

Git & GitHub Dictionary

- Git: a versioning system that keeps track of changes - allows you to go back, allows for collaboration, is locally stored
- GitHub: where we can access other's files, online repository for your codes (not only) Back up place for your rgit projects. cloud based located...remote

dfddf

Commands

- git init --> is going to initialize my timeline (my git rep)
- git add -->
- git commit -m "meaningful msg" --> saves the update version (snapshot/point in time) in the git repo

- **Saving a point in time**

-

```
### Where do I develop a project?
```

Conceptual areas

```
### Conceptual areas
```

1. Developing area: the working directory

2. staging area: temporary space to store file before comiting to the local repo

3. local repo:

4.

<style> </style>

Goal

- So, Github is not only for codes? Also for documents?

We will get the github course in github

- We will NOT talk about contribute to an open-source project.

Unix basic introduction

- Pwd and path
- Language is Case sensitive. Never a space in your folder or text name. Use f.ex. MilkBread or Milk_Bread or Milk-Bread
- Use tab key to auto-complete a folder or file name
- ls and cd

Introduction to Git and Github

Project Developing my own

tutorial

- Marktext:

Title

Subtitle

à this is allowing to track everyting, (with word you don't see what is going on behind the scenes) Marktext is more visual and more easy to follow. (nice visual way)

SO open marktext, you can choose to see the commands behind the text

à

@ will give you a list with all the codes

Text has .dm as extension

à We use this to be able to write a nice text, as in visual code you have to know yourself the codes if you want a nice way of writing

- Problem: work with different persons on same document. You will save it as v1, v2, V3, but at the end you will probably not know which is the final version.

- Solution: Git Repository = timeline!!! So first learn how to make a timeline, and also how to get back into time

Difference with google documents, is that you define the moments when there is a 'commit' = timepoint that you can go back. SO it is saved all the times, but not every change is a different document. With google docus

- GitHub : backup of your timeline

What is different between Git Repository and GitHub? See marktext

- This document is saved locally, but is NOT yet in the cloud!! It's not yet a timepoint in github!!

- Use VS code (Visual Studio Code)

- Check ssh keym make sure to have it public:

```
xLNycDDpFyVzKt//b2mKAAAFmV2bXlsQHBzYi52aWltdWdlbnQuYmUBAgMEBQYH
```

```
-----END OPENSSSH PRIVATE KEY-----
```

```
PS C:\Users\evmyl> cat C:\Users\evmyl.ssh
```

```
cat : Access to the path 'C:\Users\evmyl.ssh' is denied.
```

```
At line:1 char:1
```

- cat C:\Users\evmyl.ssh

```
+ CategoryInfo          : PermissionDenied: (C:\Users\evmyl.ssh:String) [Get-Content],  
UnauthorizedAccessException
```

```
+ FullyQualifiedErrorId :  
GetContentReaderUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetContentComm  
and
```

```
PS C:\Users\evmyl> cat C:\Users\evmyl.ssh\id_ed25519.pub
```

```
ssh-ed25519
```

```
AAAAC3NzaC1lZDI1NTE5AAAAIHRiGnjIIMhhpD9n3yW41P2dxLNycDDpFyVzKt//b2mK
```

```
||| --- | --- |||
```

evmyl@psb.vib-ugent.be

- Make sure to choose as language Git Bash, not PowerShell (as this is a windows things). By clicking on the arrow you can choose the other language (Git Bash).
- You need to go to your project C:\Users\evmyl\Desktop\My_first_Github_project
But in the command code you have to type it differently

You need to do with ungit

So install first, type ungit, and than you see there what is going on in the github. You see there if you created an extra timepoint or commit

Starting my timeline

- Initialize git: git init
- This means that git can see everything what is happening in the folder we initialized. He can NOT see anything what was from before.

Kdkkdkd

See slides

Saving a point in time

- This is taking a snapshot
- Git commit - m "meaningful message"
- What does meaningful means: Why, How, Effects, Limitations
- evmyl@italy22 MINGW64 ~/Desktop/My_first_Github_project (master)

\$ git add MyFirstGithubProject.md

à This creates a timepoint

- evmyl@italy22 MINGW64 ~/Desktop/My_first_Github_project (master)

\$ git commit -m "This is first timepoint" Brief description of differences between git and Github and how to get started with gi

t"

à This is how to add a comment

- Do NOT Initialize Git inside this subfolder as well!!!!

Where is GitHub

- Git is in your computer
- Github is in the cloud!!!
- It's a link between a github project in your computer

Where is git in your

computer?

Where do I develop a

project?

In your computer

Where is the timeline?

- Develop area
- Local repository
- A git repository: where you save snapshots
- Git add : to add a channel to the timeline
- So always git add, than git commit (git add is putting it in to the staging area, git commit is putting it into the local repository)

We made second timepoint, and than I got the following error.

Than I needed to follow these two commands.

Than he asks you your name and emailadress.

You need to do this for each project once.

Where is my context?

Git status

Could be:

- Verify Uncommitted changes
- Verify staged and unstaged changes
- Verify

You can see here the status.

Says that some files are not tracked yet. So then it's good to track them.

So, your two documents are added to the (repository'??) à so only from now it will be tracked. (it's on the timeline)

Now, you see that they are not yet committed.

Git add : putting them in the staging area

G

<style> </style>

Goal

- So, Github is not only for codes? Also for documents?

We will get the github course in github

- We will NOT talk about contribute to an open-source project.

Unix basic introduction

- Pwd and path
- Language is Case sensitive. Never a space in your folder or text name. Use f.ex. MilkBread or Milk_Bread or Milk-Bread
- Use tab key to auto-complete a folder or file name
- ls and cd

Introduction to Git and Github

Project Developing my own

tutorial

- Marktext:

Title

Subtitle

à this is allowing to track everyting, (with word you don't see what is going on behind the scenes) Marktext is more visual and more easy to follow. (nice visual way)

SO open marktext, you can choose to see the commands behind the text

à

@ will give you a list with all the codes

Text has .dm as extension

à We use this to be able to write a nice text, as in visual code you have to know yourself the codes if you want a nice way of writing

- Problem: work with different persons on same document. You will save it as v1, v2, V3, but at the end you will probably not know which is the final version.

- Solution: Git Repository = timeline!!! So first learn how to make a timeline, and also how to get back into time

Difference with google documents, is that you define the moments when there is a 'commit' = timepoint that you can go back. SO it is saved all the times, but not every change is a different document. With google docus

- GitHub : backup of your timeline

What is different between Git Repository and GitHub? See marktext

- This document is saved locally, but is NOT yet in the cloud!! It's not yet a timepoint in github!!

- Use VS code (Visual Studio Code)

- Check ssh keym make sure to have it public:

```
xLNycDDpFyVzKt//b2mKAAAFmV2bXlsQHBzYi52aWltdWdlbnQuYmUBAgMEBQYH
```

```
-----END OPENSSSH PRIVATE KEY-----
```

```
PS C:\Users\evmyl> cat C:\Users\evmyl.ssh
```

```
cat : Access to the path 'C:\Users\evmyl.ssh' is denied.
```

```
At line:1 char:1
```

- cat C:\Users\evmyl.ssh

```
+ CategoryInfo          : PermissionDenied: (C:\Users\evmyl.ssh:String) [Get-Content],  
UnauthorizedAccessException
```

```
+ FullyQualifiedErrorId :  
GetContentReaderUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetContentComm  
and
```

```
PS C:\Users\evmyl> cat C:\Users\evmyl.ssh\id_ed25519.pub
```

```
ssh-ed25519
```

```
AAAAC3NzaC1lZDI1NTE5AAAAIHRiGnjIIMhhpD9n3yW41P2dxLNycDDpFyVzKt//b2mK
```

```
||| --- | --- |||
```

evmyl@psb.vib-ugent.be

- Make sure to choose as language Git Bash, not PowerShell (as this is a windows things). By clicking on the arrow you can choose the other language (Git Bash).
- You need to go to your project C:\Users\evmyl\Desktop\My_first_Github_project
But in the command code you have to type it differently

You need to do with ungit

So install first, type ungit, and than you see there what is going on in the github. You see there if you created an extra timepoint or commit

Starting my timeline

- Initialize git: git init

This means that git can see everything what is happening in the folder we initialized. He can NOT see anything what was from before.

Kdkkd

See slides

Saving a point in time

- This is taking a snapshot
- `Git commit -m "meaningful message"`
- What does meaningful means: Why, How, Effects, Limitations
- `evmyl@italy22 MINGW64 ~/Desktop/My_first_Github_project (master)`

```
$ git add MyFirstGithubProject.md
```

à This creates a timepoint

- `evmyl@italy22 MINGW64 ~/Desktop/My_first_Github_project (master)`

```
$ git commit -m "This is first timepoint"
```

Brief description of differences between git and Github
and how to get started with gi

t"

à This is how to add a comment

- Do NOT Initialize Git inside this subfolder as well!!!!

Where is GitHub

- Git is in your computer
- Github is in the cloud!!!
- It's a link between a github project in your computer

Where is git in your

computer?

Where do I develop a

project?

In your computer

Where is the timeline?

- Develop area
- Local repository
- A git repository: where you save snapshots
- Git add : to add a channel to the timeline
- So always git add, than git commit (git add is putting it in to the staging area, git commit is putting it into the local repository)

We made second timepoint, and than I got the following error.

Than I needed to follow these two commands.

Than he asks you your name and emailadress.

You need to do this for each project once.

Where is my context?

Git status

Could be:

- Verify Uncommitted changes
- Verify staged and unstaged changes
- Verify

You can see here the status.

Says that some files are not tracked yet. So than it's good to track them.

So, your two documents are added to the (repository'??) à so only from now it will be tracked. (it's on the timeline)

Now, you see that they are not yet committed.

Git add : putting them in the staging area

Git commit: putting them in the local repository

à you can compare this with library. All books that come in get classified, they go in the box with f.ex. country books, romans, ... This is staging area!!! (this is a stage where you can structure it)

Bringing the books from the box to the shelves. This is commit.

So, you should do always all two of them.

Now they are also committed. With one command both files are done!!

Git status allow me to check what files are

See notes commit: putting them in the local repository

→ you can compare this with library. All books that come in get classified, they go in the box with f.ex. country books, romans, ... This is staging area!!! (this is a stage where you can structure it)

Bringing the books from the box to the shelves. This is commit.

So, you should do always all two of them.

Now they are also committed. With one command both files are done!!

Git status allow me to check what files are

See notes

blablab

```
@'git diff' "git diff"
```

```
'git log'
```

```
git log
```