

Git Konflikte

1. Bilden Sie zweier Gruppen.
2. Einigen Sie sich wessen Repository Sie verwenden möchten
3. Nutzen Sie ggf. `git clone` damit beide das gleiche Repository ausgecheckt haben
4. Nutzen Sie beide `git pull` um den aktuellsten Stand zu erhalten

Aktualisieren der Readme

Führen Sie Bitte folgende Schritte jeweils einzeln an ihrem Rechner aus:

1. Fügen Sie jeweils eine Sektion "**### Erwartungen an das Studium**" in die readme.md hinzu.
2. Erstellen Sie mehrere Zeilen mit Erwartungen, die Sie an ihr Studium haben. Beginnen Sie jede Zeile mit einem `"* "`. Den Stern interpretiert Markdown später als Bulletpoint.
3. Fügen Sie die neue Datei dem Staging Bereich hinzu
4. Comitten Sie Ihre Änderung, notieren Sie sich Commit Id und Message.

Aktualisieren Pushen der Änderungen

Machen Sie diesen Teil der Aufgabe bitte gemeinsam.

1. Pushen Sie die Änderung von einem Ihrer Rechner.
2. Wenn Sie nun die Änderung des 2. Rechners versuchen zu pushen erhalten Sie wieder eine Fehlermeldung das Sie Ihre Änderung nicht auf der letzten Version durchgeführt haben.
3. Führen Sie ein `git fetch` durch

```
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From github.com:PatrikSteuer/myfirstrepo
   774da85..1b72bf0  master    -> sshorigin/master
```

1. Führen Sie ein `git rebase` durch

```
rebase in progress; onto 1b72bf0
You are currently rebasing branch 'master' on '1b72bf0'.
  (fix conflicts and then run "git rebase --continue")
  (use "git rebase --skip" to skip this patch)
  (use "git rebase --abort" to check out the original branch)

Unmerged paths:
  (use "git reset HEAD <file>..." to unstage)
  (use "git add <file>..." to mark resolution)
   both modified:   Readme.md
```

1. Schauen Sie sich die Readme.md an, was stellen Sie fest? Was bedeuten die Zeichen?
2. Lösen Sie den Konflikt in der Readme.md auf.
3. Fügen Sie die Readme.md dem Staging Bereich hinzu
4. Führen Sie `git rebase --continue` aus

9. Pushen Sie ihre Änderung

Betrachten Sie die Änderung

1. Nutzen Sie `git log --graph` um die Historie ihrer Commits zu betrachten.
2. Vergleichen Sie die Commit Id's mit den zuvor notierten Id's. Was fällt Ihnen auf?
3. Betrachten Sie Ihre Änderung auf Github.com

Aktualisieren Sie die Readme

Wiederholen Sie die Übung. Fügen Sie eine Sektion mit "Erwartungen an die Vorlesung hinzu". Anstelle eines `git rebase` machen Sie bitte ein `git merge`. Wie unterscheiden sich die Ergebnisse?