


The Group of Applied Mathematical Modeling, Static and Optimization (MATHMODE) has available a workstation namely ρK (“*La roca*”). This computer has been purchased with MATHMODE funds and it is available for our use!

Technical Resources:



(4x) GPU: NVIDIA Quadro GV100; 32GB VRAM
 (2x) CPU: Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz
 (12x) Memory: 64GiB
 Disk : TOSHIBA MG04ACA6; 5589GiB

SSH: 10.227.85.95

Your account:

User:	matteo-mathmode
Pass:	montecroci

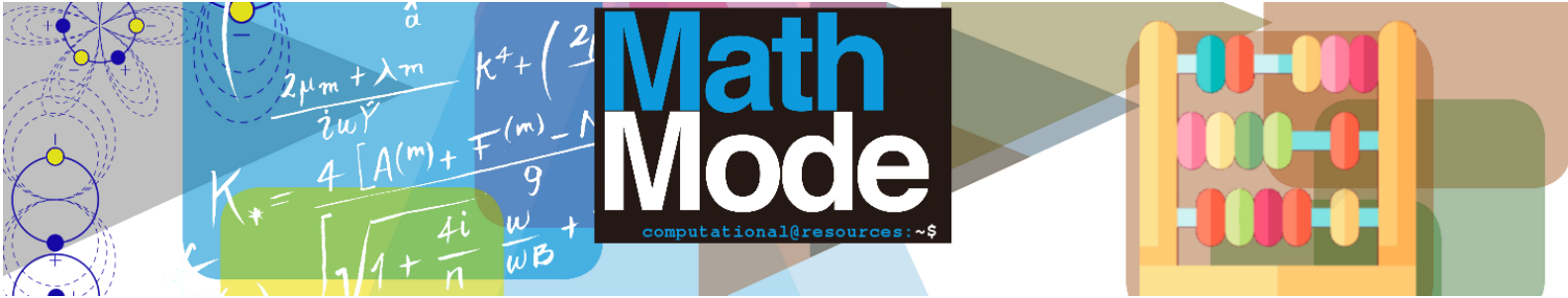
The account holder undertakes not to stores illicit files, private data and, in general, any data that is not related to your research.

Test your connection to the workstation ρK :

(Press Ctrl + c to close.)

```
local@pc:~$ ping 10.227.85.95
PING 10.227.85.95 (10.227.85.95) 56(84) bytes of data.
64 bytes from 10.227.85.95: icmp_seq=1 ttl=61 time=26.2 ms
64 bytes from 10.227.85.95: icmp_seq=2 ttl=61 time=26.7 ms
64 bytes from 10.227.85.95: icmp_seq=3 ttl=61 time=26.9 ms
64 bytes from 10.227.85.95: icmp_seq=4 ttl=61 time=24.9 ms
64 bytes from 10.227.85.95: icmp_seq=5 ttl=61 time=25.7 ms
--- 10.227.85.95 ping statistics ---
5 packets transmitted, 5 received, 100% packet loss, time 4006ms
```

The connection to the workstation ρK must be done through the UPV/EHU intranet. If you are using an external connection, first you must connect to this intranet in secure mode. For that, go to <https://www.ehu.eus/es/web/ikt-tic/vpn> and install the software AnyConnect. You will be able to connect to vpn.ehu.es by using your ldap user and your password.



Install Miniconda and tensorflow in your ρK user:

- 1- Create a new folder in your Desktop namely `install`. There:
 - a) Create the two files detailed in [appendix A](#)
`mathmode.tf.yml` `test_devices.py`
 - b) the last version of Python 3.10 Miniconda Linux 64-bit installer, it is available in the link <https://docs.conda.io/en/latest/miniconda.html#linux-installers>.
- 2- Open a terminal and copy the `install` folder with the 3 files to ρK :

```
local@pc:~$ scp -r Desktop/install matteo_mathmode@10.227.85.95:
matteo_mathmode@10.227.85.95's password: montecroci
```

- 3- Connect to the ρK workstation:

```
local@pc:~$ ssh matteo_mathmode@10.227.85.95
matteo_mathmode@10.227.85.95's password: montecroci
matteo_mathmode@azken-WS-C621E-SAGE-Series:~$
```

- 4- Execute the file copied to install Miniconda:

```
matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ bash install/Miniconda3-latest-Linux-x86_64.sh
Welcome to Miniconda3 4.7.12

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
...
Do you approve the license terms? [yes|no]
```

Type **yes** to accept the license.

```
Miniconda3 will now be installed into this location:
/home/matteo_mathmode/miniconda3

- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below

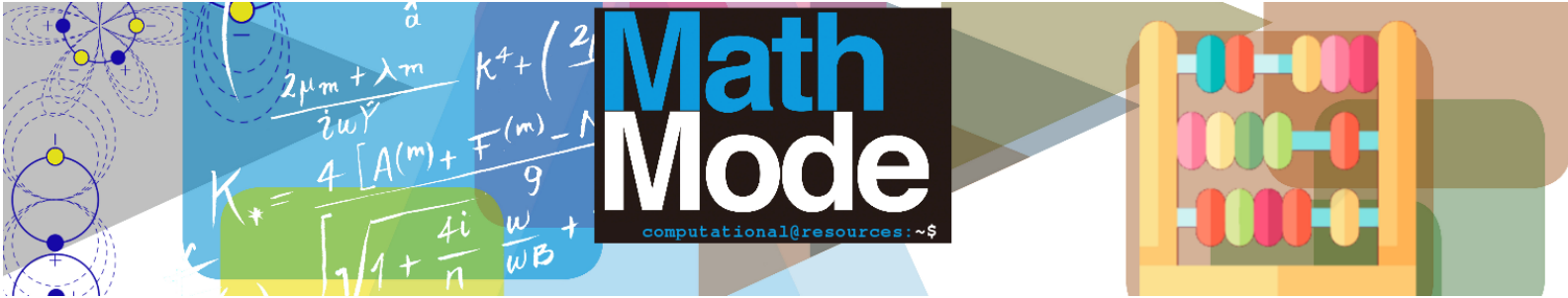
[/home/matteo_mathmode/miniconda3] >>>
```

Press **ENTER** to confirm the default location, and the installation starts.

```
installation finished.
Do you wish the installer to initialize Miniconda3
by running conda init? [yes|no]
[no] >>>
```

It is recommended to type **yes** to automatically modify your `.bashrc`.

You must close the connection to apply the changes made in your ρK `.bashrc` file. Press **Ctrl + d** or type **logout** to kill your connection with ρK .



5- Connect to the ρK workstation to install the environment:

```
local@pc:~$ ssh matteo_mathmode@10.227.85.95
matteo_mathmode@10.227.85.95's password: montecroci
```

Notice that the (base) environment is launched by default. You should be able to use conda to create/manage environments.

```
(base)matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda update --all
...
Proceed ([y]/n)? y
...
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$
```

Create the environment *tf* to install tensorflow:

```
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda env create -f install/mathmode_tf.yml
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda activate tf
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda update --all
...
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda deactivate
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$
```

In this tutorial, we only have installed the packages that requires tensorflow, you can create more environments with the packages or versions that you need for others projects.

The commands to list your environments and packages are:

```
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda env list
...
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda list
...
```

For further information on using the conda command, see

<https://docs.conda.io/projects/conda/en/latest/user-guide/index.html>

6- Install tensorflow The best way is to follow the updated instructions given in:

https://www.tensorflow.org/install/pip#linux_1

you should read all the documentation before start!

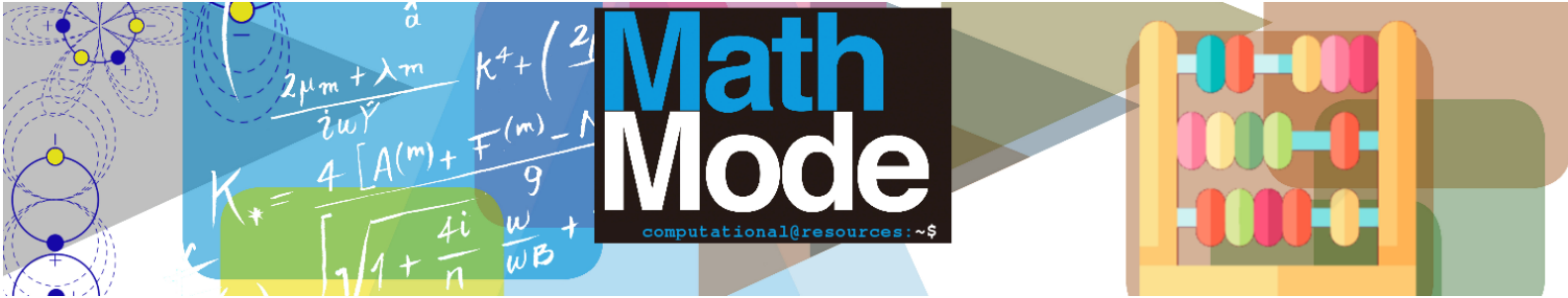
7- Remove the installer folder and create an empty work folder namely Projects:

```
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ :rm -rfv install
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ mkdir Projects
(base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ ls
miniconda3  Projects
```

(remove as well examples.desktop if you have it)

Execute your code in the ρK workstation:

1- Copy the folder of your code from your pc to the work folder Projects placed in your ρK user. In this tutorial we copy the CODE and the User folder TEST_PROBLEMS.



```
local@pc:~$ scp -r my_rute/CODE matteo_mathmode@10.227.85.95:Projects
local@pc:~$ scp -r my_rute/TEST_PROBLEMS matteo_mathmode@10.227.85.95:Projects
```

2- Execute the code in the ρK CPU or GPU:

```
(Base) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ conda activate tf
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ cd Projects/TEST_PROBLEMS
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~/Projects/TEST_PROBLEMS$ python main.py
```

3- Copy the results from ρK to your pc.

```
local@pc:~$ scp -r matteo_mathmode@10.227.85.95:Projects/TEST_PROBLEMS/results Desktop/
```

(Use another terminal, you do not need kill the connection)

4- Delete the files that you do not need any more. Keep ρK free of waste!

```
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ rm -rfv Projects/TEST_PROBLEMS
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ logout
```

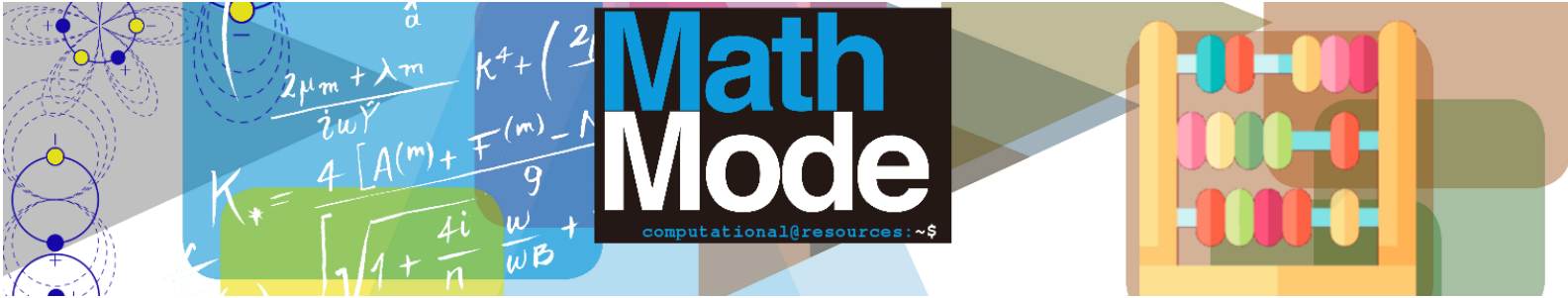
Practical notes:

In order to have a good coexistence between all users, **it is mandatory to limit the use of the GPU memory to 1GB**. For that, your code might start with some similar to:

```
gpus = tf.config.list_physical_devices('GPU')
if gpus:
    # Restrict TensorFlow to only allocate 1GB of memory on the first GPU
    try:
        tf.config.set_logical_device_configuration(
            gpus[0],
            [tf.config.LogicalDeviceConfiguration(memory_limit=1024)])
        logical_gpus = tf.config.list_logical_devices('GPU')
        print(len(gpus), "Physical GPUs,", len(logical_gpus), "Logical GPUs")
    except RuntimeError as e:
        # Virtual devices must be set before GPUs have been initialized
        print(e)
```

It is your responsibility to choose the adequate GPU to launch your code! You can get information about the usage of the GPU with this command:

```
(tf) matteo_mathmode@azken-WS-C621E-SAGE-Series:~$ watch -n 1 nvidia-smi
```



Frequently Asked Questions:

What can I do if Anaconda doesn't update the latest version of TensorFlow?

You can try to install pip in your environment: `$ conda install pip`

Then, you can force the desired package installation via:

- (tf) `$ pip install tensorflow==2.X.X`

How can I edit a file that is in the server?

a) use vi or vim (not requires graphical interface)

`$ vi filename` `$ vim filename`

Useful commands	
Esc	to exit of one mode
i	enables insert-mode
:w	save changes
:q!	quit without changes
:q	quit
:x	save and quit

Can I disconnect form the ssh connection and leave the code running?

When you stop the ssh connection, the process that is in the server dies. To avoid this, you need to use *tmux* as follows:

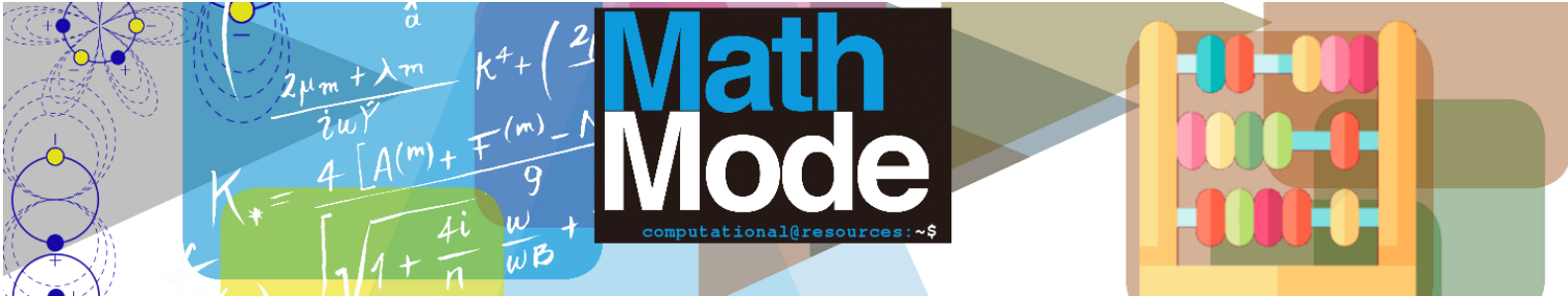
1. Connect to the the workstation ρK via ssh
2. Activate the conda environment
3. Create a new session by the command:
`$ tmux new -s name`
4. Run the code
5. Detach the session by one option:
 - a) press CTRL+b, release both keys and then press d
 - b) `$ tmux detach` (not available if you code is running)
6. Disconnect from the server

How I can check if the code ended its execution in the server?

1. Connect to the the workstation ρK via ssh
2. Attach the session
`$ tmux attach -t name`
3. **Remember to kill the session after the simulation is finished!**
`$ tmux kill-session -t name`

To get a list of the currently running sessions:

`$ tmux ls` More useful commands are in in [appendix B](#).

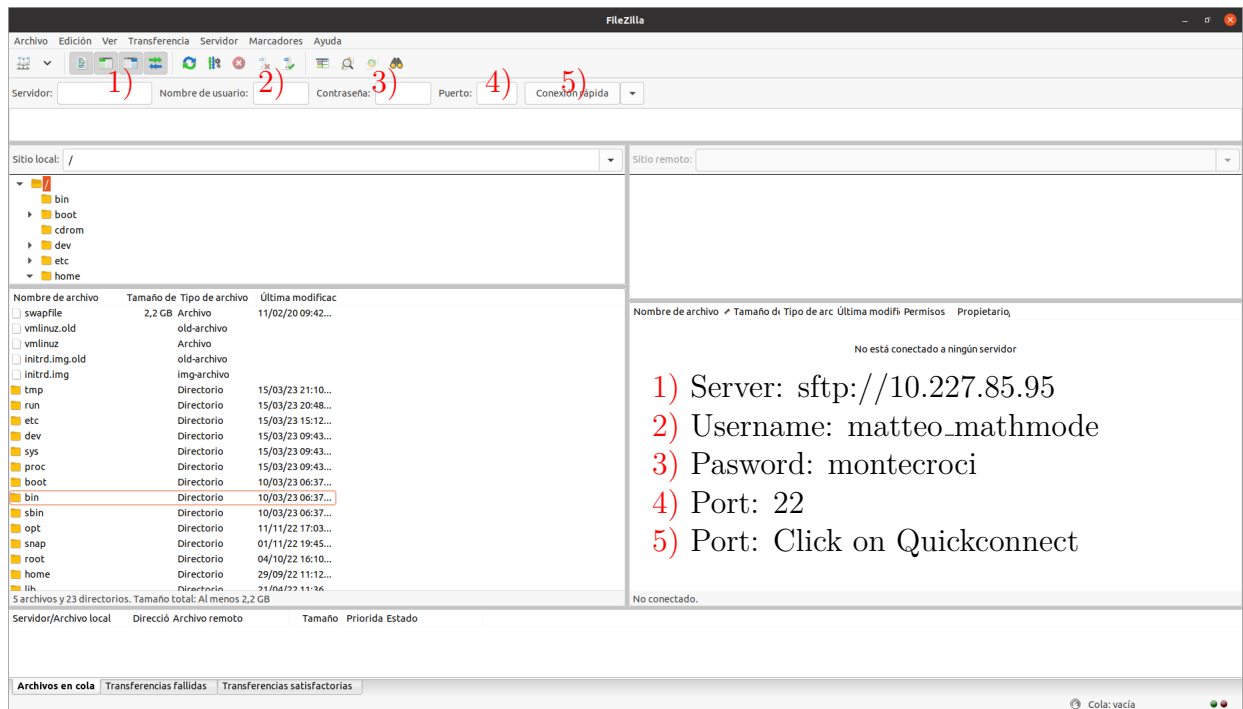


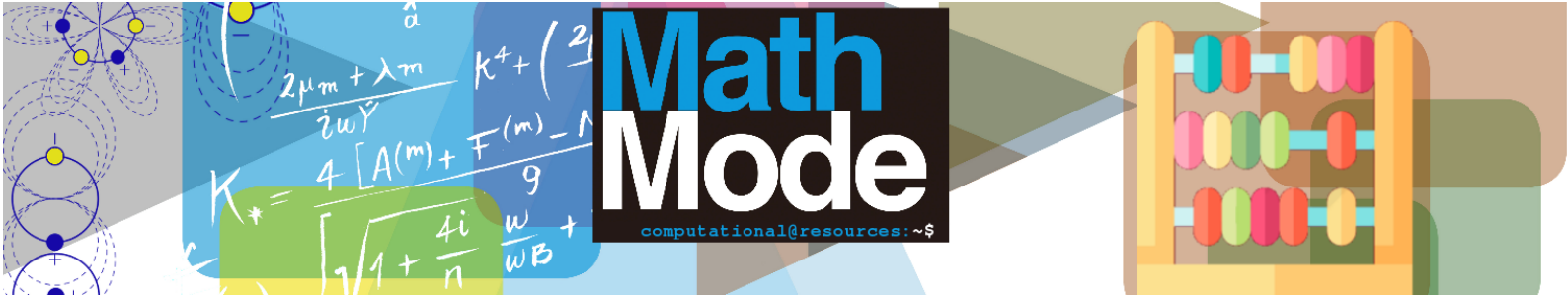
The copy procedure from the server to my local pc and vice versa is so tedious.
Can I do something?

Sure! We recomend to install in your pc fileZilla Client from:

<https://filezilla-project.org/>

To connect with the server configure as:





Appendix A

You can include/exclude the packages that you want in the `.yaml` files.

mathmode_tf.yaml

name: tf

dependencies:

- python=3.9
- numpy
- holoviews
- pandas
- matplotlib
- seaborn
- scikit-learn

test_devices.py

```
import tensorflow as tf
```

```
print(tf.__version__)
```

```
print('Num GPUs available: ', len(tf.config.experimental.list_physical_devices('GPU')))
```

```
tf.debugging.set_log_device_placement(True)
```

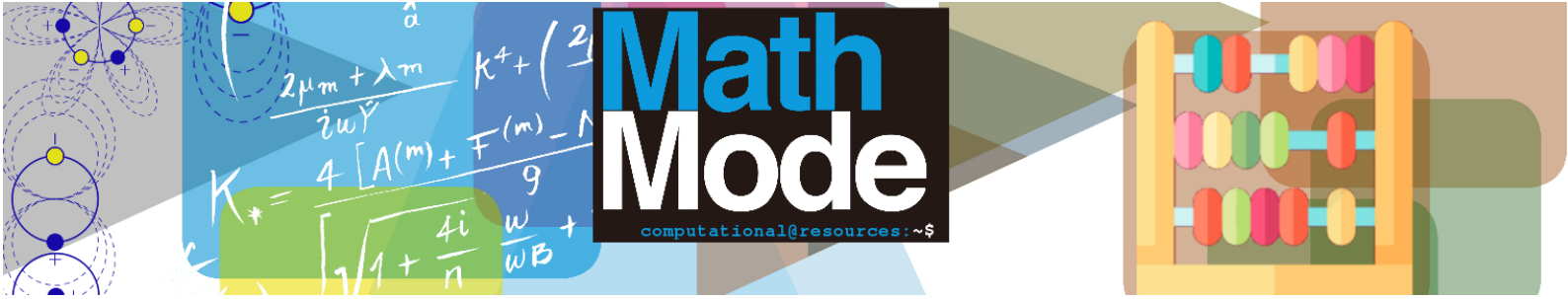
```
# Create some tensors
```

```
a = tf.constant([[1.0, 2.0, 3.0], [4.0, 5.0, 6.0]])
```

```
b = tf.constant([[1.0, 2.0], [3.0, 4.0], [5.0, 6.0]])
```

```
c = tf.matmul(a, b)
```

```
print(c)
```



Appendix B

Table 1: Starting and Exiting Tmux

Command	Description
tmux new -s <session-name>	Start a new tmux session
tmux attach	Attach to the last tmux session
tmux attach -t <session-name>	Attach to a specific tmux session
tmux ls	List all tmux sessions
tmux kill-session -t <session-name>	Kill a specific tmux session
tmux kill-server	Kill the tmux server and all sessions
Ctrl-b d	Detach from the current session

Table 2: Tmux Windows and Panes

Command	Description
Ctrl-b c	Create a new window
Ctrl-b n	Switch to the next window
Ctrl-b p	Switch to the previous window
Ctrl-b ;number;	Switch to a specific window
Ctrl-b w	List all windows
Ctrl-b %	Split the current pane vertically
Ctrl-b ”	Split the current pane horizontally
Ctrl-b o	Switch to the next pane
Ctrl-b <arrow key>	Move between panes

Table 3: Tmux Miscellaneous Commands

Command	Description
Ctrl-b ?	List all tmux commands and their key bindings
Ctrl-b :	Enter a tmux command prompt
tmux source-file <file>	Reload the tmux configuration file