GIT - cheat sheet

Install (Windows)

https://git-scm.com/

https://desktop.github.com/

set up new repo

(create git repo in web gui first)

```
git init
git add .gitignore
touch README.md
git add README.md
git remote add myOrigin git@github.com:maxbis/examengesprekken
git push -u origin master
```

ssh key

```
add ~/.ssh/id_rsa.pub in git hub web gui
```

add file

```
git add <file name>
```

update repo

```
git commit -a -m "comment"
git push
```

Get updated files

```
git pull
```

Overwrite local changes

```
git reset --hard
git pull
```

New clone - install new

```
git clone https://github.com/maxbis/examengesprekken

// for Ubuntu server
sudo chgrp -R www-data examengesprekken
```

Go to older version

```
git reset --hard 0ad5a7a6.....
```

Git clear cache (if ignore file is changed)

```
git rm --cached config
```

List all tracked files

```
git ls-tree -r HEAD --name-only
```

Git issues with .ignore

```
git rm -r --cached .
git add .
git commit -m "Stop tracking files that are now ignored"
```

Init new repo

```
cd path/to/class-cloud
git init
git add .
git commit -m "Initial commit"

<create new repo on githib>

git remote add origin https://github.com/maxbis/class-cloud.git
git push -u origin master

<on Linux>
git clone https://github.com/maxbis/class-cloud.git
```

Set Remote Git && Check git remote

SSH key

ssh-keygen -t ed25519 -C "your_email@example.com"

eval "$(ssh-agent -s)"
ssh-add ~/.ssh/id_ed25519

cat ~/.ssh/id_ed25519.pub

```
git remote set-url origin git@github.com:USERNAME/REPO.git
git remote -v
```

Test Connection

```
ssh -T git@github.com
```

Set up local repo (git), what?

Create git repository on own server and auto deploy.

Init new git repo NEW

In /srv/repo

```
mkdir NEW; cd NEW
git init --bare --initial-branch=main /srv/git/NEW
```

Init local repo NEW

```
git remote add origin ssh://max@qool.ovh:1611/srv/git/NEW
git push -u origin main
```

Clone

local

```
  git clone /srv/git/NEW /var/www/qool/NEW
```

remote

```
git clone ssh://max@qool.ovh:1611/srv/git/deployment /var/www/NEW
```

Auto deploy upon push

```
vi /srv/git/deployment/hooks/post-receive
```

post_receive script

```
  #!/bin/bash

  set -euo pipefail

  TARGET="/var/www/test"
  BRANCH="main"
  GIT_DIR="/srv/git/deployment"

  while read -r oldrev newrev refname; do
    if [ "$refname" = "refs/heads/$BRANCH" ]; then
      echo "Deploying $BRANCH to $TARGET..."
      git --git-dir="$GIT_DIR" --work-tree="$TARGET" checkout -f "$BRANCH"
```

```
      # optional: composer install, cache clear, reload apache, etc.
      # cd "$TARGET" && composer install --no-dev
      # sudo systemctl reload apache2
    fi
  done
```

--

Revision #23
Created 2020-07-07 07:36:08 UTC by Max
Updated 2026-08-12 18:50:39 UTC by Max