

topicus
ACADEMY



TA-110

Introductie GIT

Studiegids



TA110 Introductie GIT

Topicus Academy

Inhoudsopgave

Inleiding	1
1. Online tutorial van Github	2
1.1. Video uitleg over GIT	2
1.2. Tutorial 1	2
1.3. Tutorial 2	2
2. De GIT prompt	4
2.1. Command line / DOS-box / Terminal	4
2.2. GIT Shell	4
2.3. GIT commandos	5

Inleiding

GIT wordt gebruikt om teams van ontwikkelaars samen te laten werken in één project zonder elkaar te veel in de weg te zitten. Deze cursus leert je de basisbegrippen van GIT te kennen en te begrijpen.

Na deze cursus kan je samen met je buddy code toevoegen aan een project, en die delen met de rest van het team, en de wijzigingen van het team weer lokaal op jouw laptop krijgen.

Er is een heleboel te lezen over GIT, en dat kan je prima zelf oppakken na deze cursus als verdieping.

Wees niet ongerust als je na deze cursus niet alles precies meer weet. In het dagelijks ontwikkelwerk maak je zoveel gebruik van GIT dat je dat automatisch allemaal weer oppakt en je kennis snel uitbreidt.

Chapter 1. Online tutorial van Github

Bij Github (het bedrijf dat leeft van het succes van Git, en al onze programmacode voor ons beheert) hebben ze veel trainingsmateriaal klaar staan voor beginnende ontwikkelaars om met Git te leren omgaan.

1.1. Video uitleg over GIT

Voor een uitleg over GIT, wat het is, en wat je ermee kan doen kan je deze video bekijken: [Webcast - The Basics of Git and GitHub - July 2013](#). Deze video duurt ongeveer 30 minuten.



Het is handig om ook de *Git cheat sheet* te downloaden en af te drukken voor naast je toetsenbord. Je kan deze vinden op deze URL: <https://education.github.com/git-cheat-sheet-education.pdf>.

Na deze video kan je zelf aan de slag met de online tutorials van Github. Voor veel van deze tutorials hoef je geen software te installeren, maar je gaat in de nabije toekomst (lees tutorial 2) een github account nodig hebben. Hoe je aan zo'n account komt en welke eisen er voor gelden staat in de *Cobra in Action* handleiding: *Je eerste dag*.

Om de eerste stappen met Git te doorlopen begin je bij tutorial 1 van Github.

1.2. Tutorial 1

Werk de Github tutorial door die je kan vinden op het volgende adres. Voor deze tutorial heb je waarschijnlijk een account nodig.

- <https://lab.github.com/githubtraining/introduction-to-github>

Je kan dit zelfstandig doen.

1.3. Tutorial 2

Werk de Github tutorial door die je kan vinden op het volgende adres:

- <https://guides.github.com/activities/hello-world/>

Deze introductie zou ongeveer 2 uur duren. Je kan dit zelfstandig doen.

In deze tutorial leer je:

- het maken en gebruiken van een *git-repository*
- het starten en beheren van een *branch*
- het maken van wijzigingen in een bestand en deze te *pushen* naar github
- het openen en mergen van een *pull request*

Om deze tutorial te kunnen doorlopen moet je wel een Github account hebben. Zie de handleiding

voor je eerste dag wat je daarvoor moet doen.

Na deze tutorial heb je in je persoonlijke github profiel een project met de naam "hello-world". Dit project gaan we in de volgende module gebruiken om met de *git prompt* te werken.

Chapter 2. De GIT prompt

Met de *git-prompt* kan je alle functionaliteit van git ontsluiten. In deze module leer je de basiscommando's waarmee je volwaardig kan meedraaien in een project.

Zo leer je een project naar je lokale machine te halen (`git clone`), te werken met github (`git remote`) en je wijzigingen te delen met de rest van het ontwikkelteam (`git add`, `git commit` en `git push`). Hiertoe leer je werken op de command line, ook bekend als DOS-box, of terminal venster.

2.1. Command line / DOS-box / Terminal

Bovenstaande termen bedoelen hetzelfde: een venster waarmee je met tekstuele commando's je computer kan besturen.

Vroeger, voor de Macintosh, waren er alleen maar computers met tekstuele interfaces (op computers bij research instituten na) en moest alles met de hand gebeuren:

- het laden van een programma van floppy
- het starten van de programma om een muiscursor te krijgen (je had tekstuele schermen met menubalken, knoppen en dergelijke)
- het starten van spelletjes

Tegenwoordig kan je deze zaken nog steeds (alhoewel je geen programma voor muisbesturing hoeft te laden op moderne systemen).

Met name op UNIX achtige systemen (Linux/macOS) is het gebruik van de terminal vrij normaal.



Wanneer je een productieserver wil beheren maak je vaak verbinding via een beveiligd kanaal met het commando `ssh`. Dit staat voor *secure shell*. Als je bijvoorbeeld `ssh productieserver` uitvoert in een terminal venster zal er een verbinding gemaakt worden met de productieserver en wordt op die server een lokale terminal gestart met een *shell*. Je kan dan op die productieserver commando's uitvoeren alsof je toetsenbord direct aan die machine vast zit, ook al staat die server in de wereldstad Hengelo.

Op Windows systemen is het landschap van terminals enigszins verwarrend. De meest gebruikte is de DOS box (of prompt). Daarnaast heeft Windows tegenwoordig een zogeheten PowerShell. Ook zie je dat bijvoorbeeld GIT een eigen shell omgeving bijlevert voor het kunnen uitvoeren van GIT commandos.

2.2. GIT Shell

Op Windows systemen werk je gedurende deze module in de GIT shell, op Linux of macOS gewoon in de terminal.

Onderstaand voorbeeld voert het commando `git status` uit in de terminal.

```
$ git status ①
```

```
fatal: Not a git repository (or any of the parent directories): .git ②
```

① Het commando dat je moet intypen

② Het resultaat van het commando

Waar je in deze voorbeelden op moet letten is dat je niet het dollar teken overneemt. Dit dollar teken wordt in terminals gebruikt als de plek waar je een commando kan invoeren: de **command prompt**.

Nu komen we op het punt dat je GIT commandos kan invoeren.

2.3. GIT commandos

De volgende commandos van GIT komen aan bod:

2.3.1. Commandos voor het werken met je repository

- clone
- remote
- fetch
- branch
- tag

2.3.2. Commandos voor werken met bestanden

- status
- checkout
- add
- rm
- commit