

Inhoud

Git en GitHub Tutorial voor Beginners	2
Set up Git en Github.....	2
Git instellen Username, Email en default main branch	2
Werken met extra branches	3
Initialiseren van repository.....	3
Status	4
Tracked	4
Commit.....	4
Github Repository	5
Create github repository	5
Quick set up	5
Delete online repository.....	6
Verbind je online repository met je lokale repository	6
Push	6
Pull	6
Clonen	7
Quick steps	7
Commands op een rij	8

Git & Github

Git en GitHub Tutorial voor Beginners

<https://www.youtube.com/watch?v=tRZGeaHPoaw&t=1505s>

Set up Git en Github

Download git: <https://git-scm.com/downloads>

Maak account op: <https://github.com>

Git instellen Username, Email en default main branch

Open je git bash: (Druk op je start knop in windows links onder en zoek naar git bash)

Eerst een paar eenmalige instellingen instellen

Stel hier je gebruikersnaam in

User name: git config --global user.name "Jan Jansen"

Stel hier je email in

Email: git config --global user.email JanJansen@hotmail.com

Stel hier je default branch in (main)

Branch: git config --global init.default branch main

Een branch is als een aparte werkruimte waar je wijzigingen kunt aanbrengen zonder je hoofdversie te beïnvloeden.

Werken met extra branches

Als je een branch genaamd main hebt aangemaakt dan zou je een 2^e branch kunnen aanmaken.

Je 2^e branch is eigenlijk een kopie van je main branch waarin je kunt werken of wijzigingen in kunt aanbrengen.

Als alles goed werkt kan je deze mergen (intergreren) met je main branch.

Een extra branch maken

2^e branch: git branch 2eBranch

Kijken hoeveel branches je hebt

Controlleer branches: git branch

Nu wordt de lijst met branches getoond.

De branch met de * ervoor is de branch die actief is en waarin je nu aan het werk bent.

Wisselen tussen branches: git switch 2eBranch

Je aangepaste branch in je main branch mergen (intergreren)

Zorg eerst dat je main branch actief is door met git switch naar je main branch te wisselen.

Merge 2ebranch met main branch: git merge -m "Typ hier een bericht" 2ebranch

In de commando hier boven geef je dus de naam van de branch waarmee je je main branch wilt mergen.

Initialiseren van repository

Om een repository van een map te maken ga je eerst in windows naar de betreffende map.

Klik 1 keer op de map met je linker muis knop en dan met je rechter muisknop.

In de sub menu klik op kopiëren als pad.

Of je kunt dubbel klikken op de map, en als je in de map bent de pad in de url kopiëren.

Dan ga je naar je git bash en typ: cd "pad naar je map"

Repository maken: git init

In je map is nu een verborgen map .git gemaakt, in windows kun je die zien door in je map te gaan, boven in je scherm op weergaven klikken en dan onder in de drop down menu weer op weergaven klikken en dan in die drop down menu onder aan op verborgen items klikken.

Status

Nu dat er een repository aangemaakt is kan je in je git bash de status van je repository zien.

Status controlleren: git status

Na het uitvoeren van deze commando krijg je de status van de repository te zien d.m.v. een lijst welke bestanden binnen de repository niet gevolgd (untracked) worden.

Tracked

Om deze bestanden te volgen kun je de volgende commando uitvoeren.

Track 1 untracked bestand : git add naam_bestand

Track alle untracked bestanden: git add --all

Untrack 1 bestand : git rm --cached naam_bestand

Untrack alle bestanden: git rm --cached --all

Als je je files volgt dan kan je met git status checken of dat er geen untracked files meer zijn.

Commit

Na dat je al je bestanden volgt moet je een commit maken.

Een commit is een soort van opslaan, je slaat je wijzigingen op.

Commit je repository: git commit -m "Typ hier een bericht"

In een log kan je zien welke commits er in het verleden zijn gemaakt.

Inzien log : git log

Inzien log in makkelijke overzicht: git log --online

Je krijgt de commit te zien met een hash code, daaronder de author en daaronder de date.

Als er iets in de toekomst fout gaat en je wilt weer terug gaan naar een oudere commit, dan heb je de hash code nodig van die commit.

Na de commit kan je de inhoudt van je git repository naar je online github repository sturen.

Github Repository

Create github repository

Als je nog geen github repository hebt aangemaakt kan je naar je github page en inloggen. Klik links boven op de knop **new**.

- Vul in de repository naam
- Vul in beschrijving (optioneel).
- Kies: -Public (Voor iedereen toegankelijk)
-Private (Alleen toegankelijk voor mensen die jij toegang geeft)
- Add readme file (Optioneel)
- Add git ignore (Optioneel)
- Choose license (Optioneel)

Klik nu op creat repository.

Je repository wordt nu gemaakt, je ziet nu een quick start schema met daarin een paar git commands wat je makkelijk kan kopiëren en gebruiken.

Quick set up

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

Set up in Desktop or HTTPS | SSH <https://github.com/digitaldutch1/test.git>

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

```
echo "# test" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/digitaldutch1/test.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/digitaldutch1/test.git
git branch -M main
git push -u origin main
```

Om bestanden van je git repo naar je github repo moet je ze eerst met elkaar verbinden.

Verbind je repositories (eerste keer):

```
git remote add origin https://github.com/digitaldutch1/test.git
git branch -M main
```

Nu is er verbinding en kan je beginnen met het versturen van de inhoud.

Van je git op je pc naar de online repository is **Push**, van je online repository naar je git op je pc versturen is **Pull**

Delete online repository

Ga naar je hoofd pagina en klik op de repository dat je wilt verwijderen.

Klik op de navigatie bar op settings.

Scroll naar beneden, daar vindt je de optie om de repository te verwijderen

Verbind je online repository met je lokale repository

In je Git bash als je in je lokale repository zit

`git remote add origin <naam van je repository>`

Bijvoorbeeld: `git remote add origin https://github.com/gebruikersnaam/naam-repo.git`

Wil je de verbinding verbreken

`git remote remove origin <naam van je repository>`

Bijvoorbeeld: `git remote remove origin https://github.com/gebruikersnaam/naam-repo.git`

Push

Nu dat je repositories met elkaar verbonden zijn kan je de inhoud van je git naar je online github pushen.

hiermee push je de inhoud van je git repository main branch naar je online github repository.

Verstuur (eerste keer) maar github : `git push -u origin main`

Daarna kan je gebruikmaken van : `git push origin main`

Push al je branches : `git push --all`

Pull

Je kan ook de inhoud van je online repository naar je repository op je pc sturen.

Pull naar je pc: `git pull origin main`

Als je op een nieuwe apparaat voor het eerst je repository met je online repository wilt synchroniseren moet je als eerst clonen, als dat gedaan is kan je de keren daarna pull gebruiken

Clonen

Je kan ook een kloon (kopie) van je github online repository maken en op je pc zetten. Er zijn twee opties, je kan een kloon maken en die map wordt in je lokale repository gezet die actief is.

Kloon naar lokale repository: git clone <https://github.com/digitaldutch1/test.git>

Je kan ook een kloon maken en die map wordt in een aangegeven map gezet.

Kloon naar aangewezen map:

git clone <https://github.com/digitaldutch1/test.git> naam_map

Quick steps

- Stel user name, user email en default branch in (eerste keer)
- Initieer repository (eerste keer)
- Controleer status
- Track alle untracked bestanden
- Commit
- Push naar github

Commands op een rij

- **User name:** git config --global user.name "Jan Jansen" (Eenmalig)
- **Email:** git config --global user.email JanJansen@hotmail.com (Eenmalig)
- **Default branch:** git config --global init.default branch main (Eenmalig)
- **Extra branch:** git branch 2eBranch
- **Controlleer branches:** git branch
- **Wisselen tussen branches:** git switch 2eBranch
- **Merge 2ebranch met main branch:** git merge -m "Typ hier een bericht" 2ebranch
- **Repository maken (git):** git init
- **Status controlleren:** git status
- **Track 1 untracked bestand** : git add naam_bestand
- **Track alle untracked bestanden:** git add --all
- **Untrack 1 bestand** : git rm --cached naam_bestand
- **Untrack alle bestanden:** git rm --cached --all
- **Commit je repository:** git commit -m "Typ hier een bericht"
- **Inzien log** : git log
- **Inzien log in makkelijke overzicht:** git log --online
- **Verbind je repositories (eerste keer):**
 - git remote add origin <https://github.com/digitaldutch1/test.git>
 - git branch -M main
- **Verstuur (eerste keer) naar github** : git push -u origin main
- **Daarna kan je gebruikmaken van** : git push origin main
- **Push al je branches** : git push --all
- **Pull naar je pc:** git pull origin main
- **Kloon naar lokale repository:** git clone <https://github.com/digitaldutch1/test.git>
- **Kloon naar aangewezen map:**
 - git clone <https://github.com/digitaldutch1/test.git> naam_map