

# Setting up a development pyxnat

The latest release of `pyxnat` in the [Python Package Index \(pypi\)](#) has a few bugs and missing features that get in the way of following the HCP-pyxnat tutorial. I strongly recommend using a [Python virtual environment](#) with [our fork of pyxnat](#), which also includes some customizations for ConnectomeDB and the OHBM Hackathon.

To set up the virtual environment on Linux or Mac OS X:

1. Make sure you have `virtualenv` in your python installation: look [here](#) for details.
2. Create the virtual environment.

## Creating a Python virtual environment

```
$ mkdir ~/ohbm-hcp
$ cd ~/ohbm-hcp
$ virtualenv pyxnat-env
```

3. Source the virtual environment's activate script; you'll need to do this for any shell where you're using this virtual environment. The exact commands needed depend on which shell you use: the most common choices are `bash` (or `sh`, the original Bourne shell) or `csh`.

## Selecting the virtual environment (bash/sh)

```
$ . pyxnat-env/bin/activate
(pyxnat-env) $
```

## Selecting the virtual environment (csh)

```
% source pyxnat-env/bin/activate.csh
[pyxnat-env] %
```

Note that your shell prompt now includes the name of the virtual environment (`pyxnat-env`).

4. Install `pyxnat` into the virtual environment:

## Installing pyxnat

```
(pyxnat-env) $ pip install git+git://github.com/Human-Connectome-Project/pyxnat@hcp-db#egg=pyxnat
```

*Aside for Python experts: since I work on pyxnat as well as with pyxnat, I usually clone the pyxnat source tree and run `python setup.py develop` instead. Unless you anticipate making changes to pyxnat itself, the above sequence seems less complicated.*

Now that you have the virtual environment set up, you can `cd` to whatever directory you need; your `PATH` has been changed so that you'll be running the Python executable from the virtual environment, with the packages you've installed. If you rely on other non-standard-library Python packages, you'll need to install those into this virtual environment as well.

Remember you must source the activate script in every shell where you intend to use `pyxnat`. The easiest way to verify this is, before you run `python`, to make sure that the virtual environment name (`pyxnat-env`) appears in the shell prompt.