# Setup guide to work in OpenAlea from Rhizodep model

## Python version

Python 3.9.15 installation with anaconda install

## IDE

Pycharm Community edition ou VisualCode

## OpenAlea

Creating new environment and installing selected packages with the following command in shell :

*conda create -n <env-name> -c openalea3 -c conda-forge openalea.plantgl openalea.mtg*

Adding a new OpenAlea package to <env>:

*conda activate <env>*

*conda install -c openalea3 -c conda-forge openalea.<packagename>*

## Project

New project in pycharm -> existing environnement -> add interpreter -> search named interpreter

## Git

IDE project folder associating right environment is also the git folder to exchange with remote servers (Github, Gitlab).

Server upstream

Your server

Other developer’s server

Local server

main

Edition

*add then commit*

*push*

*pull*

*fork*

*clone*

*pull*

*request*

*PR*

*fork*

* Go to Github/Lab repertory

*cd <path>*

To go back

*cd..*

To list files in directory

*dir* or *ls*

* Remote servers management

Check connected remotes

*git remote -v*

Add remote. Usually name *origin* is used for main project remote.

*git remote add <remotename> HTTPS*

Remove remote

*git remote rm <remotename>*

To push to multiple remotes (Github and Gitlab), we can create an all remote from one HTTPS and then specify :

*git remote set-url --add --push all HTTPS1*

*git remote set-url --add --push all HTTPS2*

Then

*git push all <branchname>*

* Branches management

Check local active branch

*git branch*

Change local branch focus

*git checkout <branchname>*

Add new local branch

git checkout -b <branchname>

Remove local branch

git branch –-delete <branchname>

Push new local branch to create branch on remote

*git push –set-upstream <remotename> <branchname>*

* In-branch management

Add changes before commit to local server

*git add .*

To check accounted files for commit (not excluded by .gitignore)

*git status*

Commit to local server

*git commit -m “label”*

Push commit to remote server branch

*git push <remotename> <branchname>*

…if for some initial reasons local branch and remote branch have different names or are not explicitly related

*git push <remotename> <localbranch>:<serverbranch>*

Update local active branch

*git pull <remotename> <branchname>*