

Frontend & Backend leerlijn

# Workshop Git

---

**Deel 2**

# Sessie 1

## 15:00 – 16:10

- Huiswerk clonen en pushen naar eigen remote
- Feature branch maken
- Pull Request klaarzetten en *checken!*

# 1. Clonen en een repository maken

1. Clone deze nep-huiswerkopdracht naar jouw locale machine door via jouw IDE te kiezen voor:

*VCS > Get from version control > [URL plakken].*

<https://github.com/hogeschoolnovi/git-workshop-fake-homework>

2. Maak daarna alvast een **nieuwe repository** aan via GitHub.com, waar je jouw toekomstige huiswerk straks naartoe kunt gaan pushen.

## 2. Nieuwe repo koppelen

Huiswerk is altijd een **clone** van een bestaand project. Voordat je jouw eigen repo kunt koppelen, moet je daarom de referentie naar de huiswerk-repo eerst verwijderen met het volgende commando:

```
git remote remove origin
```

Daarna kun je de stappen op GitHub volgen om main te pushen:

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

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



## 2. Nieuwe repo koppelen

Huiswerk is altijd een **clone** van een bestaand project. Voordat je jouw eigen repo kunt koppelen, moet je daarom de referentie naar de huiswerk-repo eerst verwijderen met het volgende commando:

```
git remote remove origin
```

Daarna kun je de stappen op GitHub

**Vraag:** Waarom moet de *main* branch altijd minimaal 1 keer gepusht zijn als je met feature branches gaat werken?

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

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



# 3. Lokaal een branch aanmaken

Je maakt een **nieuwe branch** aan met het commando:

```
git checkout -b naam-van-branch
```

Maak vervolgens het “huiswerk”. Experimenteer door meerdere commits te maken (deze zijn allemaal nog lokaal)

Ben je klaar? Dan push je de *gehele feature branch* met alle commits erop naar de remote repository met:

```
git push origin naam-van-branch
```

# 4. PR maken en checken

Wanneer je een feature branch hebt gepusht zal de terminal altijd vragen of je een PR wil maken via een link. Maar op GitHub krijg je ook altijd een pop-up:



Volg de stappen in GitHub en typ een leuk berichtje voor de nakijker!

## Check daarna altijd of je PR klopt

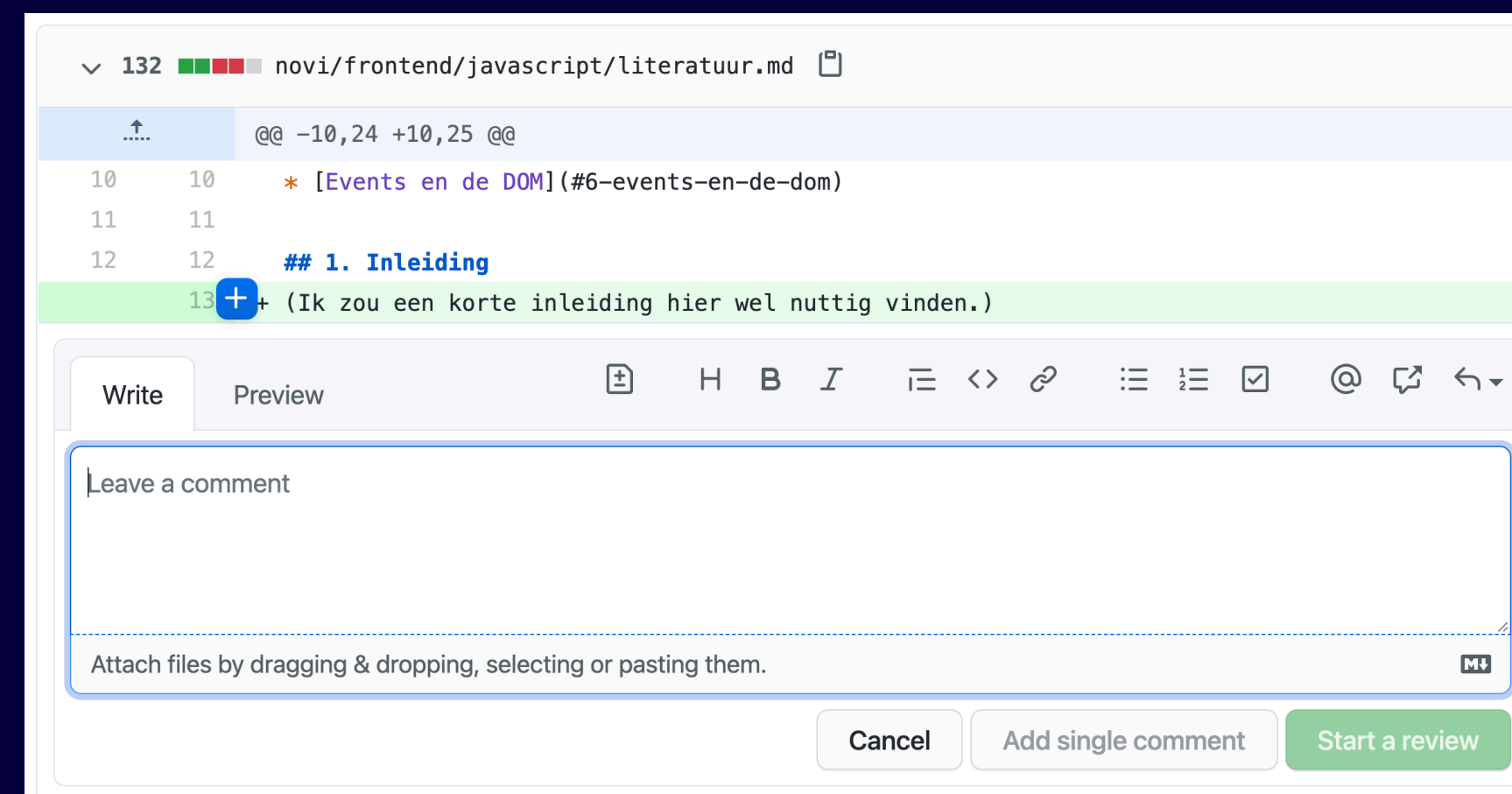
1. Zie je al jouw commits staan?
2. Zie jij alle bestanden staan die je hebt aangepast?

**Wissel nu jouw PR uit  
met een studiegenoot!**



# Feedback geven op Pull Request

Wissel de link naar jouw PR uit met een medestudent. Klik het tabje 'Files' aan en bekijk de veranderingen. De regels waar je feedback op hebt klik je aan, en daar typ je een bericht.



Als je klaar bent, druk je rechtsboven op **Finish review**. Kies bij deze workshop voor **Request changes**. Het huiswerk is nu nagekeken!

# Sessie 3

**16:20 – 16:50**

- Feedback verwerken
- Remote mergen
- Lokaal up date maken

# Eigen feedback verwerken

Jij hebt inmiddels ook feedback gekregen op jouw huiswerk en je ziet dat er nog het een en ander aangepast moet worden!

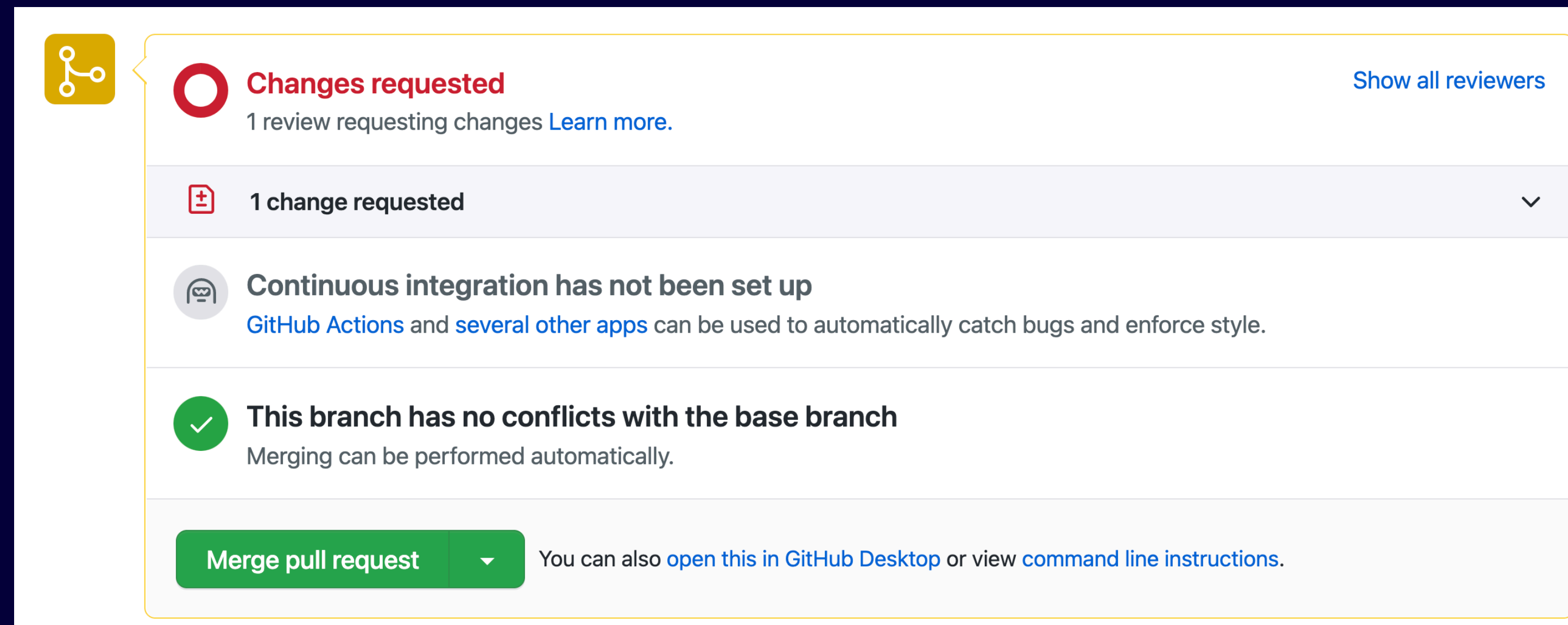
Pas lokaal wat dingen aan op de feature branch. Maak hier een *commit* van en *push* opnieuw **de hele branch**:

```
git push origin naam-van-branch
```

Je zult zien dat het commentaar in de PR nu “outdated” is, want jij hebt alle feedback verwerkt.

# Remote mergen

Merge de remote feature branch nu naar master via GitHub:



Als je wil, mag je de remote feature branch ook verwijderen!

# Lokaal up to date maken

De remote master is geüpdate, **maar daar weet jouw lokale project nog niets vanaf**. Ga eerst lokaal naar jouw hoofd-branch:

```
git checkout main
```

Haal dan de “nieuwe” master binnen van de remote:

```
git pull origin main
```

Verwijder nu jouw oude, lokale feature branch:

```
git branch -d naam-branch
```

**Terug naar de  
plenaire sessie!**