

Opdracht 2: Git repository lokaal maken en gebruiken!

Een **repository** (repo) is het basiselement van Git, alles start hiermee. Je kunt van elke Windows folder/map een **repository** maken.

De opdracht is om een Git repository via Visual Studio Code (VSC) te maken en te gebruiken.

Opdracht tijdslimiet: ~25 minuten

Stappenplan

1. VSC open maken

Doelstelling is om een lege folder genaamd **git-opdracht-2** te maken op de D: schijf in de folder D:\uwv-robotframework-cursus\dagdeel-4 en die in VSC als folder open te maken. Je kunt kiezen om de nieuwe folder via Windows Verkenner (This PC) te maken of via VSC.

Hierbij de stappen om dit via VSC te doen:

- Klik links boven op **File > Open Folder...**
- Klik in de opengaande window op **This PC >** dubbelklik op de **(D:) DATA** schijf knop
- Navigeer naar de folder uwv-robotframework-cursus\dagdeel-4\
- Klik op de **New Folder** knop en type de naam **git-opdracht-2** druk op enter
- Klik op **Select Folder** knop om de nieuwe folder open te maken
- Klik op de blauwe knop **Yes, I trust the authors**

Nu heb je een folder gemaakt en die in VSC opengemaakt.

Indien je **screenshots** wilt zien van de stappen, klik dan hieronder op de link:

► Screenshots stap 1

2. Git repository aanmaken

Momenteel heb je een project folder gemaakt en in VSC opengemaakt.

Echter is het nog geen Git repository, hiervoor moet je Git nog initialiseren.

Hierbij de stappen om dit via VSC te doen:

- Klik op de **Source Control** knop



- Klik op de **Initialize Repository** knop

Initialize Repository

VSC heeft onderwater de Git command `git init` uitgevoerd

3. Files toevoegen, stagen en committen

Er is nu een Git repository gemaakt. Laten wij de repository vullen met enkele files, stagen en committen zodat Git ze kan bijhouden.

Hierbij de stappen om dit via VSC te doen:

3.1. File README.md maken

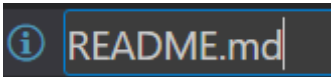
- Klik op de **Explorer** knop



- Klik op de **New File...** knop



- Typ `README.md` als file naam



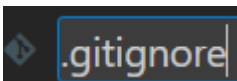
- Copy of typ `### Cursus dagdeel 4: Basis Git` als eerste regel in de **README.md** file en bewaar dit (save)

3.2. File .gitignore maken

- Klik op de **New File...** knop



- Typ `.gitignore` als file naam



- Copy of `**typ *.log` in de **.gitignore** file en bewaar dit (save)

3.3. Files stagen

- Klik op de **Source Control...** knop

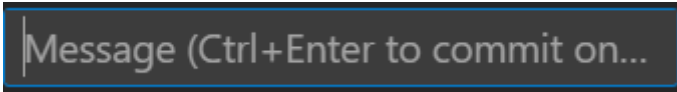


- Klik op de **Stage Changes** knop



3.4. Files committen

- Copy of typ `Eerste commit direct op main` als commit bericht



- Klik op de **Commit** knop



4. Branch maken en mergen

In de vorige stap hebben wij een **commit** met 2 files direct op de **main** branch gedaan.

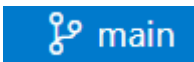
Dit is **afgeraden**. Laten wij daarom een **nieuwe branch** maken vanaf de **main** branch en daar verder werken en aan het einde die weer terug mergen met de **main** branch.

Laten wij na de commit, kijken naar de verschillen tussen de feature en main branch.

Hierbij de stappen om dit via VSC te doen:

4.1. Branch maken

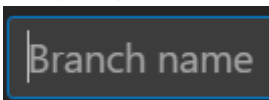
- Klik beneden links op de **View Branches** knop



- Klik op de **+ Create new branch...** knop



- Copy of typ `feature/jira-001/robot-file` als branch naam en druk op Enter



4.2. Bestaande files aanpassen, nieuwe browsertest.robot file maken

Nu dat wij een aparte branch hebben gemaakt, laten we enkele wijzigingen doorvoeren in een reeds bestaande file en een nieuwe Robot Framework file toevoegen.

- Copy of typ Tweede regel toegevoed op branch feature/jira-001/robot-file als tweede regel in de bestaande **README.md** file en bewaar dit (save)
- Klik op de **Explorer** knop



- Klik op de **New File...** knop



- Typ `browsertest.robot` als file naam



- Copy of typ onderstaande test in de `browsertest.robot` file en bewaar dit (save)

*** Settings ***

Library Browser

*** Test Cases ***

Testgeval

 New Browser headless=False

 New Page url=https://duckduckgo.org

 Sleep time_=5

4.3. Wijzigingen stagen en committen

Wij zijn klaar met onze werk op de feature branch, laten wij onze wijzigingen veilig stellen door deze te stagen en hierna te committen.

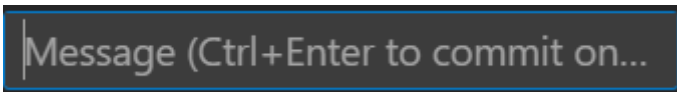
- Klik op de **Source Control...** knop



- Klik op de **Stage All Changes** (+ teken) knop bij kopje **Changes**



- Copy of typ Eerste commit op feature branch als commit bericht



- Klik op de **Commit** knop



Door de - teken te gebruiken, kun je files un-stagen. Die je niet wilt committen.

4.4. Ga heen en weer tussen de feature en main branch

Wij zijn klaar om te mergen met de **main** branch, maar voor dat wij dit gaan doen. Is het handig om visueel (in Windows Verkenner en/of VSC) te zien dat een branch een parallele versie is van je werk.

Dit kan door te switchen tussen de main en feature branch, in ons geval.

- Klik beneden links op de **View Branches** knop



- Klik op de **main** branch en bekijk het verschil

```
🔗 feature/jira-001/robot-file bd58240a
🔗 main 925d687d
```

- Klik beneden links op de **View Branches** knop



- Klik op de **feature/...** branch en bekijk het verschil weer

```
🔗 feature/jira-001/robot-file bd58240a
🔗 main 925d687d
```

Zie je dat de `browsertest.robot` file komt en weg gaat?

Zie je dat de inhoud van de `README.md` file veranderd?

4.5. Merge feature met de main branch

Laten wij onze werk die wij op de feature branch hebben gemaakt, **mergen** met de **main** branch. Om de **feature** branch met de **main** branch te mergen, moeten we ervoor zorgen om dit proces van **vanaf** de **main** branch te starten.

- Klik beneden links op de **View Branches** knop



- Klik op de **main** branch knop

```
🔗 feature/jira-001/robot-file bd58240a
🔗 main 925d687d
```

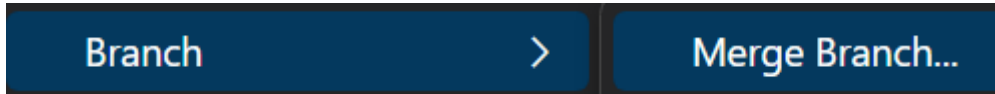
- Klik op de **Source Control...** knop



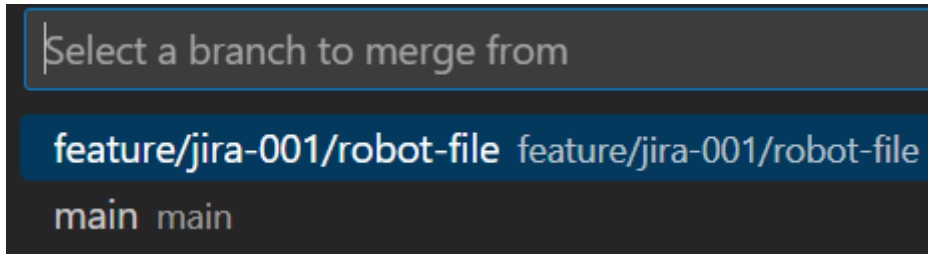
- Klik op de **Views and More Actions** knop



- Klik op de **Merge Branch...** knop



- Klik op de **feature/...** branch knop



Zie je dat de verschillen nu weg zijn?

Als je heen en weer gaat tussen de main en feature branch!

5. Optioneel

- Verwijder de **feature** branch
- Maak een nieuwe file **app.log** direct op de **main** branch en bekijk dat die file onzichtbaar blijft voor Git. Weet jij waarom, dit zo is?

Deze opdracht is **klaar**.