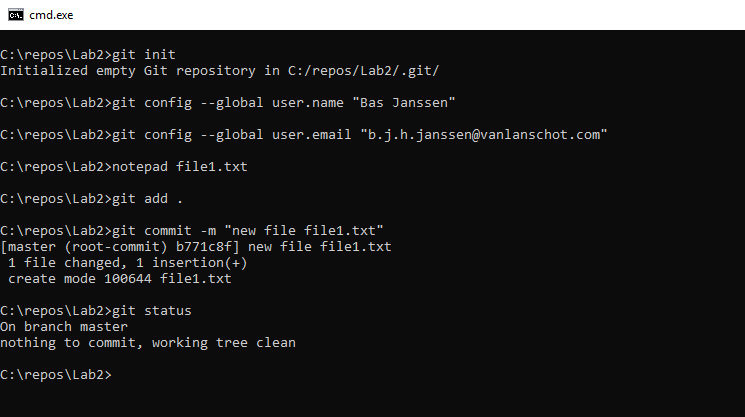
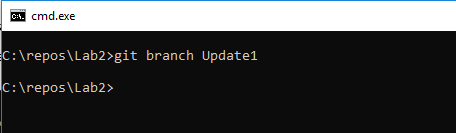
Lab1 : Branches

Voor deze lab en alle andere labs is het noodzakelijk de git client software geinstalleerd te hebben.

* Maak een sub folder Lab2 aan in C:\repos en geef de volgende commando’s :

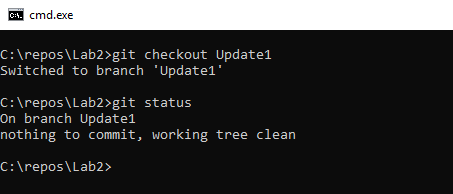


* + Het resultaat is een eerste commit met file1.txt in het snapshot
* Maak nu een **branch** aan met : git branch Update1 ( Update1 is de naam ) :

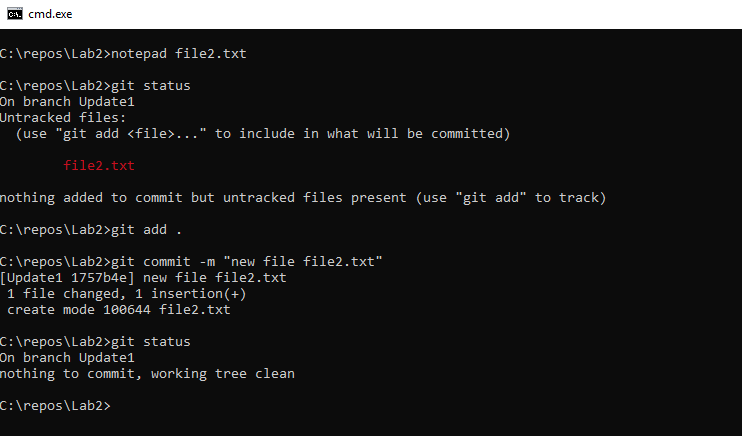


* Checkout branch Update1 : git checkout Update1

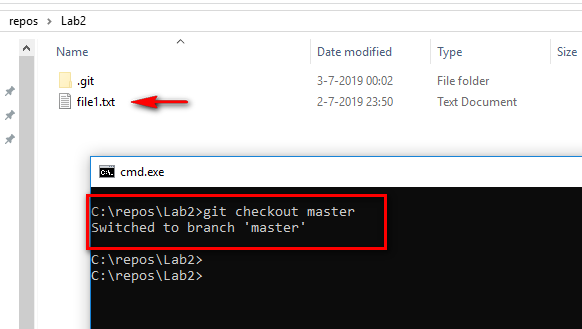
En check de status met : git status :



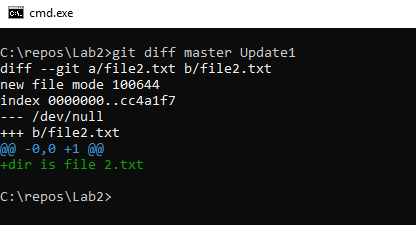
* Maak nu file2.txt aan en geef : git add . en : git commit -m “new file file2.txt »



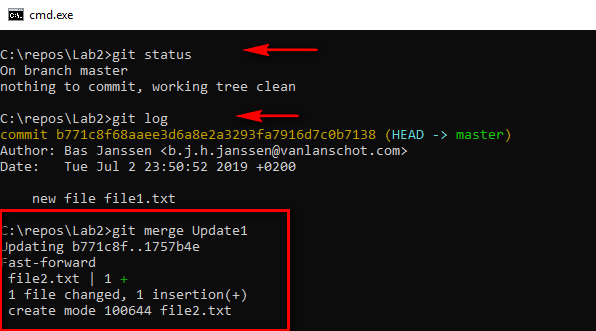
* + Status is nu dat we in branch Update1 en nieuwe file hebben toegevoegd. Dus Update1 kent nu 2 files , branch master kent alleen nog file1.txt
* Geef : git checkout branch master en kijk in de windows explorer :



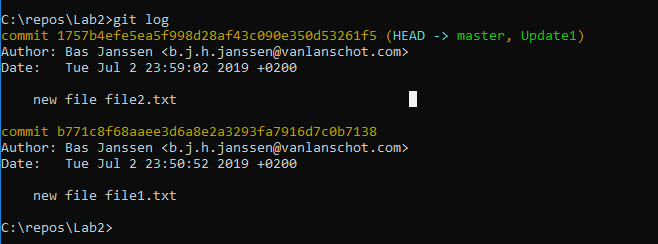
* + Je ziet nu alleen file1.txt omdat alleen deze file in de laatste commit van de master branch zit.
* Geef : git diff master Update1



* + Hier is het verschil tussen beide branches zichtbaar. Branch Update1 ( b ) heeft een extra file : file2.txt met de inhoud in het groen.
* Geeft nu : git merge Update1

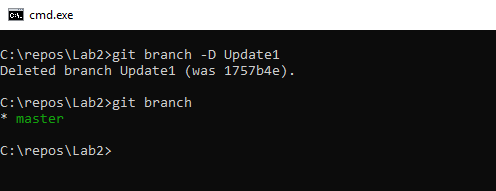


* + Het resultaat is dat in de master branch , de Update1 branch wordt samengevoegd met alle wijziginen :



* + Het valt hier op dat HEAD nu zowel de laatste commit is van master als van Update1
* We kunnen nu branch Update1 verwijderen omdat alle wijzigingen zijn doorgevoerd :

Dit doen we met : git branch -D Update1 :



* + Git branch : laat zien dat er nog maar 1 branch is : master
  + Met het juiste command is het mogelijk de merge en de delete van de Update1 branch in één keer uit te voeren.

Het resutaat is nu een folder Lab2 met twee files waarvan we file2.txt via een branch hebben toegevoegd aan de master branch.

Einde Lab2