 ~/info_hazi`: nobody can see anything, except me (of course superusers still can see it).
- `chmod 750 ~/important_work`: In this folder one can share data with group members, but not others. Also group can only see it, not modify.
- `chmod 754 -R ~/public_html`: The **-R** applies the permissions recursively in the subfolders of the folder, and every file in it.

## More info

- `man chmod`
- [manual](#)

## Html

We will make our own homepage today!

## Creating your homepage

Using the **Linux** konsole create a directory named `public_html` in your home directory:

```
$ cd ~
$ mkdir public_html
$ cd public_html

~/public_html$ wget http://sandbox.hlt.bme.hu/~gaebor/ea_anyag/Info1/sample.html
~/public_html$ mv sample.html index.html
```

Now we can check out our homepage at: **math.bme.hu/~<username>**. Check out the contents of index.html:

```
~/public_html$ gedit index.html
```

## Important

Every browser has a *View source* function. This is the raw HTML data of the site, when you open a webpage this is the first thing that gets downloaded to show you the site.

## Syntax

HTML is a type of XML markup language, it consists of *tags*:

```
<tag>
... content
</tag>
```

It usually isn't case sensitive, but there are many types. The basic structure of HTML:

```
<!DOCTYPE ...>
<html>
  <head>
    ... content descriptions, meta-data
  </head>
  <body>
    ... the homepage itself
  </body>
</html>
```

There are tags that don't need to be closed, e.g.:

```
<img ...>
<br/>
```

Instead of:

```
<img ...>XYZ</img>
<br> ... </br>
```

Some tags have optional *attributes*:

```
<div align="center"> ... </div>
<img width="100" ... />
```

We can provide *comments* in the source code. These don't appear on the webpage itself, but can be read through the source of the webpage.

```
<body>
  Content
  <!-- Comment -->
</body>
```

## Tasks

Before starting these tasks rename your **index.html** to **sample.html** and start with a new file named **index.html** that only includes this HTML code:

```
<!DOCTYPE html>
<html lang="http://wiki.math.bme.hu"http://wiki.math.bme.hu>
  <head>
    <meta charset="http://wiki.math.bme.huutf-8"http://wiki.math.bme.hu>
    <title>
      Title of the page
    </title>
  </head>
  <body>
    Website
  </body>
</html>
```

See the source of [this page](#) for example to start with.

1. Provide a title to your homepage e.g.: Homepage of XY (title tag).
2. Write a welcome message for your homepage (h1, h2, etc. tags).
3. Write a short CV for your homepage, it doesn't have to be perfect, but try to use nice formatting with div, p, table, list tags.
4. Write some links, maybe a list of links (ol, ul, dl these are list tags), you can write links that point to useful sites, or someone else's homepage. (a tag for links)
5. Put a picture on your homepage, it should be a picture of you, but for the sake of practice it can be anything.
6. Create another page (a separate .html file), on this page create a table with your current timetable. Provide a link on your main page (index.html) that points to this page. And another from this timetable page to the main page.
7. Validate your page with the [validator](#), green means okay, if there are some red errors, try to correct those.
8. Beautify your site, write about yourself.

## Useful links

- [validator](#)
- [HTML5 Cheat Sheet](#)
- <http://www.w3schools.com/>

[Previous](#) - [Up](#) - [Next](#)