Using GitHub Classroom

BBM103 Introduction to Programming Lab 1

Fall 2018
Signing Up to GitHub Classroom
Signing Up to GitHub Classroom

Fill textboxes and click sign in button to authorize

Click to sign up unless you have an educational account
Signing Up to GitHub Classroom

You MUST create your account with your IDs beginning with ‘b’. 
Signing Up to GitHub Classroom

There are two options. We recommend that you choose the 1st option unless you need a private repository.
Signing Up to GitHub Classroom

Open your mailbox to verify your github account.
Signing Up to GitHub Classroom

[GitHub] Please verify your email address.

Help us secure your GitHub account by verifying your email address. (hacettepe.edu.tr). This lets you access all of GitHub’s features.

Verify email address

Button not working? Paste the following link into your browser:
https://github.com/user/emails/37319780/confirm_verification/f4f871643650c9c6c66c5f3d8bb88360ce7014f

You’re receiving this email because you recently created a new GitHub account or added a new email address. If this wasn’t you, please ignore this email.

Click the link provided within the mail content
Signing Up to GitHub Classroom

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide  Start a project
Signing Up to GitHub Classroom
Joining BBM103 Classroom

Now authorize github account.
Joining BBM103 Classroom

Accept the test assignment

Accepting this assignment will give you access to the repository in the organization on GitHub.

Accept this assignment

You should accept the assignment activated to push(submit) your works.
Joining BBM103 Classroom

You are ready to go!

You may receive an invitation to join via email invitation on your behalf. No further action is necessary.

Your assignment has been created here: https://github.com
Joining BBM103 Classroom

Quick setup — if you’ve done this kind of thing before

- **Set up in Desktop** or **HTTPS**

> We recommend every repository include a `README`, `LICENSE`, and `.gitignore`.

...or create a new repository on the command line

```bash
# Create a new repository
# echo README.md | git init
# git add README.md
# git commit --amend "first commit"
# git remote add origin https://github.com/<username>/<repository>.git
# git push -u origin master
```
How to Use the Linux Command Line

BBM103 Introduction to Programming Lab I

Fall 2018
The Shell & Terminal

• **The Shell** is a program that takes commands from the keyboard and gives them to the operating system to perform.

• **Terminal Emulator** is a program that opens a window and lets you interact with the shell.
Basic Commands

- When you open a terminal emulator, by default you are in the home directory of the logged in user.
- You will see the name of the logged in user followed by the hostname.
  - $ means you are logged in as a regular user
  - # means you are logged in as root.
pwd

- **pwd** prints the full path of your current working directory.

```
[bahargezici@rdev ~]$ pwd
/home/akd/bahargezici
[bahargezici@rdev ~]$  
```
You can list all directories and files inside the current directory by using the `ls` (or `ls -l`; `ll` for listings including information such as the owner, size, date last modified and permissions) command.

```
[bahargezici@rdev ~]$ ls
204 cloud cloud.old Maildir public_html
[bahargezici@rdev ~]$ ll

```

```diff
total 20

  drwxr-xr-x.  5 bahargezici akd  4096 Oct 18 13:49 204
  drwxr-xr-x.  2 bahargezici akd  4096 Mar 10 2016 cloud
  drwxr-xr-x.  2 root root  4096 Oct  9 2016 cloud.old
  drwxr-xr-x.  9 bahargezici akd  4096 Nov 17 2016 Maildir
  drwxr-xr-x+  2 bahargezici akd  4096 Mar 10 2016 public_html

[bahargezici@rdev ~]$  ```
The `cd` command is used to change the current directory.

To change to the parent of the current directory use `cd ..`

To return directly to the home directory use a tilde as the argument:
Manipulating Files

- **cp** - copy files and directories
- **rm** - remove files and directories
- **mv** - move or rename files and directories
- **mkdir** - create directories
- **cat** - create new file, concatenate files
**cp**

- **cp** copies files and directories. In its simplest form, it copies a single file:

```
[bahargezici@rdev ~]$ ls
204 cloud cloud.old Maildir public_html pythonersleri.py
[bahargezici@rdev ~]$ cp pythonersleri.py python.py
[bahargezici@rdev ~]$ ls
204 cloud cloud.old Maildir public_html pythonersleri.py python.py
[bahargezici@rdev ~]$
```
cp (cont.)

• You can specify the full path to where you want to copy your file:

```
[bahargezici@rdev ~]$ cp pythondersler.py 204/pythondersler.py
[bahargezici@rdev ~]$ cd 204
[bahargezici@rdev 204]$ ls
BBM204-17-E-2  BBM204-17-E-4  deneme4.sh  input1  pythondersler.py  test
```

rm

If you want to delete any file or directory the command is 'rm' (for files) and 'rm -r' (for directories).
mv

- **mv** command moves or renames files and directories depending on how it is used.

```bash
[bahargezici@rdev ~]$ mv pythonersleri.py 204
[bahargezici@rdev ~]$ ls
204  cloud  cloud.old  Maildir  public_html
[bahargezici@rdev ~]$ cd 204
[bahargezici@rdev 204]$ ls
BBM204-17-B-2  deneme4.sh  pythonersleri.py  test
BBM204-17-B-4  inputl  pythonersler.py

[bahargezici@rdev 204]$ ls
BBM204-17-B-2  deneme4.sh  pythonersleri.py  test
BBM204-17-B-4  inputl  pythonersler.py
[bahargezici@rdev 204]$ mv pythonersleri.py python.py
[bahargezici@rdev 204]$ ls
BBM204-17-B-2  deneme4.sh  pythonersleri.py  test
BBM204-17-B-4  inputl  python.py
```
mkdir

• If you want to create new directories the command is `mkdir`.
cat stands for Concatenate (birleştirmek). It is used to create new file (with or without content), concatenate files and display the output of files on the standard output.
zip & unzip

- **zip** and **unzip** commands create and extract zip archive files respectively.
- The * character serves as a "wild card" for filename expansion. By itself, it matches every filename in a given directory.
• Most executable programs intended for command line use provide a formal piece of documentation called a **manual** or **man page**. A special paging program called **man** is used to view them.
**ssh**

- **ssh** (Secure Shell client) is a program for logging into a remote machine and for executing commands on a remote machine.

```
selim@DESKTOP-5HD0AAS:~$ ssh cemil@dev.cs.hacettepe.edu.tr
cemil@dev.cs.hacettepe.edu.tr's password:
```
scp

- scp allows files to be copied to, from, or between different hosts. It uses ssh for data transfer and provides the same authentication and same level of security as ssh.

**A simple example that illustrates how to send a file to dev space.**

```
scp <localfile> <username>@dev.cs.hacettepe.edu.tr:/home/ogr/b****/<directory>
```

```
selim@selim-PC:~$ scp DPSO.pdf selimy@dev.cs.hacettepe.edu.tr:/home/akd/selimy/selimy@dev.cs.hacettepe.edu.tr's password:
```
About chmod

- **chmod** is used to change the permissions of files or directories.
- Example: `chmod 777 myFile`

<table>
<thead>
<tr>
<th>#</th>
<th>Permission</th>
<th>rwx</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>read, write and execute</td>
<td>rwx</td>
</tr>
<tr>
<td>6</td>
<td>read and write</td>
<td>rw-</td>
</tr>
<tr>
<td>5</td>
<td>read and execute</td>
<td>r-x</td>
</tr>
<tr>
<td>4</td>
<td>read only</td>
<td>r--</td>
</tr>
<tr>
<td>3</td>
<td>write and execute</td>
<td>-wx</td>
</tr>
<tr>
<td>2</td>
<td>write only</td>
<td>-w-</td>
</tr>
<tr>
<td>1</td>
<td>execute only</td>
<td>--x</td>
</tr>
<tr>
<td>0</td>
<td>none</td>
<td>---</td>
</tr>
</tbody>
</table>
Exercise

• All tasks must be performed using Linux commands:
  1) Make a new directory named `playing_with_linux_cmd`
  2) Change your current working directory to the newly created one.
  3) List the contents of this directory to see that it is empty.
  4) Create a new text file `jibberish.txt` and write something funny in it before closing it.
  5) Create another new text file `README.txt` and write your life motto in it.
  6) Copy `jibberish.txt` into a text file named `wise_sayings.txt`
  7) Delete `jibberish.txt`
  8) Print out the content of `wise_sayings.txt`
  9) Create a new directory named `my_precious` and move `wise_sayings.txt` into that newly created directory. List the content of the current working directory to make sure that you have successfully moved the file.
  10) Change the permission of the file `wise_sayings.txt` to read, write and execute.
  11) Change your working directory to the parent directory of `playing_with_linux_cmd`
  12) Zip `playing_with_linux_cmd` as `gameover.zip`
  13) Use `scp` command to copy this zipped folder from your local computer to your home directory on our remote server `dev.cs.hacettepe.edu.tr`