# fMRIprep through docker in Windows/macOS

**Step 1: A valid BIDS dataset**

fMRIprep will only work on a dataset in BIDS format. Please check Ineke’s github here <https://github.com/InekePillet/InekeP/tree/master/BIDS%20code/BIDSify> and the BIDS documentation for further information on how to achieve this: <https://bids-specification.readthedocs.io/en/latest/02-common-principles.html>. Also, always check the dataset with BIDS validator here <https://bids-standard.github.io/bids-validator/>.

**Step 2: The fMRIprep documentation on installation**

<https://fmriprep.org/en/stable/installation.html>

<https://fmriprep.org/en/stable/docker.html>

Note that the documentation covers multiple installation methods. We are interested in the installation through docker, and more specifically the fmriprep-wrapper.

**Step 3: Install docker for Windows or macOS**

If you have some time and are interested, look up an intro about docker for data science.

<https://docs.docker.com/get-docker/>

Follow the installation instruction provided throughout the installation process on this website.

You might have to make an account once docker is installed on your computer and to be able to run fmriprep.

You can check that docker is properly installed by opening the terminal window (PowerShell on Windows should do the trick) and entering:

docker run --rm hello-world

You should get the following output:

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.

1. The Docker daemon pulled the "hello-world" image from the Docker Hub.

(amd64)

1. The Docker daemon created a new container from that image which runs the

executable that produces the output you are currently reading.

1. The Docker daemon streamed that output to the Docker client, which sent it

to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:

https://hub.docker.com/

For more examples and ideas, visit:

https://docs.docker.com/get-started/

**Step 4: Install fmriprep-wrapper**

For the fmriprep-wrapper to work you need to have Python installed on your computer. Get version 3.7 or up. If you do not already have it, make sure to install it before installing the wrapper.

You can get python here <https://www.python.org/>

Or use anaconda <https://www.anaconda.com/products/individual>

Once Python is installed, open up the terminal window (PowerShell in Windows) and enter:

py -m pip install --user --upgrade fmriprep-docker

Note, if you have an older version of Python the command may be:

Python -m pip install --user --upgrade fmriprep-docker

This should initiate a download sequence.

You will probably get a very important warning: where fmriprep-docker is installed, and that it is not on the current PATH. It should always be on the PATH if you want to be able to run it. Two options: you add it to the PATH every time you open a terminal window, or you make it part of a profile file that is loaded automatically whenever you open a terminal.

Option one: add to PATH manually every time

On Windows:

Every time you open a Powershell, type $env:Path += ";YourLocationOfFmriprepDocker" Just change the ‘YourLocationOfFmriprepDocker’ with the pathway to where your fmriprep docker was installed. !!! Leave the ; sign in front of it!!! e.g. $env:Path += ";C:\Program Files\GnuWin32\bin"

On macOS:

To see what’s on the path, type echo $PATH

To add to the PATH the location to your fmriprep docker, type export PATH=$PATH:/YourLocationOfFmriprepDocker e.g. export PATH=$PATH:/Users/inekepillet/.local/bin

!!! On macOS, although after installation there may be a message that the path is something like ./.local/lib/etcetera, the path that you need to add should end with ‘/.local/bin’. Find this folder on your mac, go to your computer, your Macintosh disk, go to Users, then press command+dot+shift all at once, then you should be able to see the .local folder and inside a bin folder. Right click the bin folder, press option and then click ‘copy as pathname’. This will be the path you need to add and will end with ‘/.local/bin’.

Option two: load into the PATH automatically every time

On Windows:

Open Powershell

Type $profile

This should give a file location to your profile that gets loaded in automatically. It might not exist, the command below will tell you that. If it does not exist, type: New-Item $profile -Type File -Force

Type notepad $profile

Put in this line, just change the ‘YourLocationOfFmriprepDocker’ with the pathway to where your fmriprep docker was installed. !!! Leave the ; sign in front of it!!! $env:Path += ";YourLocationOfFmriprepDocker" e.g. $env:Path += ";C:\Program Files\GnuWin32\bin"

On macOS:

Type open ~/.bash\_profile

! If you do not work with a bash shell, this profile is probably not loaded in automatically whenever you start up the terminal.

Change this so you use bash automatically in terminal: Go to terminal preferences and to ‘shell opens with’, check ‘command (complete path)’ and type there ‘/bin/bash’. And/or you can also type chsh -s /bin/bash.

If you do not want to change it to bash, you probably know a little bit about shell. Check what shell you are using (maybe you already know it), find the profile for whatever shell you’re using (google should be able to tell you the location of the file), and then open the file and add the fmriprep docker location to the PATH.

In the text editor; if you don’t know how it works it may be safest to add the lines of code either at the very start or very end of the file; add the following line: export PATH=$PATH:/YourLocationOfFmriprepDocker e.g. export PATH=$PATH:/Users/inekepillet/.local/bin

!!! On macOS, although after installation there may be a message that the path is something like ./.local/lib/etcetera, the path that you need to add should end with ‘/.local/bin’. Find this folder on your mac, go to your computer, your Macintosh disk, go to Users, then press command+dot+shift all at once, then you should be able to see the .local folder and inside a bin folder. Right click the bin folder, press option and then click ‘copy as pathname’. This will be the path you need to add and will end with ‘/.local/bin’.

🡪 If everything goes well, you should now be able to use fmriprep-wrapper to run fmriprep using the fmriprep-docker command.

**Step 5: fMRIprep code**

The basics of fMRIprep code consist of the following:

* You need the command that will run fmriprep through docker: fmriprep-docker
* This command is followed by the input directory (aka your BIDS folder) and then the output directory (the derivatives folder inside your BIDS folder would be a good place)
* You will also need to indicate which participants to run fmriprep for (even you want to run it for all, you need to provide the subject ids)
  + participant --participant-label sub-01
* You will additionally need to indicate where your freesurfer license is located, as fMRIprep makes use of freesurfer
  + If you do not have a license, get one here: <https://surfer.nmr.mgh.harvard.edu/registration.html>
  + To indicate location of the license, you add the following text to the command:

--fs-license-file YourLocationOfLicense

* You might want to add the output-spaces argument, where you can decide what kind of files you get. By default you get normalized files (MNI space), but maybe you want the images preprocessed and coregistered, but not normalized. Or both. Or the images preprocessed but not coregistered, …
  + --output-spaces MNI152NLin2009cAsym T1w func fsaverage fsnative

What does this look like put together?

fmriprep-docker M:\SUB01inBIDS M:\SUB01inBIDS\derivatives participant --participant-label sub-01 --output-spaces MNI152NLin2009cAsym T1w func fsaverage fsnative --fs-license-file C:\Users\r0733658\Documents\BIDS\_tutorial\license.txt

More generally:

fmriprep-docker inputdirectory outputdirectory participant --participant-label label –output-spaces spaces --fs-license-file licensefilelocation

You can pass additional commands to indicate what you want your preprocessing to look like, for instance:

[-h] [--version]

[--participant\_label PARTICIPANT\_LABEL [PARTICIPANT\_LABEL ...]]

[-t TASK\_ID] [--debug] [--nthreads NTHREADS]

[--omp-nthreads OMP\_NTHREADS] [--mem\_mb MEM\_MB] [--low-mem]

[--use-plugin USE\_PLUGIN] [--anat-only] [--boilerplate]

[--ignore-aroma-denoising-errors] [-v]

[--ignore {fieldmaps,slicetiming,sbref} [{fieldmaps,slicetiming,sbref} ...]]

[--longitudinal] [--t2s-coreg] [--bold2t1w-dof {6,9,12}]

[--output-space {T1w,template,fsnative,fsaverage,fsaverage6,fsaverage5} [{T1w,template,fsnative,fsaverage,fsaverage6,fsaverage5} ...]]

[--force-bbr] [--force-no-bbr]

[--template {MNI152NLin2009cAsym}]

[--output-grid-reference OUTPUT\_GRID\_REFERENCE]

[--template-resampling-grid TEMPLATE\_RESAMPLING\_GRID]

[--medial-surface-nan] [--use-aroma]

[--aroma-melodic-dimensionality AROMA\_MELODIC\_DIMENSIONALITY]

[--skull-strip-template {OASIS,NKI}]

[--skull-strip-fixed-seed] [--fmap-bspline] [--fmap-no-demean]

[--use-syn-sdc] [--force-syn] [--fs-license-file PATH]

[--no-submm-recon] [--cifti-output | --fs-no-reconall]

[-w WORK\_DIR] [--resource-monitor] [--reports-only]

[--run-uuid RUN\_UUID] [--write-graph] [--stop-on-first-crash]

[--notrack]

bids\_dir output\_dir {participant}

For more information on which options exist and what their commands are, please check the fMRIprep documentation for your version of fMRIprep:

<https://fmriprep.org/en/stable/>

Another code example using an extra argument (using the defaults except slice timing correction):

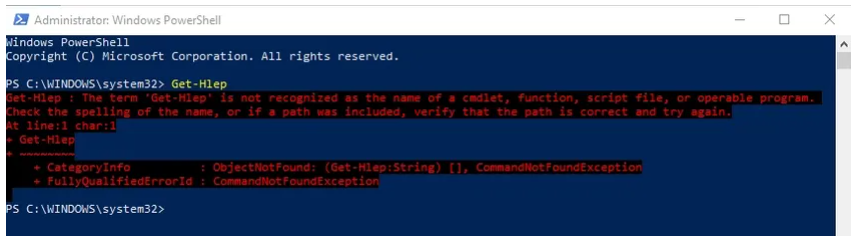
fmriprep-docker M:\SUB01inBIDS M:\SUB01inBIDS\derivatives participant --participant-label 01 --fs-license-file C:\Users\r0733658\Documents\BIDS\_tutorial\license.txt --ignore slicetiming

**Problems I encountered and how to solve them:**

Note: If you encounter problems first make sure docker is running and you are logged in!

For me, fmriprep was successfully installed but when I tried to use it I would get a message saying that fmriprep-docker is not a known command.

Here’s an example of what such an error would look like in Windows PowerShell:



The reason for this error was because the location in which fmriprep was installed was not on the path. Go back to the installation step and see the explanation there.

Other possible problems:

* If fmriprep-docker command functions properly but you still get problems this might be related to the command itself
  + The BIDS dataset could be not BIDS-compliant, check BIDS validator and BIDS documentation to solve it
  + Order matters, for instance putting the participants command before the directories will give you an error
  + When you get such an error it indicates that this is the problem, you can check the documentation and the links below for help should that occur

**Useful links:**

<https://rpubs.com/sarenseeley/bids-fmriprep-mriqc>

<https://www.youtube.com/watch?v=J0npRWV2zTY>

<https://reproducibility.stanford.edu/fmriprep-tutorial-running-the-docker-image/>

<https://andysbrainbook.readthedocs.io/en/latest/OpenScience/OS/fMRIPrep_Demo_2_RunningAnalysis.html>

<https://github.com/nipreps/fmriprep>