Jupyterhub Documentation (old Version)

Jupyter Notebook is a web-based application for interactive work on Jupyter Notebook documents. These documents can combine source code, the source code's output, and further text and images.

The system is accessible via a web browser. The Jupyter Notebook software connects a notebook document with an R or Python interpreter to allow interactive execution of the source code.

The software comes with preinstalled interpreters and a set of basic libraries for Python 2.7, Python 3.6, and R 3.6 in a multiuser configuration called Jupyterhub. These environments are named "Anaconda-Python2.7", "Anaconda-Python3.6", and "R".

Users cannot modify this basic set of libraries. Instead, users can create own environments for Python 2.7 or Python 3.6, which are stored in the user's home folder, and add additional libraries therein. For R, users can extend the basic library set, by installing additional libraries in a local package repository, which also stored in the user's home folder.

A central instance of the Jupyter Notebook Server is available under https://jupyterhub.awi.de for all users with a valid AWI account. Personal instances of Jupyterhub are available for users or workgroups on request. They have a limited life span, after which they are deleted automatically. Own environments, as described before, remain, as they are stored in the user's home folder, and will be available in all instances of Jupyterhub at AWI.

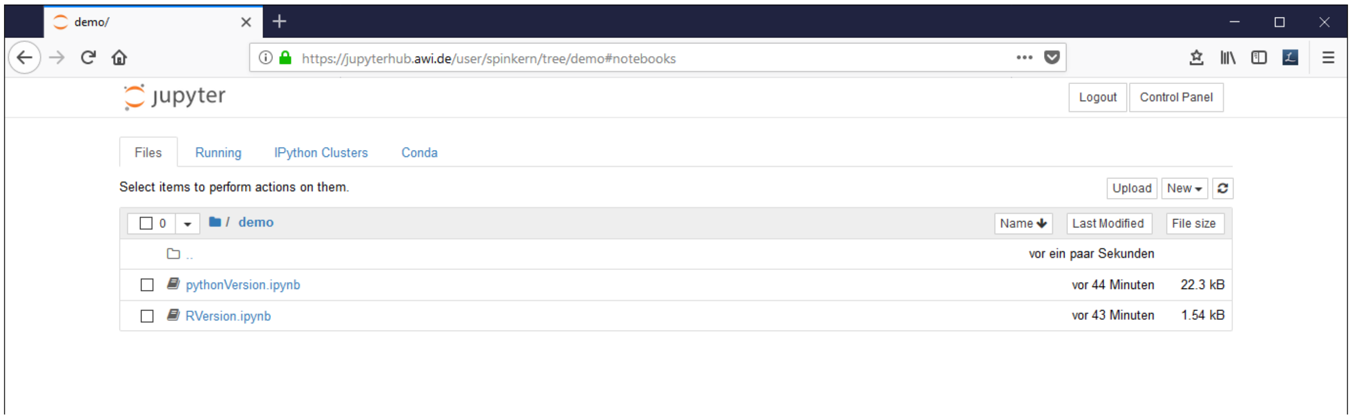
Working with the Jupyterhub System

The system is used by a web interface in the browser. After login, the interface offers four tabs, described below: "Files", "Running", "Clusters", and "Conda".

The "Files" tab

This tab shows a folder listing of the user's home folder and offers a simple file browser including up- and download functionality.

It is not possible to navigate outside the own home folder. To access data on the central storage system, this needs to be linked (see description below).



Select items to perform actions on them.

/ demo

..	vor ein paar Sekunden	
pythonVersion.ipynb	vor 44 Minuten	22.3 kB
RVersion.ipynb	vor 43 Minuten	1.54 kB

Notebook documents

To start a new notebook document click on "New" and select a preinstalled environment.

All input in a notebook document is organized in cells, of which two types exist:

Code cells contain (Python or R) source code and can be executed interactively using the selected kernel. The output is shown below the cell.

Markdown cells contain formatted text. The formatted code replaces the markdown code when the cell is executed.

A cell can be executed by clicking the "Run" button or by the shortcut "CTRL" + "Enter".

Each code cell gets a sequential number to indicate the order in which the cells have been executed.

The active cell has a colored bar at the left side indicating whether it is in "editing" mode (green) or "command" mode (blue). The modes are switched using the "Esc" and "Enter" key respectively.

Several shortcuts are predefined, e.g. in the command mode:

- Up/down keys: scroll up and down the cells
- A: Create a new cell above the selected one
- B: Create a new cell below the selected one
- M: Treat the active cell as a markdown cell
- R: Raw cell (like a code cell, but won't be executed)
- Numbers 1-6 Format cell to heading
- Y: Treat the active cell as a code cell
- DD: Delete the active cell
- Z: undo a cell delete
- Shift + M: Merge active and following cell

In edit mode:

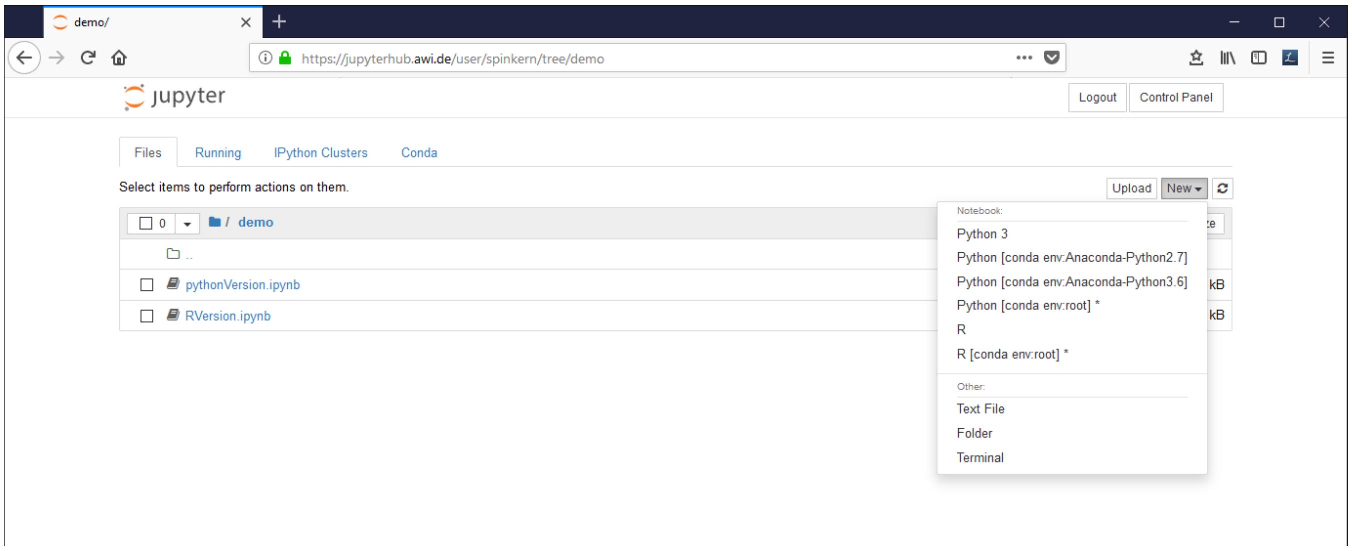
- CTRL + Enter: Execute the cell
- CTRL + SHIFT + "-" (minus): Split the cell at the cursor position

A notebook document needs a connected kernel to be executable. The active kernel is shown in the upper right corner. To change the kernel, click "kernel" à "change kernel" in the menu bar, and select the desired kernel.

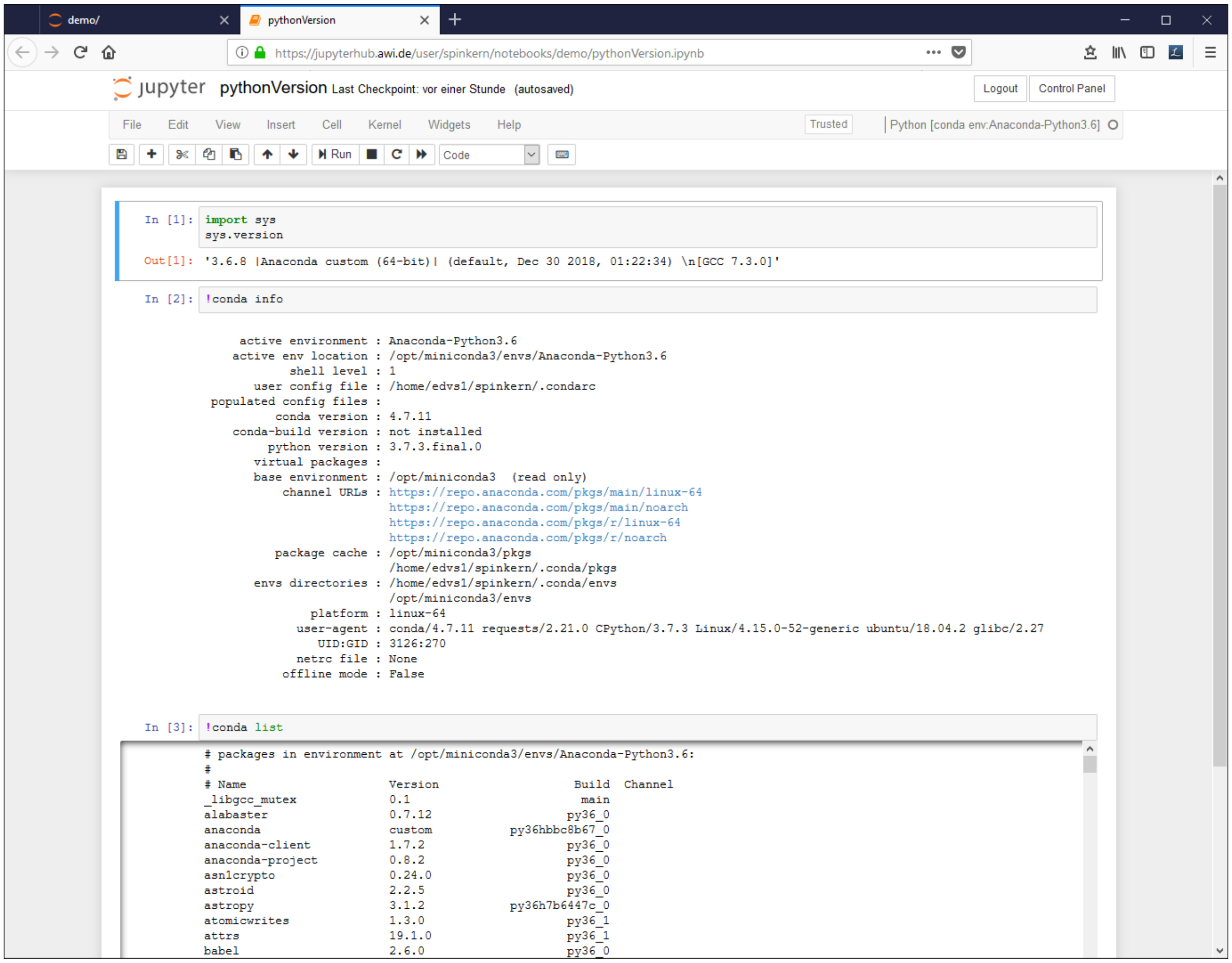
Preinstalled kernels are:

- Python [conda env:Anaconda-Python2.7] : Python 2.7 and default libraries (Anaconda)
- Python [conda env:Anaconda-Python3.6] : Python 3.6 and default libraries (Anaconda)
- R : R 3.6 and default libraries

These kernels are read-only! This means users cannot add additional libraries to the kernels. If further libraries are needed, a new environment for own libraries has to be created first.

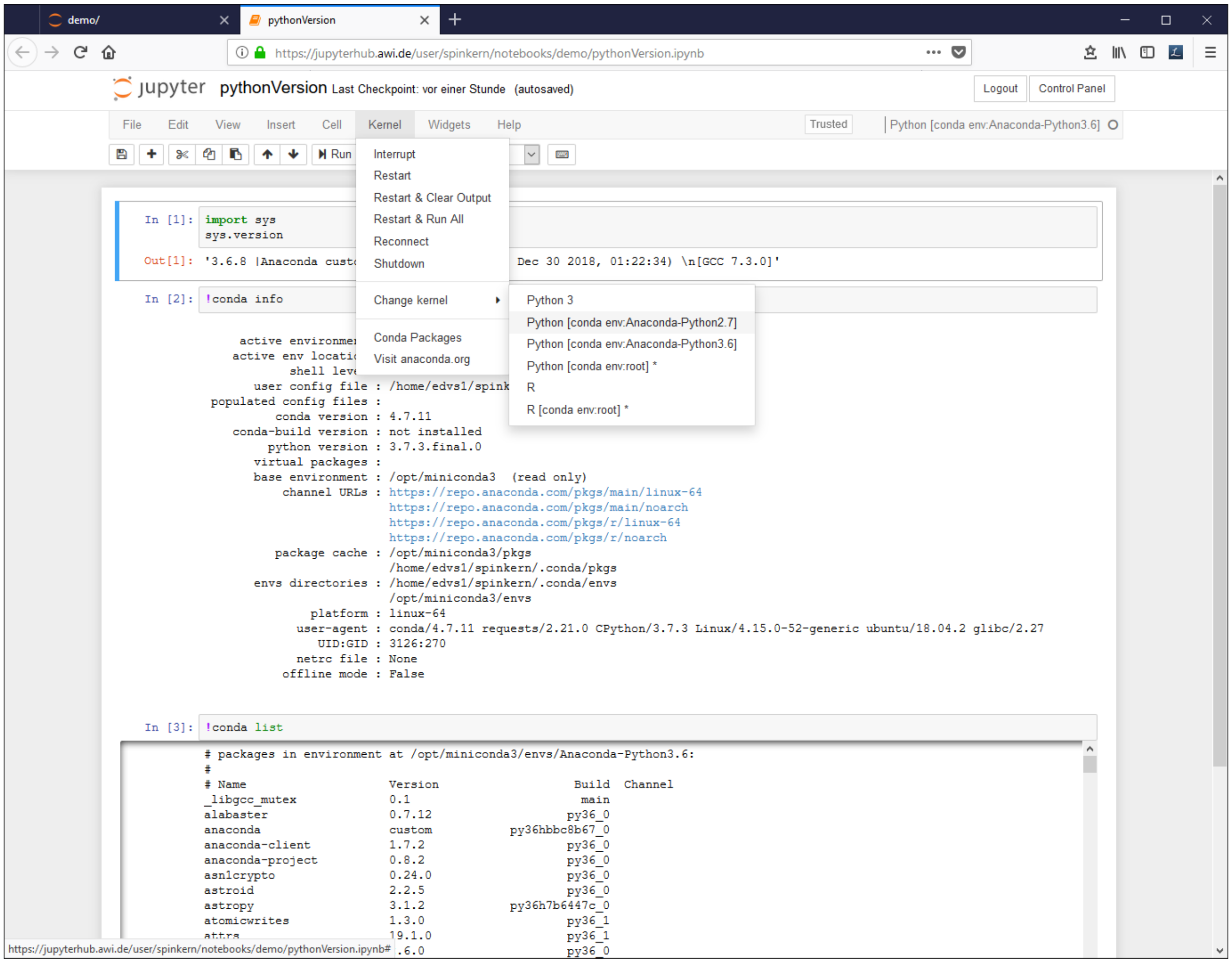

demo/
https://jupyterhub.awi.de/user/spinkern/tree/demo
jupyter
Logout
Control Panel
Files
Running
IPython Clusters
Conda
Select items to perform actions on them.
Upload
New
0
/ demo
..
pythonVersion.ipynb
RVersion.ipynb
Notebook:
Python 3
Python [conda env:Anaconda-Python2.7]
Python [conda env:Anaconda-Python3.6]
Python [conda env:root] *
R
R [conda env:root] *
Other:
Text File
Folder
Terminal

Edit the document. A source code (Python in this case) cell can be executed by pressing CTRL and Enter buttons:


demo/
pythonVersion
https://jupyterhub.awi.de/user/spinkern/notebooks/demo/pythonVersion.ipynb
jupyter
pythonVersion Last Checkpoint: vor einer Stunde (autosaved)
Logout
Control Panel
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python [conda env:Anaconda-Python3.6]
Run
Code

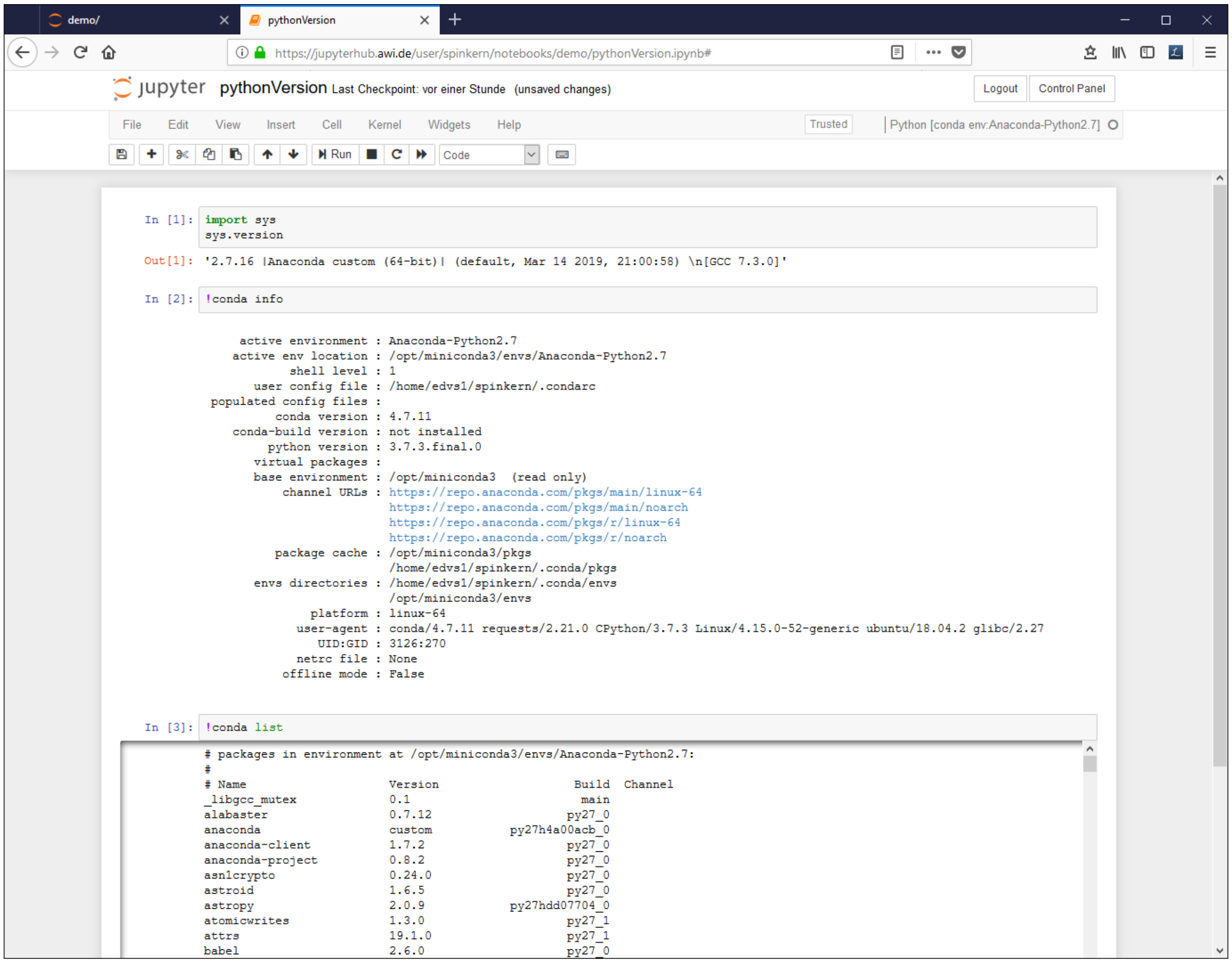
```
In [1]: import sys
        sys.version

Out[1]: '3.6.8 |Anaconda custom (64-bit)| (default, Dec 30 2018, 01:22:34) \n[GCC 7.3.0]'

In [2]: !conda info

     active environment : Anaconda-Python3.6
    active env location : /opt/miniconda3/envs/Anaconda-Python3.6
            shell level : 1
       user config file : /home/edvs1/spinkern/.condarc
 populated config files :
          conda version : 4.7.11
    conda-build version : not installed
         python version : 3.7.3.final.0
       virtual packages :
       base environment : /opt/miniconda3  (read only)
           channel URLs : https://repo.anaconda.com/pkgs/main/linux-64
                          https://repo.anaconda.com/pkgs/main/noarch
                          https://repo.anaconda.com/pkgs/r/linux-64
                          https://repo.anaconda.com/pkgs/r/noarch
          package cache : /opt/miniconda3/pkgs
                          /home/edvs1/spinkern/.conda/pkgs
       envs directories : /home/edvs1/spinkern/.conda/envs
                          /opt/miniconda3/envs
               platform : linux-64
             user-agent : conda/4.7.11 requests/2.21.0 CPython/3.7.3 Linux/4.15.0-52-generic ubuntu/18.04.2 glibc/2.27
                UID:GID : 3126:270
             netrc file : None
           offline mode : False

In [3]: !conda list

# packages in environment at /opt/miniconda3/envs/Anaconda-Python3.6:
#
# Name                    Version                   Build  Channel
_libgcc_mutex             0.1                        main
alabaster                 0.7.12                   py36_0
anaconda                  custom           py36hbbc8b67_0
anaconda-client           1.7.2                    py36_0
anaconda-project          0.8.2                    py36_0
asn1crypto                0.24.0                   py36_0
astroid                   2.2.5                    py36_0
astropy                   3.1.2            py36h7b6447c_0
atomicwrites              1.3.0                    py36_1
attrs                     19.1.0                   py36_1
babel                     2.6.0                    py36_0
```

To change the environment, click “Kernel”, “Change kernel”, and select another environment from the list.


demo/
pythonVersion
https://jupyterhub.awi.de/user/spinkern/notebooks/demo/pythonVersion.ipynb
jupyter
pythonVersion Last Checkpoint: vor einer Stunde (autosaved)
Logout
Control Panel
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python [conda env:Anaconda-Python3.6]
Run
Interrupt
Restart
Restart & Clear Output
Restart & Run All
Reconnect
Shutdown
Change kernel
Conda Packages
Visit anaconda.org
Python 3
Python [conda env:Anaconda-Python2.7]
Python [conda env:Anaconda-Python3.6]
Python [conda env:root] *
R
R [conda env:root] *
https://jupyterhub.awi.de/user/spinkern/notebooks/demo/pythonVersion.ipynb#

```
In [1]: import sys
        sys.version

Out[1]: '3.6.8 |Anaconda custo                    Dec 30 2018, 01:22:34) \n[GCC 7.3.0]'

In [2]: !conda info

             active environme
           active env locati
                 shell leve
           user config file : /home/edvs1/spink
     populated config files :
              conda version : 4.7.11
        conda-build version : not installed
             python version : 3.7.3.final.0
           virtual packages :
           base environment : /opt/miniconda3  (read only)
               channel URLs : https://repo.anaconda.com/pkgs/main/linux-64
                              https://repo.anaconda.com/pkgs/main/noarch
                              https://repo.anaconda.com/pkgs/r/linux-64
                              https://repo.anaconda.com/pkgs/r/noarch
              package cache : /opt/miniconda3/pkgs
                              /home/edvs1/spinkern/.conda/pkgs
           envs directories : /home/edvs1/spinkern/.conda/envs
                              /opt/miniconda3/envs
                   platform : linux-64
                 user-agent : conda/4.7.11 requests/2.21.0 CPython/3.7.3 Linux/4.15.0-52-generic ubuntu/18.04.2 glibc/2.27
                    UID:GID : 3126:270
                 netrc file : None
               offline mode : False


In [3]: !conda list

        # packages in environment at /opt/miniconda3/envs/Anaconda-Python3.6:
        #
        # Name                    Version                   Build  Channel
        _libgcc_mutex             0.1                        main
        alabaster                 0.7.12                   py36_0
        anaconda                  custom           py36hbbc8b67_0
        anaconda-client           1.7.2                    py36_0
        anaconda-project          0.8.2                    py36_0
        asn1crypto                0.24.0                   py36_0
        astroid                   2.2.5                    py36_0
        astropy                   3.1.2            py36h7b6447c_0
        atomicwrites              1.3.0                    py36_1
        attrs                     19.1.0                   py36_1
                                  .6.0                     py36_0
```

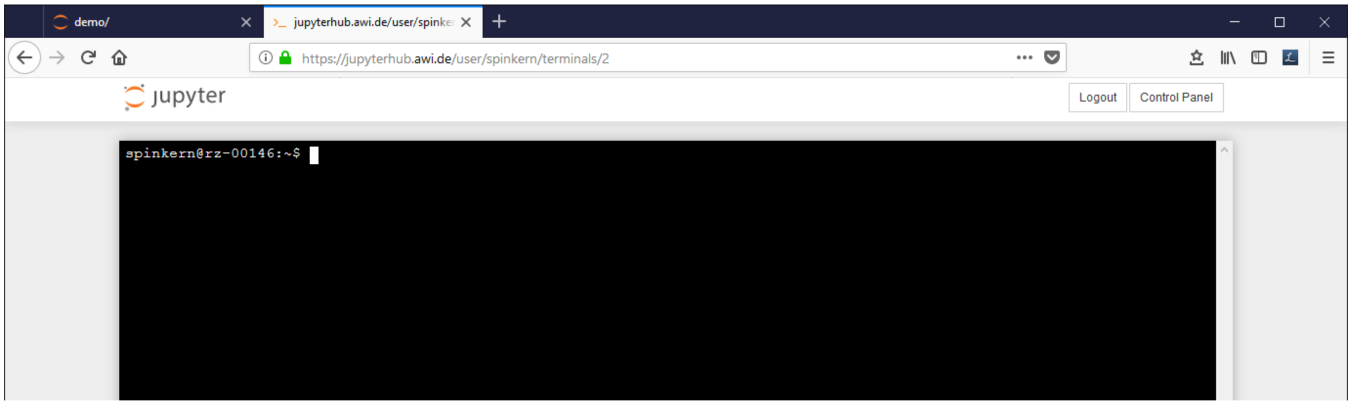
Executing the notebook again changes the output:


demo/
pythonVersion
https://jupyterhub.awi.de/user/spinkern/notebooks/demo/pythonVersion.ipynb#
jupyter pythonVersion Last Checkpoint: vor einer Stunde (unsaved changes)
Logout
Control Panel
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python [conda env:Anaconda-Python2.7]
Run
Code

```
In [1]: import sys
        sys.version

Out[1]: '2.7.16 |Anaconda custom (64-bit)| (default, Mar 14 2019, 21:00:58) \n[GCC 7.3.0]'

In [2]: !conda info

             active environment : Anaconda-Python2.7
            active env location : /opt/miniconda3/envs/Anaconda-Python2.7
                    shell level : 1
               user config file : /home/edvs1/spinkern/.condarc
         populated config files :
                  conda version : 4.7.11
            conda-build version : not installed
                 python version : 3.7.3.final.0
               virtual packages :
               base environment : /opt/miniconda3  (read only)
                   channel URLs : https://repo.anaconda.com/pkgs/main/linux-64
                                  https://repo.anaconda.com/pkgs/main/noarch
                                  https://repo.anaconda.com/pkgs/r/linux-64
                                  https://repo.anaconda.com/pkgs/r/noarch
                  package cache : /opt/miniconda3/pkgs
                                  /home/edvs1/spinkern/.conda/pkgs
               envs directories : /home/edvs1/spinkern/.conda/envs
                                  /opt/miniconda3/envs
                       platform : linux-64
                     user-agent : conda/4.7.11 requests/2.21.0 CPython/3.7.3 Linux/4.15.0-52-generic ubuntu/18.04.2 glibc/2.27
                        UID:GID : 3126:270
                     netrc file : None
                   offline mode : False

In [3]: !conda list

        # packages in environment at /opt/miniconda3/envs/Anaconda-Python2.7:
        #
        # Name                    Version                   Build  Channel
        _libgcc_mutex             0.1                        main
        alabaster                 0.7.12                   py27_0
        anaconda                  custom           py27h4a00acb_0
        anaconda-client           1.7.2                    py27_0
        anaconda-project          0.8.2                    py27_0
        asn1crypto                0.24.0                   py27_0
        astroid                   1.6.5                    py27_0
        astropy                   2.0.9            py27hdd07704_0
        atomicwrites              1.3.0                    py27_1
        attrs                     19.1.0                   py27_1
        babel                     2.6.0                    py27_0
```

Terminal

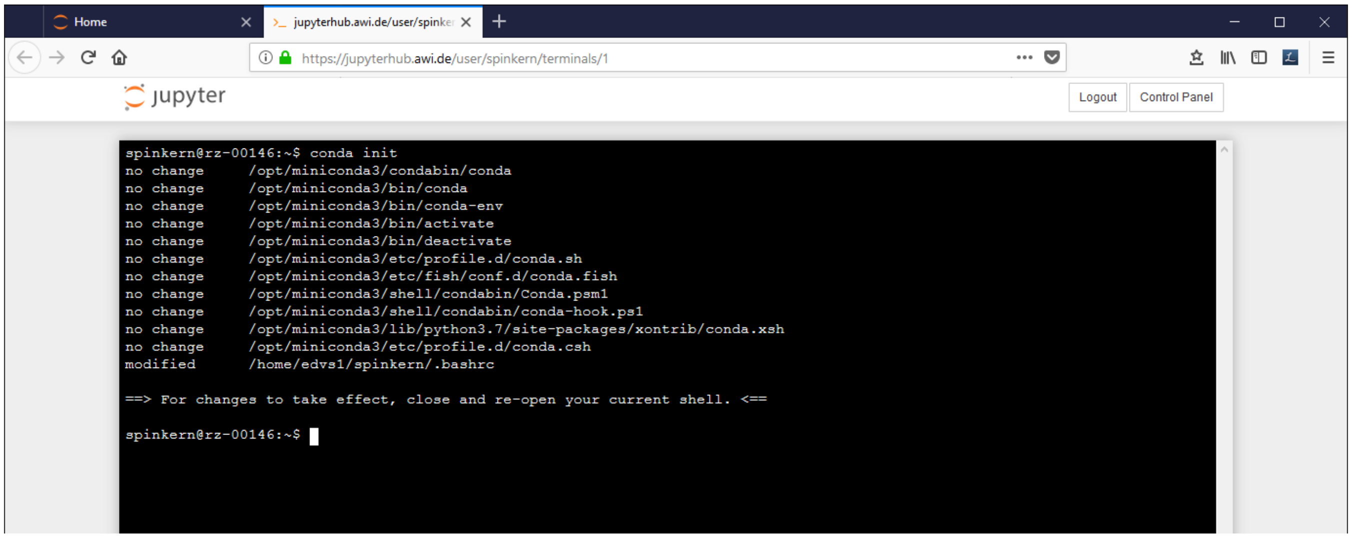
In the “Files” tab, click “New” and “Terminal”.


demo/
jupyterhub.awi.de/user/spinker
https://jupyterhub.awi.de/user/spinkern/terminals/2
jupyter
Logout
Control Panel

```
spinkern@rz-00146:~$
```

A config tool automatically sets the paths for the conda tools. This has to be activated once by:

> */opt/miniconda3/bin/conda init bash*


Home
jupyterhub.awi.de/user/spinker
https://jupyterhub.awi.de/user/spinkern/terminals/1
jupyter
Logout
Control Panel

```
spinkern@rz-00146:~$ conda init
no change     /opt/miniconda3/condabin/conda
no change     /opt/miniconda3/bin/conda
no change     /opt/miniconda3/bin/conda-env
no change     /opt/miniconda3/bin/activate
no change     /opt/miniconda3/bin/deactivate
no change     /opt/miniconda3/etc/profile.d/conda.sh
no change     /opt/miniconda3/etc/fish/conf.d/conda.fish
no change     /opt/miniconda3/shell/condabin/Conda.psm1
no change     /opt/miniconda3/shell/condabin/conda-hook.ps1
no change     /opt/miniconda3/lib/python3.7/site-packages/xontrib/conda.xsh
no change     /opt/miniconda3/etc/profile.d/conda.csh
modified      /home/edvs1/spinkern/.bashrc

==> For changes to take effect, close and re-open your current shell. <==

spinkern@rz-00146:~$
```

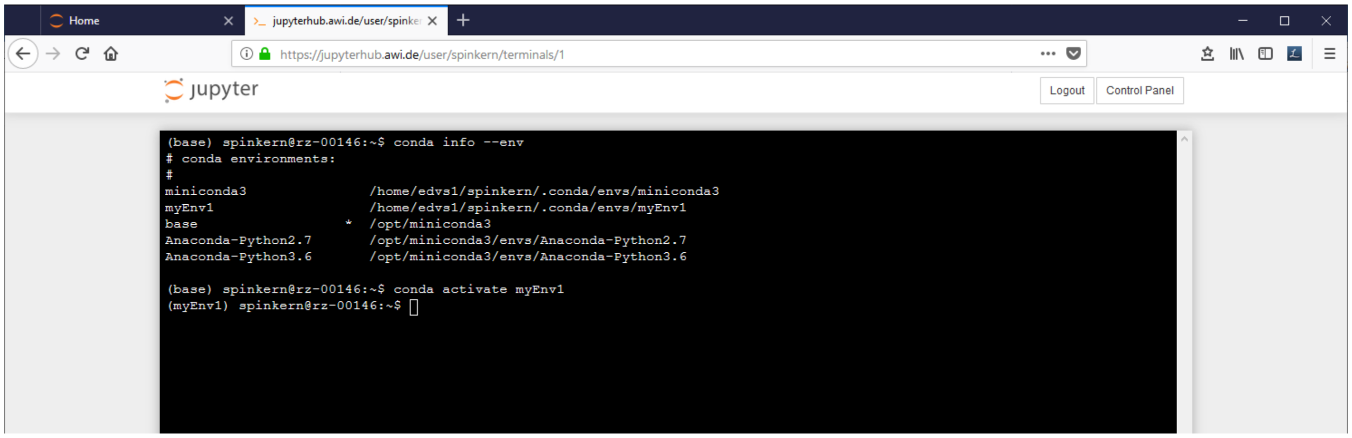
To get a listing of the environments:

> *conda info –env*

To change the active environment:

> *conda activate myEnv1*

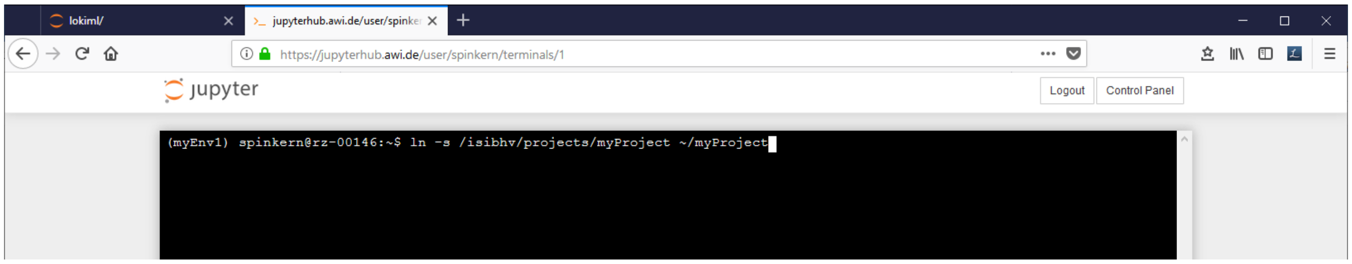
The active environment is indicated in brackets in the beginning of the line in the terminal.


Home
jupyterhub.awi.de/user/spinker
https://jupyterhub.awi.de/user/spinkern/terminals/1
jupyter
Logout
Control Panel

```
(base) spinkern@rz-00146:~$ conda info --env
# conda environments:
#
miniconda3               /home/edvs1/spinkern/.conda/envs/miniconda3
myEnv1                   /home/edvs1/spinkern/.conda/envs/myEnv1
base                  *  /opt/miniconda3
Anaconda-Python2.7       /opt/miniconda3/envs/Anaconda-Python2.7
Anaconda-Python3.6       /opt/miniconda3/envs/Anaconda-Python3.6

(base) spinkern@rz-00146:~$ conda activate myEnv1
(myEnv1) spinkern@rz-00146:~$
```

To place a link to the MCS into your home folder type the following commands into the terminal:

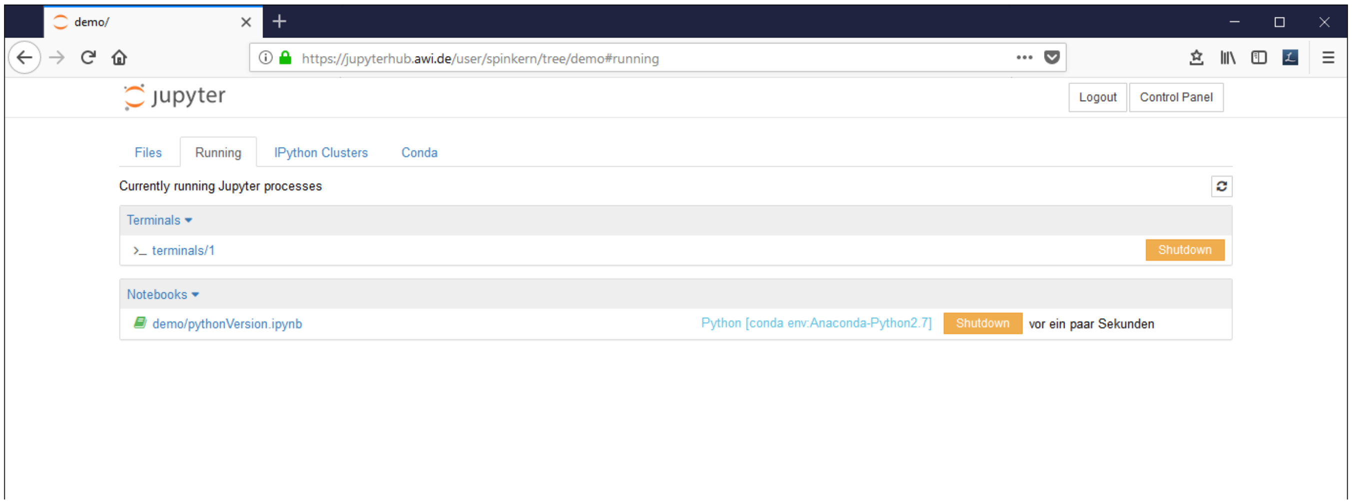
> *cd*
>
> *ln -s /isibhv/(…)*



```
(myEnv1) spinkern@rz-00146:~$ ln -s /isibhv/projects/myProject ~/myProject
```

The “Running” tab

This tab gives an overview of running notebook documents, terminals, etc. It is recommended to close all running notebooks, that are not actively used.



Currently running Jupyter processes

Terminals

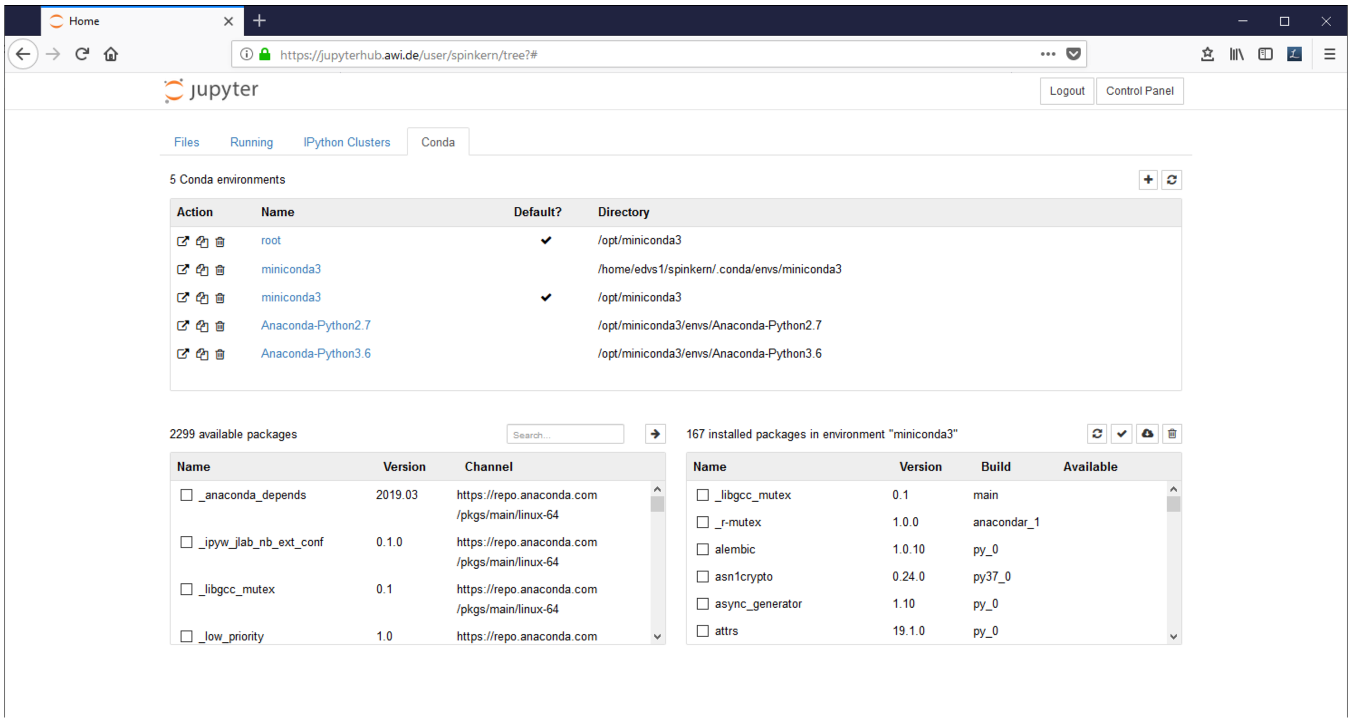
terminals/1 Shutdown

Notebooks

demo/pythonVersion.ipynb Python [conda env:Anaconda-Python2.7] Shutdown vor ein paar Sekunden

The “Conda” tab

The Conda tab can be used to create new Python 2 and Python 3 environments and to install additional libraries.



5 Conda environments

Action	Name	Default?	Directory
	root	✓	/opt/miniconda3
	miniconda3		/home/edvs1/spinkern/.conda/envs/miniconda3
	miniconda3	✓	/opt/miniconda3
	Anaconda-Python2.7		/opt/miniconda3/envs/Anaconda-Python2.7
	Anaconda-Python3.6		/opt/miniconda3/envs/Anaconda-Python3.6

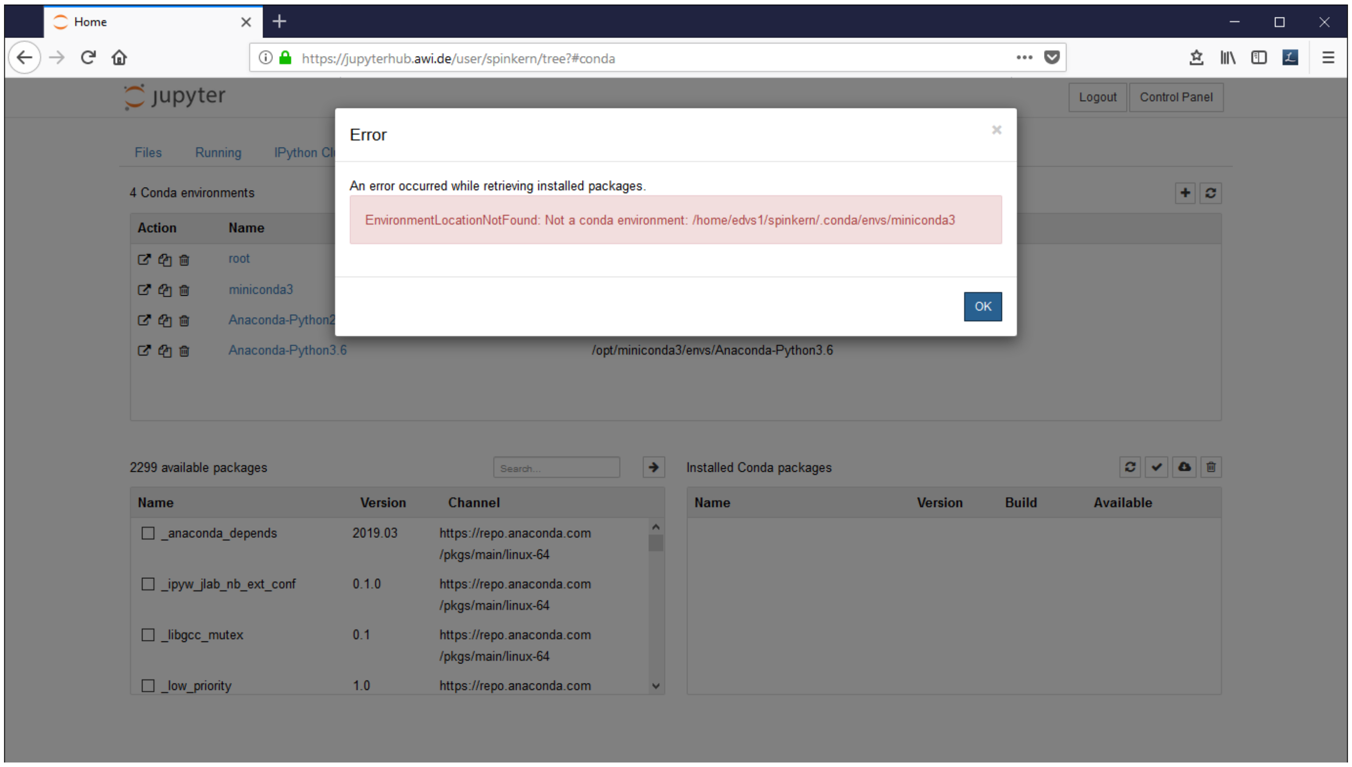
2299 available packages

Name	Version	Channel
_anaconda_depends	2019.03	https://repo.anaconda.com/pkgs/main/linux-64
_ipyw_jlab_nb_ext_conf	0.1.0	https://repo.anaconda.com/pkgs/main/linux-64
_libgcc_mutex	0.1	https://repo.anaconda.com/pkgs/main/linux-64
_low_priority	1.0	https://repo.anaconda.com

167 installed packages in environment "miniconda3"

Name	Version	Build	Available
_libgcc_mutex	0.1	main	
_r-mutex	1.0.0	anacondar_1	
alembic	1.0.10	py_0	
asn1crypto	0.24.0	py37_0	
async_generator	1.10	py_0	
attrs	19.1.0	py_0	

Suppress the error message in the Conda tab



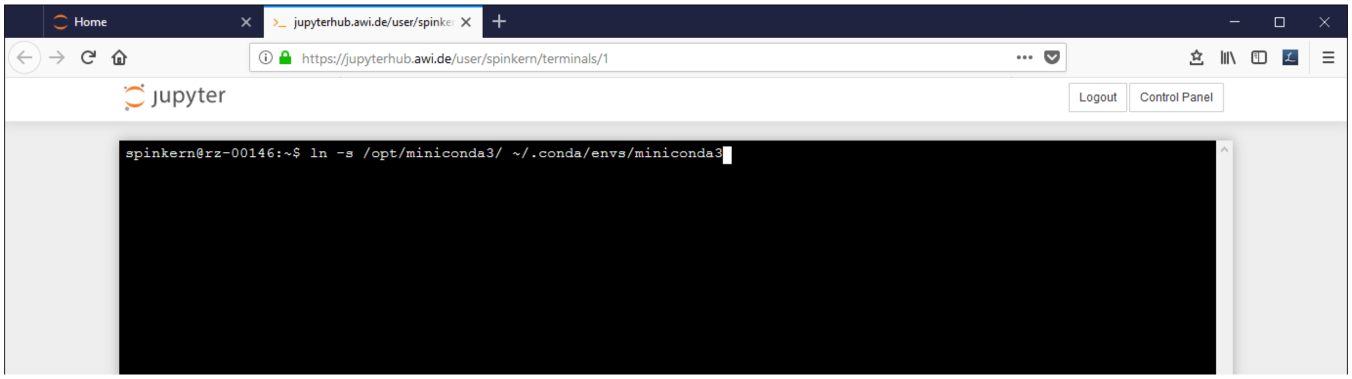
Error

An error occurred while retrieving installed packages.

EnvironmentLocationNotFound: Not a conda environment: /home/edvs1/spinkern/.conda/envs/miniconda3

In case you get this error message, execute the following command in the terminal:

> *ln -s /opt/miniconda3/ ~/.conda/envs/miniconda3*



```
spinkern@rz-00146:~$ ln -s /opt/miniconda3/ ~/.conda/envs/miniconda3
```

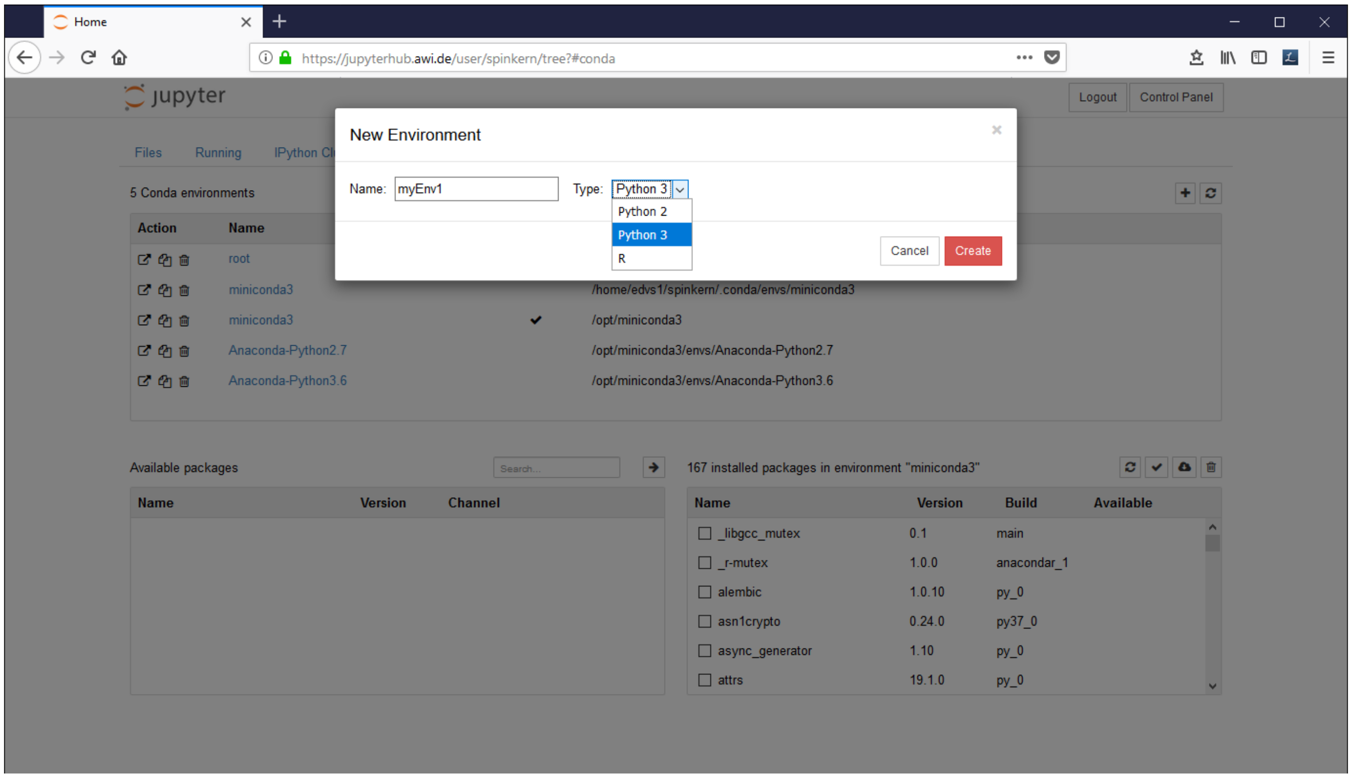
The “Clusters” tab

This option is not supported in this version.

Python

Creating an own Python Environment

In the “conda” tab, click on the plus button in the upper right corner. Set a name for the new environment, select “Python 2” or “Python 3” from the dropdown box, and click “Create”.



New Environment

Name: myEnv1 Type: Python 3

5 Conda environments

Action	Name		
	root		
	miniconda3		/home/edvs1/spinkern/.conda/envs/miniconda3
	miniconda3	✓	/opt/miniconda3
	Anaconda-Python2.7		/opt/miniconda3/envs/Anaconda-Python2.7
	Anaconda-Python3.6		/opt/miniconda3/envs/Anaconda-Python3.6

Available packages

Name	Version	Channel

167 installed packages in environment "miniconda3"

Name	Version	Build	Available
_libgcc_mutex	0.1	main	
_r-mutex	1.0.0	anacondar_1	
alembic	1.0.10	py_0	
asn1crypto	0.24.0	py37_0	
async_generator	1.10	py_0	
attrs	19.1.0	py_0	

Alternatively, a new environment can be created by the conda CLI using the terminal.

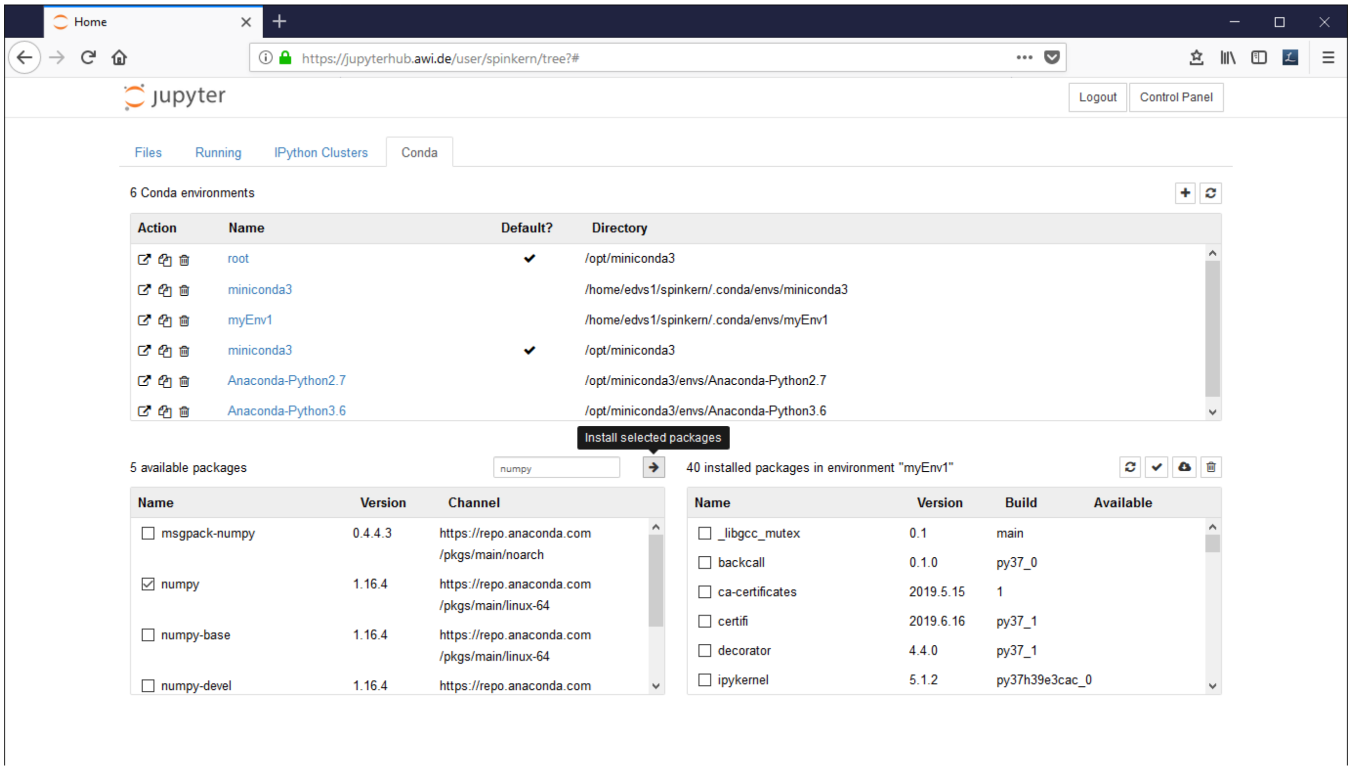
Installing Python libraries

The users cannot modify the pre-installed conda environments. So, to install additional libraries, an own environment needs to be created first.

In the "Conda" tab, select the new environment ('myEnv1' in this example).

The installed libraries are listed in the window at the bottom, on the right side. Available packages can be selected in the list at the bottom, left side.

Mark the desired packages and click the Arrow Button to install them. The underlying conda package manager automatically resolves package dependencies.


Home
https://jupyterhub.awi.de/user/spinkern/tree?#
jupyter
Logout
Control Panel
Files
Running
IPython Clusters
Conda
Install selected packages

6 Conda environments

Action	Name	Default?	Directory
	root	✓	/opt/miniconda3
	miniconda3		/home/edvs1/spinkern/.conda/envs/miniconda3
	myEnv1		/home/edvs1/spinkern/.conda/envs/myEnv1
	miniconda3	✓	/opt/miniconda3
	Anaconda-Python2.7		/opt/miniconda3/envs/Anaconda-Python2.7
	Anaconda-Python3.6		/opt/miniconda3/envs/Anaconda-Python3.6

5 available packages numpy

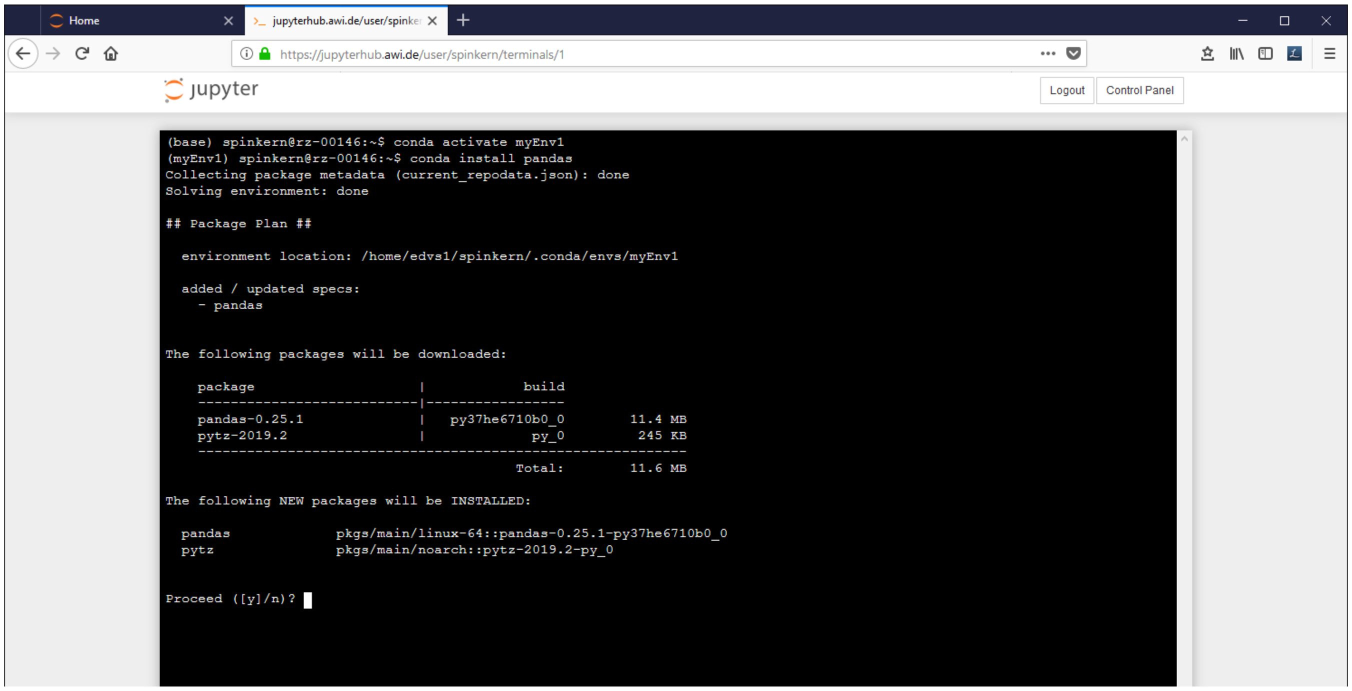
Name	Version	Channel
☐ msgpack-numpy	0.4.4.3	https://repo.anaconda.com /pkgs/main/noarch
☑ numpy	1.16.4	https://repo.anaconda.com /pkgs/main/linux-64
☐ numpy-base	1.16.4	https://repo.anaconda.com /pkgs/main/linux-64
☐ numpy-devel	1.16.4	https://repo.anaconda.com

40 installed packages in environment "myEnv1"

Name	Version	Build	Available
☐ _libgcc_mutex	0.1	main	
☐ backcall	0.1.0	py37_0	
☐ ca-certificates	2019.5.15	1	
☐ certifi	2019.6.16	py37_1	
☐ decorator	4.4.0	py37_1	
☐ ipykernel	5.1.2	py37h39e3cac_0	

Conda package manager

Instead of using the "Conda" tab mentioned before, you can also use the conda package manager via the terminal. The package manager allows to configure the conda environment, which is the base of the Jupyter Notebook application.


Home
jupyterhub.awi.de/user/spinke
https://jupyterhub.awi.de/user/spinkern/terminals/1
jupyter
Logout
Control Panel

```
(base) spinkern@rz-00146:~$ conda activate myEnv1
(myEnv1) spinkern@rz-00146:~$ conda install pandas
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/edvs1/spinkern/.conda/envs/myEnv1

  added / updated specs:
    - pandas


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    pandas-0.25.1              |   py37he6710b0_0        11.4 MB
    pytz-2019.2                |             py_0         245 KB
    ------------------------------------------------------------
                                           Total:        11.6 MB

The following NEW packages will be INSTALLED:

  pandas             pkgs/main/linux-64::pandas-0.25.1-py37he6710b0_0
  pytz               pkgs/main/noarch::pytz-2019.2-py_0


Proceed ([y]/n)?
```

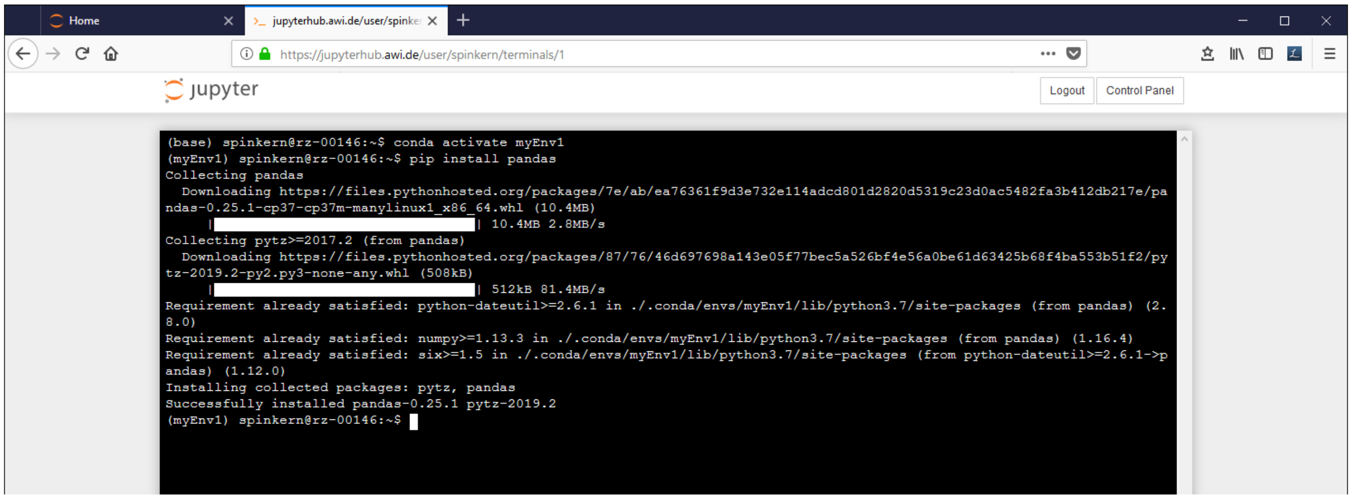
PIP

Most packages are available via the conda package manager, as described below. Alternatively, pip can be used via the terminal.

Please make sure to use the pip version, that belongs to the correct Python environment.

The environments’ version can be found under: ~/.conda/envs/[envName]/bin/pip

Alternatively, the environment has to be “activated” first, which links the global “pip” command to that of the activated environment.


Home
jupyterhub.awi.de/user/spinke
https://jupyterhub.awi.de/user/spinkern/terminals/1
jupyter
Logout
Control Panel

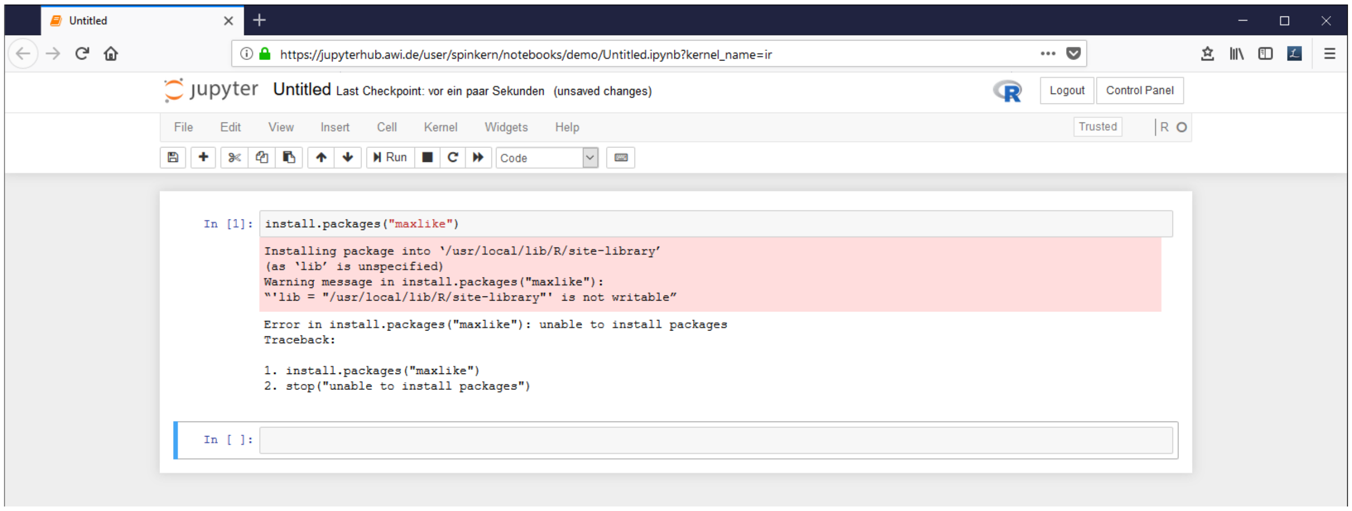
```
(base) spinkern@rz-00146:~$ conda activate myEnv1
(myEnv1) spinkern@rz-00146:~$ pip install pandas
Collecting pandas
  Downloading https://files.pythonhosted.org/packages/7e/ab/ea76361f9d3e732e114adcd801d2820d5319c23d0ac5482fa3b412db217e/pa
ndas-0.25.1-cp37-cp37m-manylinux1_x86_64.whl (10.4MB)
     |████████████████████████████████| 10.4MB 2.8MB/s
Collecting pytz>=2017.2 (from pandas)
  Downloading https://files.pythonhosted.org/packages/87/76/46d697698a143e05f77bec5a526bf4e56a0be61d63425b68f4ba553b51f2/py
tz-2019.2-py2.py3-none-any.whl (508kB)
     |████████████████████████████████| 512kB 81.4MB/s
Requirement already satisfied: python-dateutil>=2.6.1 in ./.conda/envs/myEnv1/lib/python3.7/site-packages (from pandas) (2.
8.0)
Requirement already satisfied: numpy>=1.13.3 in ./.conda/envs/myEnv1/lib/python3.7/site-packages (from pandas) (1.16.4)
Requirement already satisfied: six>=1.5 in ./.conda/envs/myEnv1/lib/python3.7/site-packages (from python-dateutil>=2.6.1->p
andas) (1.12.0)
Installing collected packages: pytz, pandas
Successfully installed pandas-0.25.1 pytz-2019.2
(myEnv1) spinkern@rz-00146:~$ 
```

R

The R environment comes with preinstalled Tidyverse meta-package. Users can add R packages, which will be installed in a personal library in their central home folder. In contrast to the Python environments, no additional environment needs to be created. Additional R libraries can be installed using the install.package() command within R, either using an R notebook or the terminal.

Please note that the install command might fail, if it is submitted from within a notebook document.

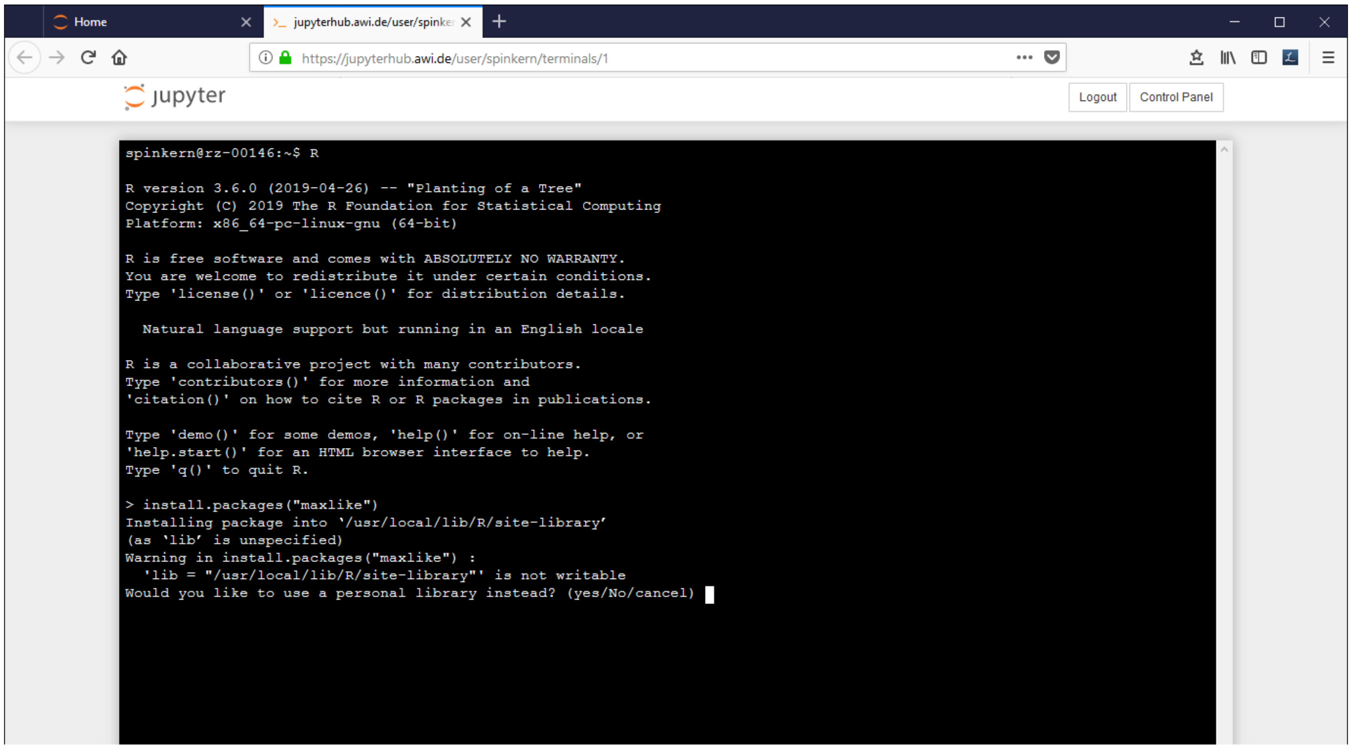
Example: The local library needs to be created before a new package can be installed. Within the notebook, the install command fails, while in the terminal the program asks what to do...


Untitled
https://jupyterhub.awi.de/user/spinkern/notebooks/demo/Untitled.ipynb?kernel_name=ir
jupyter
Untitled Last Checkpoint: vor ein paar Sekunden (unsaved changes)
Logout
Control Panel
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
R
Run
Code

```
In [1]: install.packages("maxlike")

        Installing package into ‘/usr/local/lib/R/site-library’
        (as ‘lib’ is unspecified)
        Warning message in install.packages("maxlike"):
        "'lib = "/usr/local/lib/R/site-library"' is not writable"

        Error in install.packages("maxlike"): unable to install packages
        Traceback:

        1. install.packages("maxlike")
        2. stop("unable to install packages")

In [ ]:
```

Start the terminal as described above and start the R console by typing "R".

Here, the same install command leads to a dialog, to ask if a personal library shall be used instead.


Home
jupyterhub.awi.de/user/spinker
https://jupyterhub.awi.de/user/spinkern/terminals/1
jupyter
Logout
Control Panel

```
spinkern@rz-00146:~$ R

R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> install.packages("maxlike")
Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)
Warning in install.packages("maxlike") :
  'lib = "/usr/local/lib/R/site-library"' is not writable
Would you like to use a personal library instead? (yes/No/cancel)
```