# Lage ny lokal repository:

git init

# Lage ny repository på github:

Trykk på pluss knappen og velg New repository og skriv inn.

# Klone en repository fra github:

git clone https://github.com/andreas1992/hello-world.git

Lager en ny mappe med navnet hello-world

Kan også lage en ny mappe med eget navn.

# Legge til globalt navn:

git config --global user.name "Andreas"

# Legge til global epost:

git config --global user.email "aks1992@gmail.com"

# Sjekke at de faktisk ble lagt inn:

git config user.name/email

# Sjekke status på current branch (i den mappen den ligger i):

git status

Origin = the remote repository

Master = the current branch

# De ulike statusene til filer i git mappen:

untracked: i rødt.

En fil som ikke har blitt "staged"

Staging betyr, dette er en fil vi har sagt til git at vi

ønsker å committe i neste comit.

tracked file: i grønt

En fil som har blitt "staged"

Den har ikke blitt committed enda.

# Legge til en fil til "staging":

git add <navnet på filen.filtype>

git add \* for å inkludere alle filene

git add <folder name> for å inkludere en mappe

Får en tracked file som er klar til å bli committed.

# Committe en fil (lastes ikke opp enda):

git commit -m "<brukbar forklaring på hva den gjør>"

-m = message

Får opp en oversikt over denne committen, med endringer osv

i filen

Får også opp current branch og git checksum.

insertation viser hvor mange endringer i filen man har.

# Hva som skjer når man endrer på en fil som er committed:

Får opp beskjed om at vi har en modifisert fil.

Modifiserte filer er ikke lagt til vår neste committ

Bruk git add, med \* hvis du vil for å legge den til i staged

# Hvordan se forskjellen mellom working directory og staging area?

git diff

# Hvordan se alle commits som er gjort:

git log

git log -2 Viser kun de to siste

git log --pretty=oneline

Viser alle commits på linjer rett under hverandre

# Hvordan pushe til repository? (til github feks)

git push origin master

origin = remote repository

master = current branch

(git push TO origin, and we want to push the master branch)

med brukernavn og passord

kan også pushe en branch, slik at origin kan se igjennom og godkjenne

# Hvordan sjekke om det er blitt gjort noen endringer på origin?

git fetch

# Hvordan pulle fra repository? (github feks)

git pull

Gir oss detaljer over alt som har hendt.

Puller ned alle endringer til maskinen.

Endrede filer vil nå være endret også lokalt.

# Hvordan branche?

git branch <nytt navn>

git branch <nytt navn> -b sender deg automatisk til den nye branch.

Ender ikke opp i den branchen enda, fortsatt i master branch.

Master er default

# Hvordan skifte til den nye branchen manuellt?

git checkout <navnet på branch>

# Hvorfor vises ikke den nye branchen i github?

git push origin <navn på ny branch(ikke master)>

Fordi endringene er ikke postet til github enda, man må huske å

gå et steg videre fra å committe, altså pushe den nye branchen.

# Hvordan se alle åpne branches?

git branch

# Hvordan switche tilbake til master branch?

git checkout master

# Hvordan 'merge' en branch med en annen?

git merge <navnet på branch du vil merge(med den du er på nå)>

# Hvordan slette en branch?

git branch -d <navnet på branchen>

Slettes kun lokalt, ikke på github

# Hvordan slette en branch fra github?

git push origin --delete <navnet på branchen>

# Hvordan branche fra en spesifikk commit?

git branch branchname <sha1-of-commit(hash-key)>

# Hvordan unstage endringer i staging area?

git reset (HEAD) filename

# Hvordan resette til en tidligere commit i commit history?

git reset <SHA>

# Hvordan lage en ny repository i kommandolinjen?

git init

git commit -m "first commit"

git remote add origin https://github.com/andreas1992/<name>.git

git push -u origin master

# Hvordan laste opp en allerede eksisterende repository til github?

git remote add origin https://github.com/andreas1992/<name>.git

git push -u origin master

# Hvordan slette en fil fra git tracking OG mappen?

git rm <filnavn>

Deretter committe endringen

# Hvordan spesifisere at git ikke skal adde visse filer osv?

cat .gitignore

\*.class # ignorerer alle .class filer

build/ # ignorer alle filer i build/ mappen

# Hvordan lage filen først?

touch .gitignore

# Hvordan lage .gitignore fil i windows?

1. Lage gitignore.txt

2. Legg inn regler i text editor

3. Hold inne SHIFT, høyretrykk på mappen, velg Open Commandline

4. Endre navn i kommandolinjen, ren gitignore.txt .gitignore

# Hvordan skippe staging area?

git commit -a -m