# Python environments

By **Robin Fays**

Contact: robin.fays@henallux.be

# How a perfect Python environment looks like!

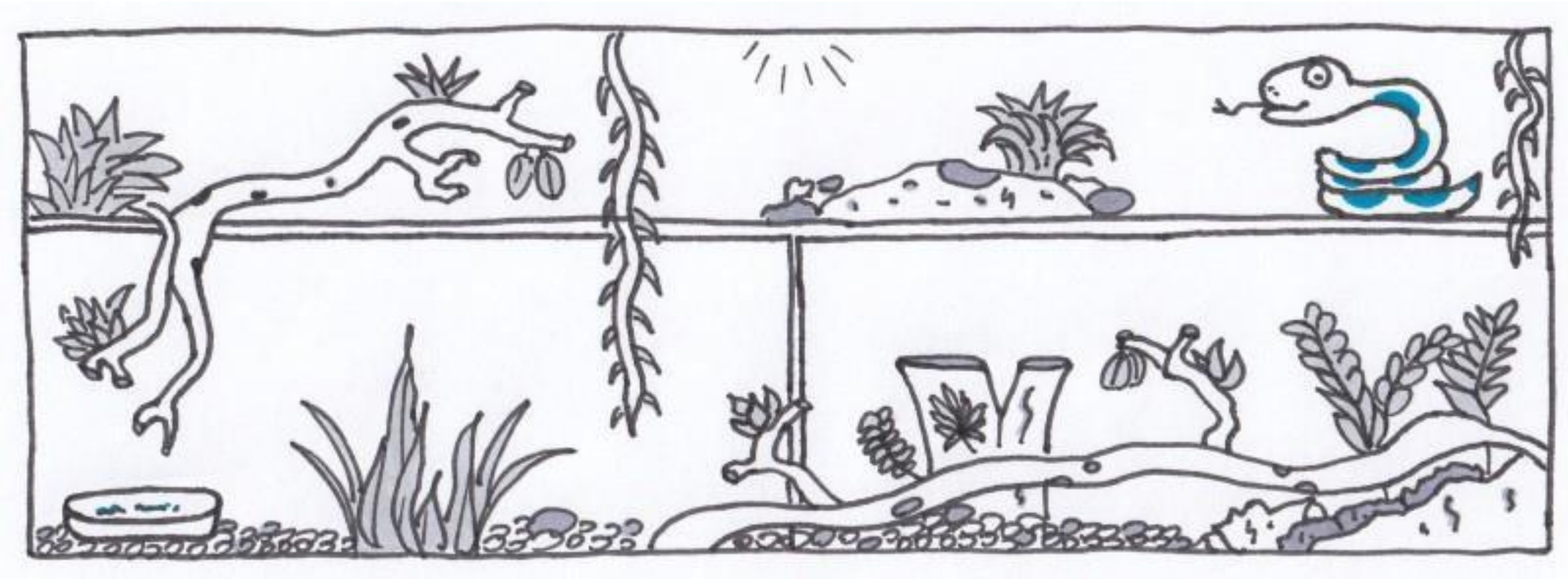
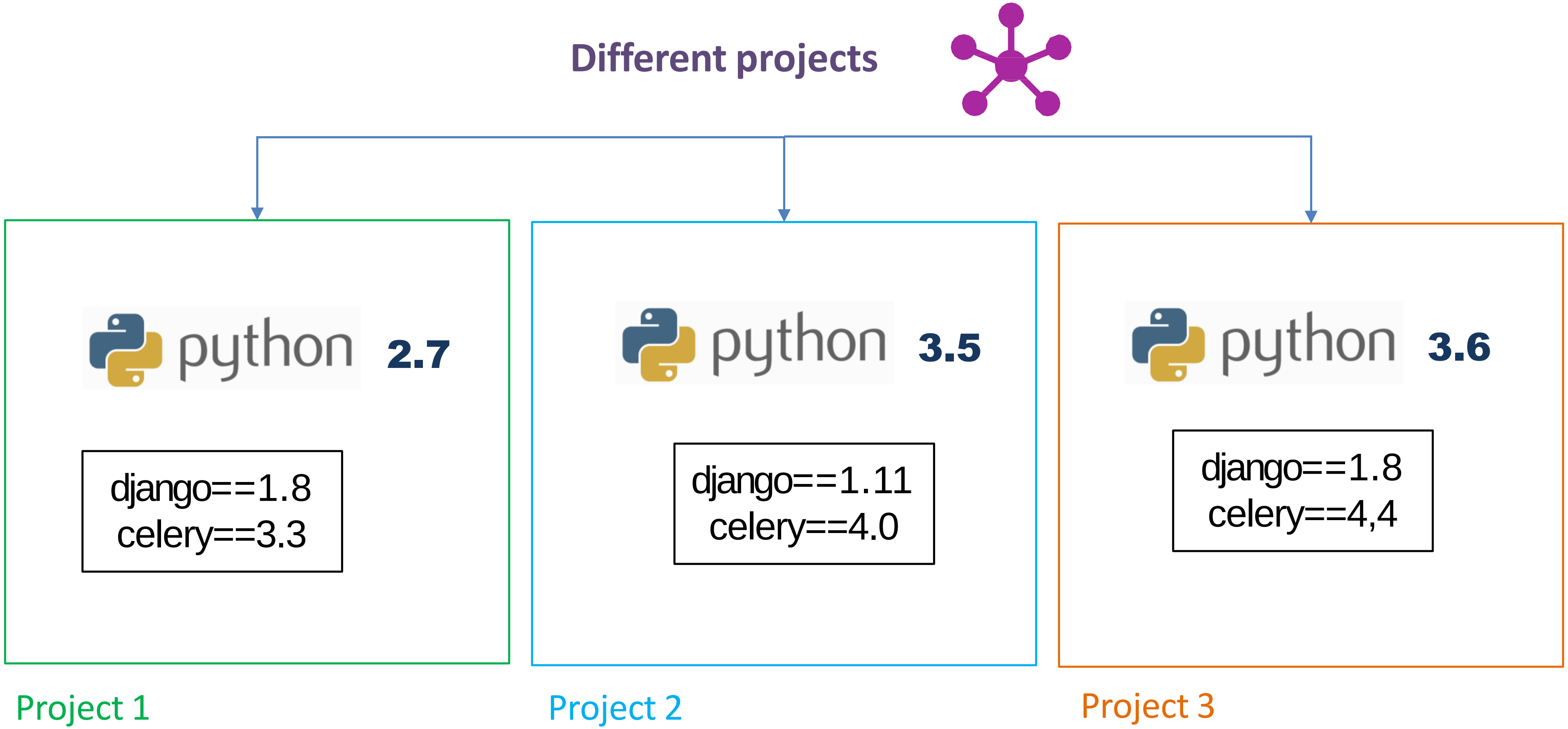


Image credit: https://medium.freecodecamp.org/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c

# Why you need multiple Python environments



**Different projects**

python **2.7**

django==1.8
celery==3.3

Project 1

python **3.5**

django==1.11
celery==4.0

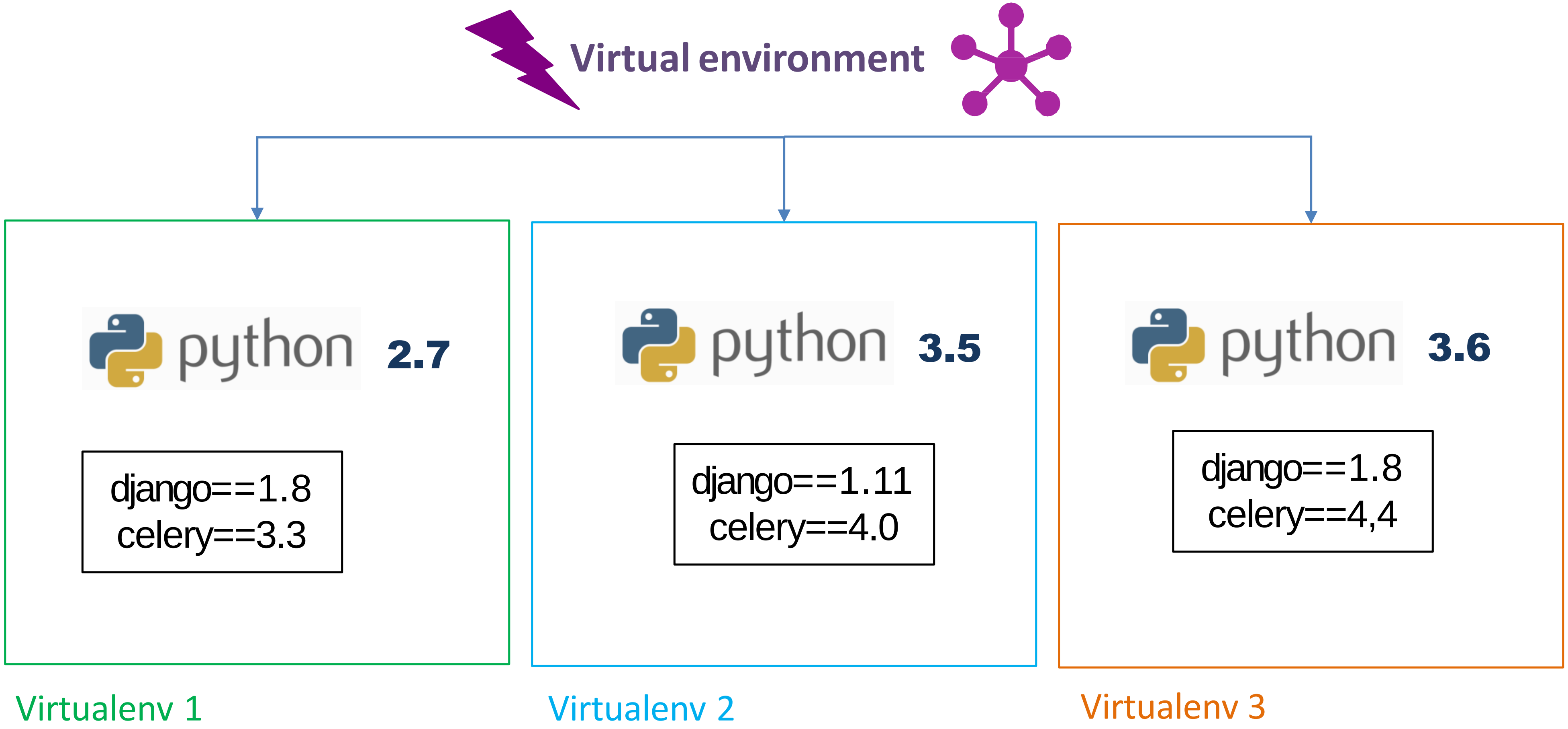
Project 2

python **3.6**

django==1.8
celery==4,4

Project 3

# Why you need multiple Python environments



Virtual environment

python 2.7

django==1.8
celery==3.3

Virtualenv 1

python 3.5

django==1.11
celery==4.0

Virtualenv 2

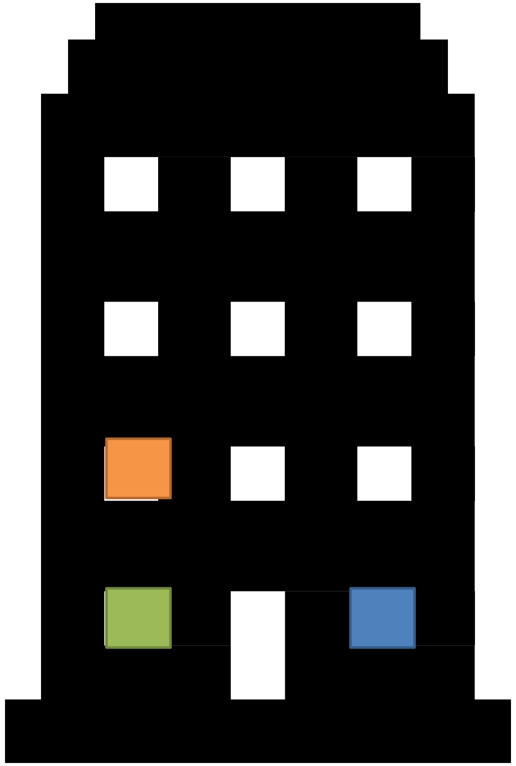
python 3.6

django==1.8
celery==4,4

Virtualenv 3

Note: If we install a package without virtualenv then it will be install as a global package.

# Virtual environment : introduction



Virtualenv 1

Python 2.7
django==1.8
celery==3.3

Virtualenv 2

Python 3,5
django==1.11
celery==4.0

Virtualenv 3

Python 3,6
django==1.8
celery==4,4

# Virtual environment : introduction

- When working with Python projects, best practice is to have a separate virtual environment for each of them
- Virtualenv is a python package
- It is used to create an isolated python environments
- virtualenv allows us to work with different versions of packages for different projects

# Virtual environments with conda

# Package and environment manager

- Conda : package and environment manager for any language

OR

- Pip : package manager for python packages only with

+

- virtualenv (a tool for creating isolated environments)

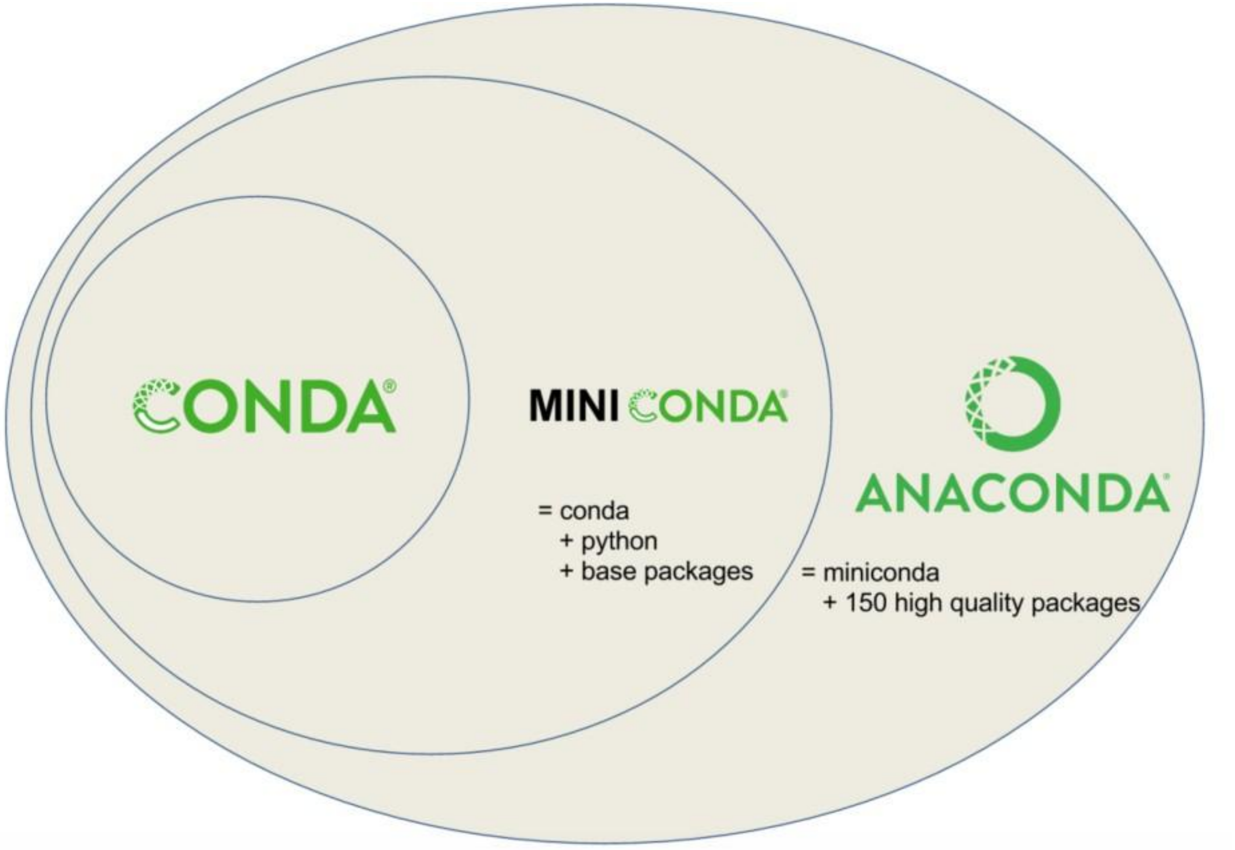
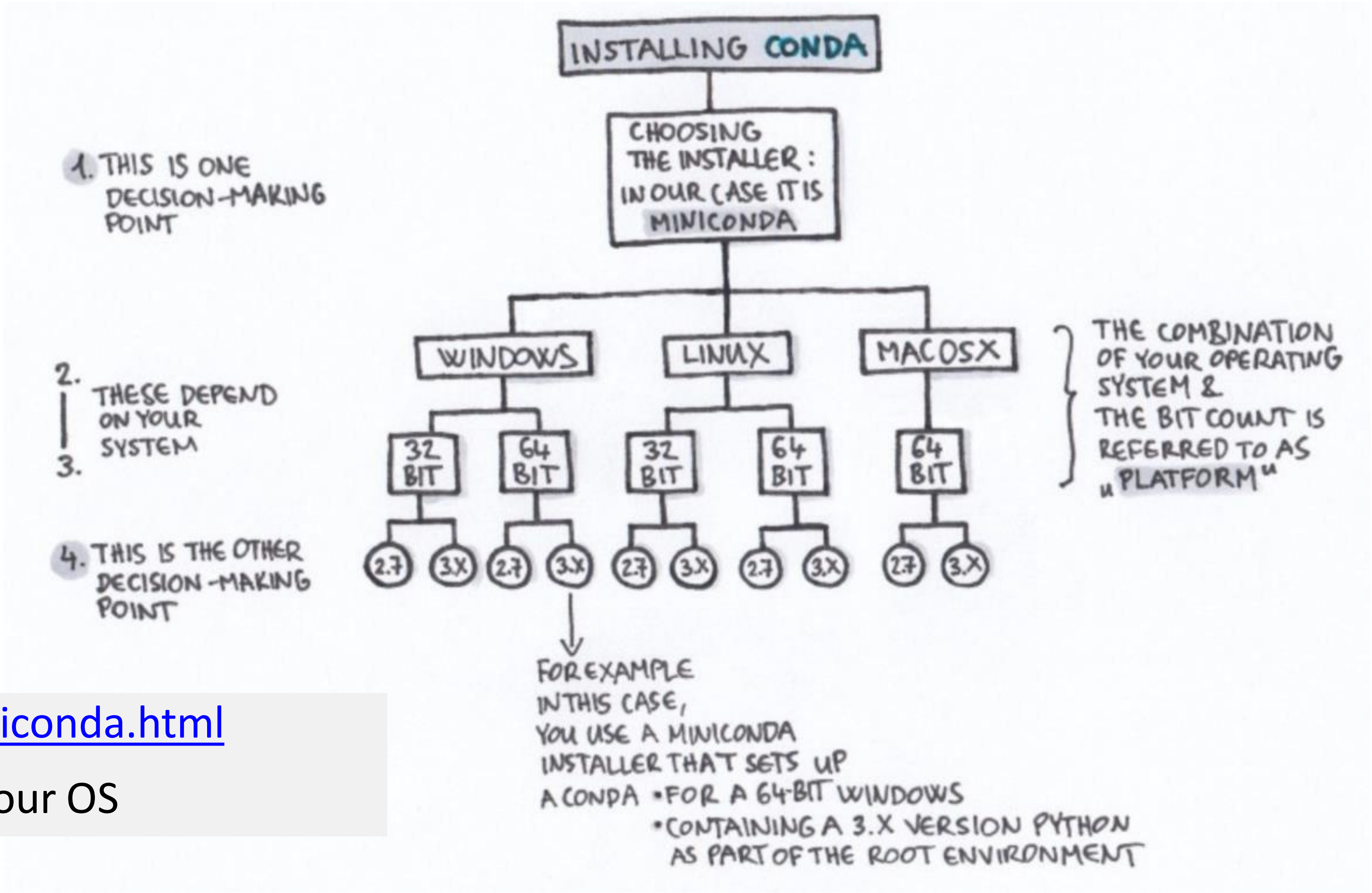


Image credit: https://medium.freecodecamp.org/

# Choosing Anaconda/Miniconda installation



1. THIS IS ONE DECISION-MAKING POINT

2.–3. THESE DEPEND ON YOUR SYSTEM

4. THIS IS THE OTHER DECISION-MAKING POINT

THE COMBINATION OF YOUR OPERATING SYSTEM & THE BIT COUNT IS REFERRED TO AS „PLATFORM“

FOR EXAMPLE IN THIS CASE, YOU USE A MINICONDA INSTALLER THAT SETS UP A CONDA
- FOR A 64-BIT WINDOWS
- CONTAINING A 3.X VERSION PYTHON AS PART OF THE ROOT ENVIRONMENT

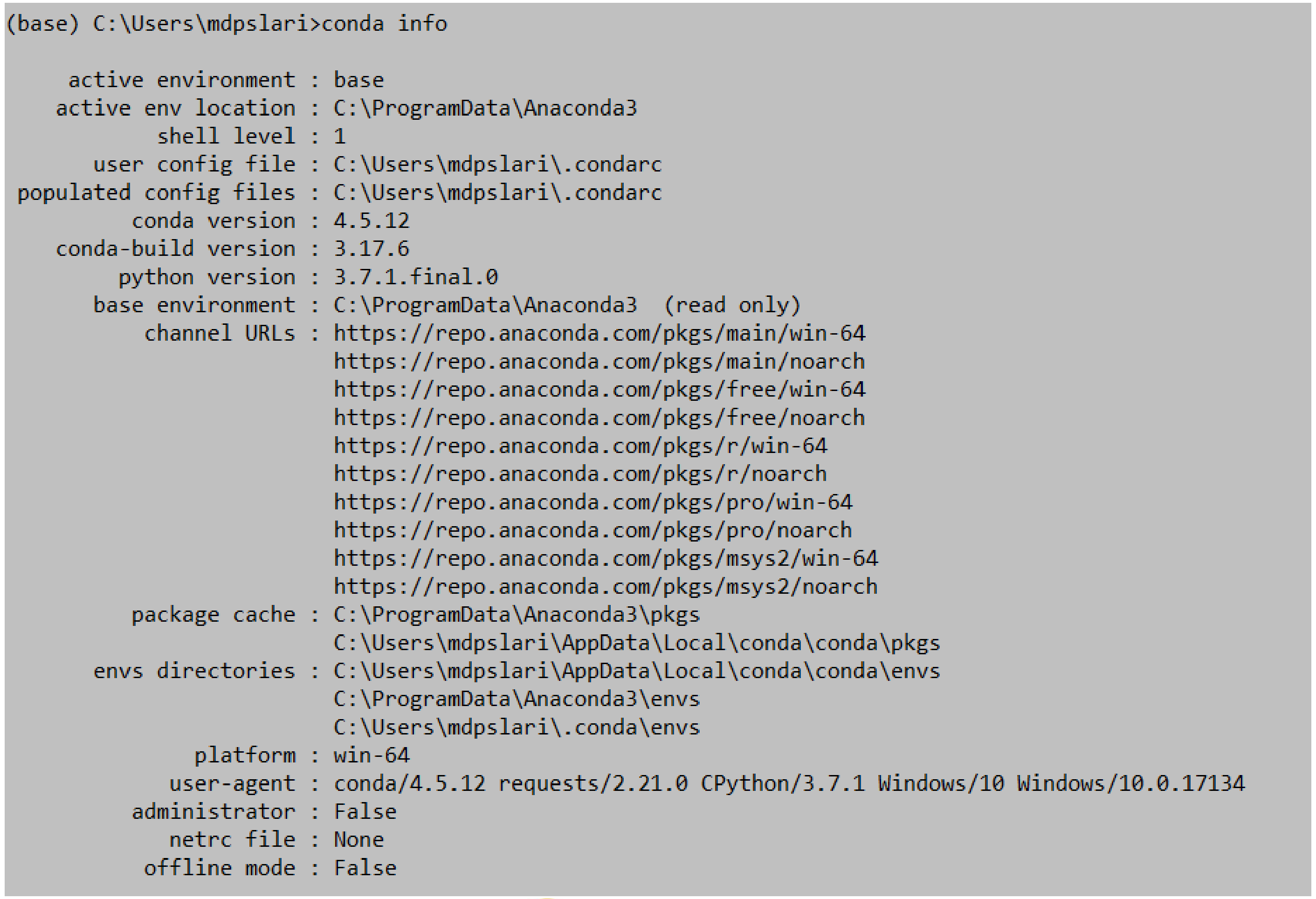
- https://docs.conda.io/en/latest/miniconda.html
- Follow installation instructions for your OS

# Root environment

Use anaconda prompt!

- Conda creates a root environment that contains a version Python and some basic packages

conda info

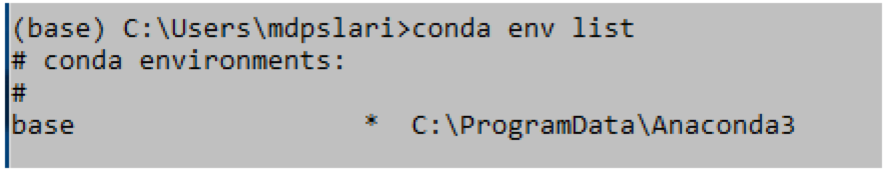


```
(base) C:\Users\mdpslari>conda info

     active environment : base
    active env location : C:\ProgramData\Anaconda3
            shell level : 1
       user config file : C:\Users\mdpslari\.condarc
 populated config files : C:\Users\mdpslari\.condarc
          conda version : 4.5.12
    conda-build version : 3.17.6
         python version : 3.7.1.final.0
       base environment : C:\ProgramData\Anaconda3  (read only)
           channel URLs : https://repo.anaconda.com/pkgs/main/win-64
                          https://repo.anaconda.com/pkgs/main/noarch
                          https://repo.anaconda.com/pkgs/free/win-64
                          https://repo.anaconda.com/pkgs/free/noarch
                          https://repo.anaconda.com/pkgs/r/win-64
                          https://repo.anaconda.com/pkgs/r/noarch
                          https://repo.anaconda.com/pkgs/pro/win-64
                          https://repo.anaconda.com/pkgs/pro/noarch
                          https://repo.anaconda.com/pkgs/msys2/win-64
                          https://repo.anaconda.com/pkgs/msys2/noarch
          package cache : C:\ProgramData\Anaconda3\pkgs
                          C:\Users\mdpslari\AppData\Local\conda\conda\pkgs
       envs directories : C:\Users\mdpslari\AppData\Local\conda\conda\envs
                          C:\ProgramData\Anaconda3\envs
                          C:\Users\mdpslari\.conda\envs
               platform : win-64
             user-agent : conda/4.5.12 requests/2.21.0 CPython/3.7.1 Windows/10 Windows/10.0.17134
          administrator : False
             netrc file : None
           offline mode : False
```

conda create --name py310 -c anaconda python

# Create a new environment

- ➢ Check the available Conda environments

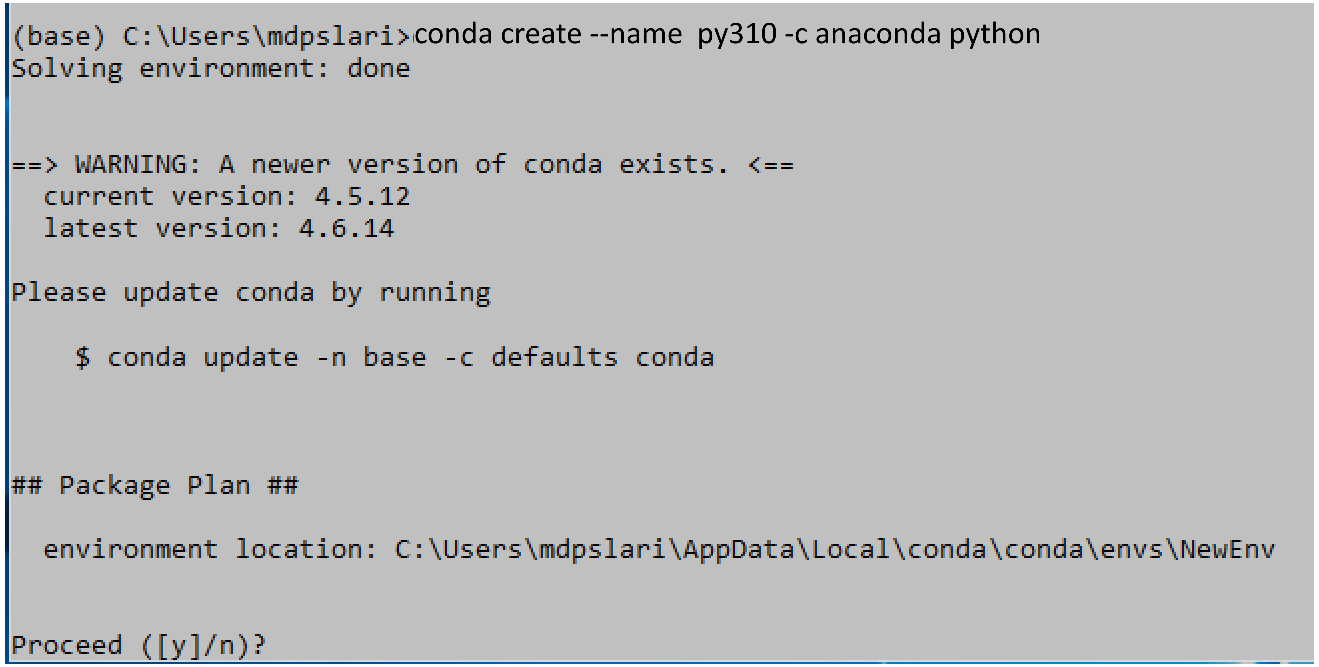


```
(base) C:\Users\mdpslari>conda env list
# conda environments:
#
base                  *  C:\ProgramData\Anaconda3
```

conda env list

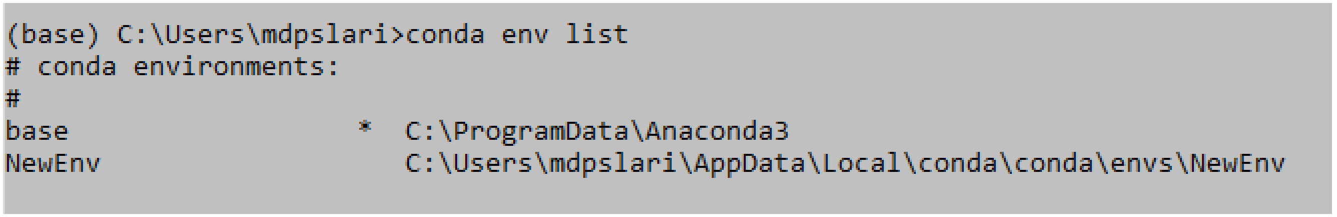
conda create --name  py310 -c anaconda python

- ➢ Create new environment



```
(base) C:\Users\mdpslari>conda create --name  py310 -c anaconda python
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.12
  latest version: 4.6.14

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: C:\Users\mdpslari\AppData\Local\conda\conda\envs\NewEnv


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

- ➢ check again the available conda environments



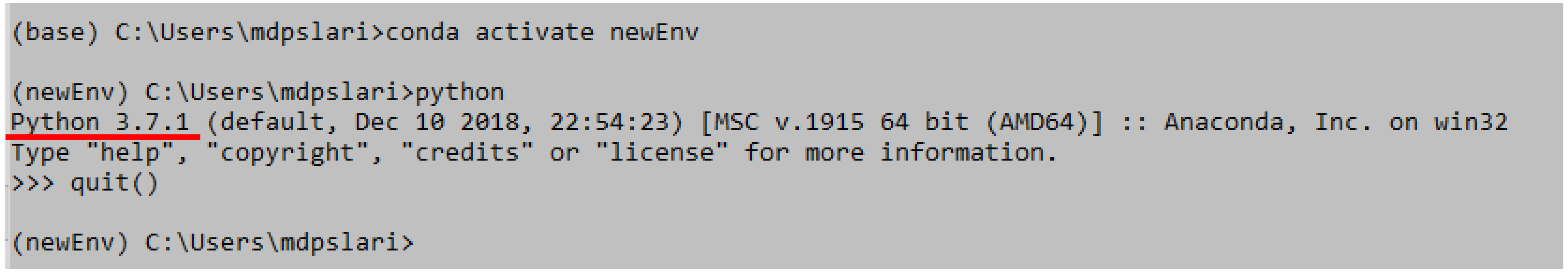
```
(base) C:\Users\mdpslari>conda env list
# conda environments:
#
base                     *  C:\ProgramData\Anaconda3
NewEnv                      C:\Users\mdpslari\AppData\Local\conda\conda\envs\NewEnv
```

# Activate/Deactivate an environment

- ➢ Activate the environment and check python version

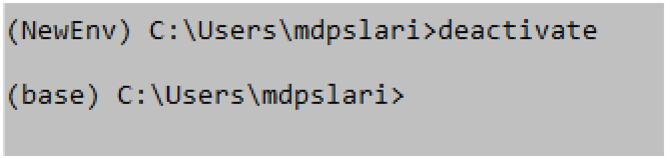
conda activate newEnv

conda deactivate



```
(base) C:\Users\mdpslari>conda activate newEnv

(newEnv) C:\Users\mdpslari>python
Python 3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()

(newEnv) C:\Users\mdpslari>
```

- ➢ Deactivate the environment

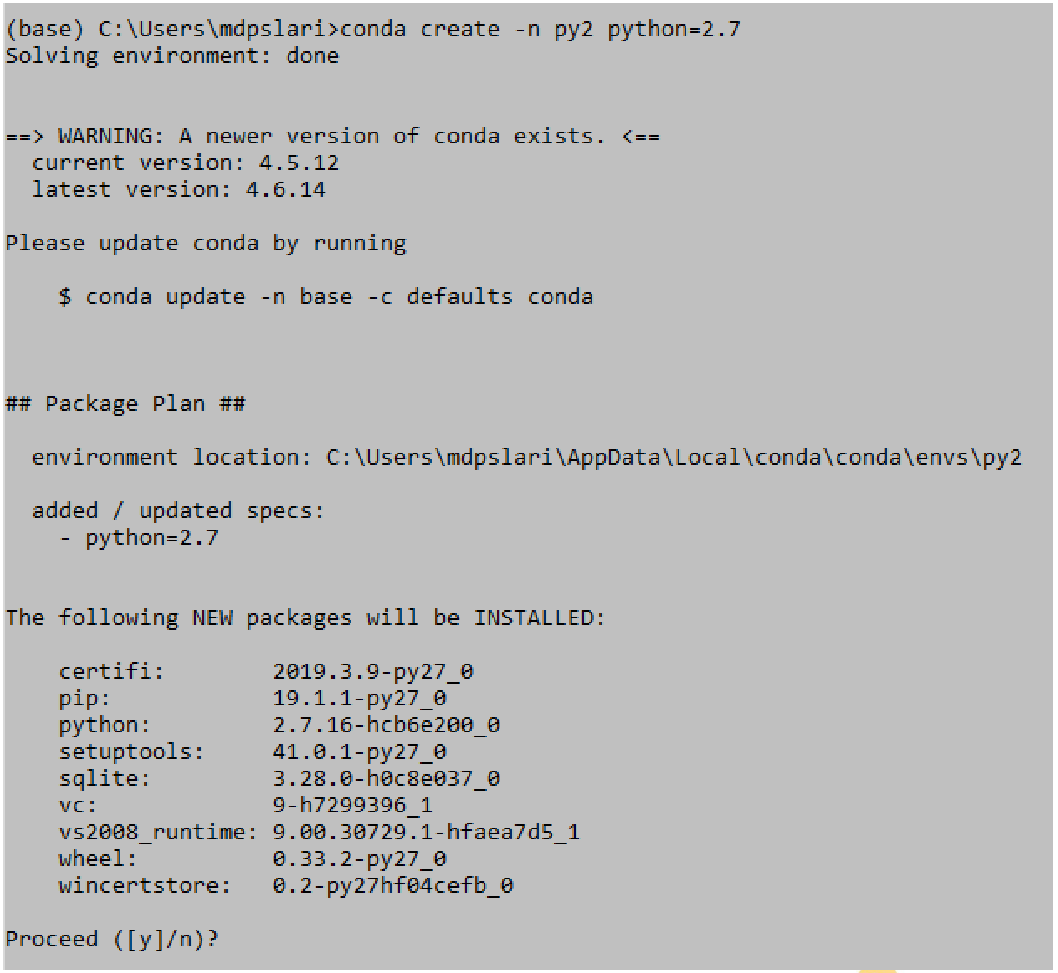


```
(NewEnv) C:\Users\mdpslari>deactivate

(base) C:\Users\mdpslari>
```

# Create a new environment with a specific version

- create an environment with a specific python version

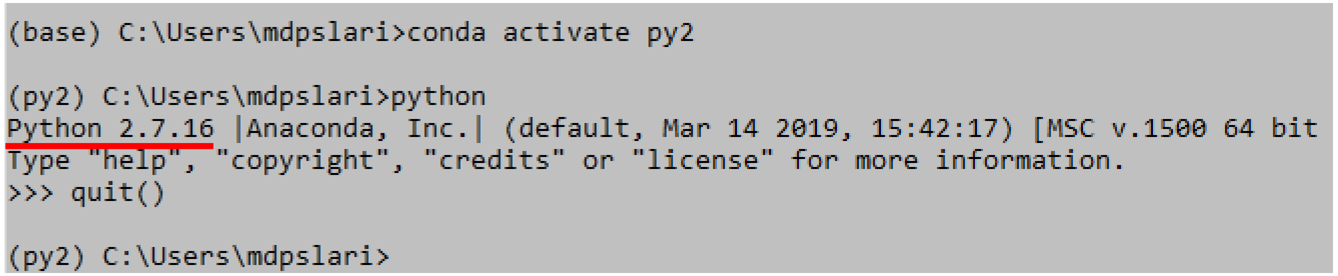
conda create –n py2 python=2.7



```
(base) C:\Users\mdpslari>conda create -n py2 python=2.7
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.12
  latest version: 4.6.14

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: C:\Users\mdpslari\AppData\Local\conda\conda\envs\py2

  added / updated specs:
    - python=2.7


The following NEW packages will be INSTALLED:

    certifi:        2019.3.9-py27_0
    pip:            19.1.1-py27_0
    python:         2.7.16-hcb6e200_0
    setuptools:     41.0.1-py27_0
    sqlite:         3.28.0-h0c8e037_0
    vc:             9-h7299396_1
    vs2008_runtime: 9.00.30729.1-hfaea7d5_1
    wheel:          0.33.2-py27_0
    wincertstore:   0.2-py27hf04cefb_0

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

conda create --name  py310 -c anaconda python=2.7

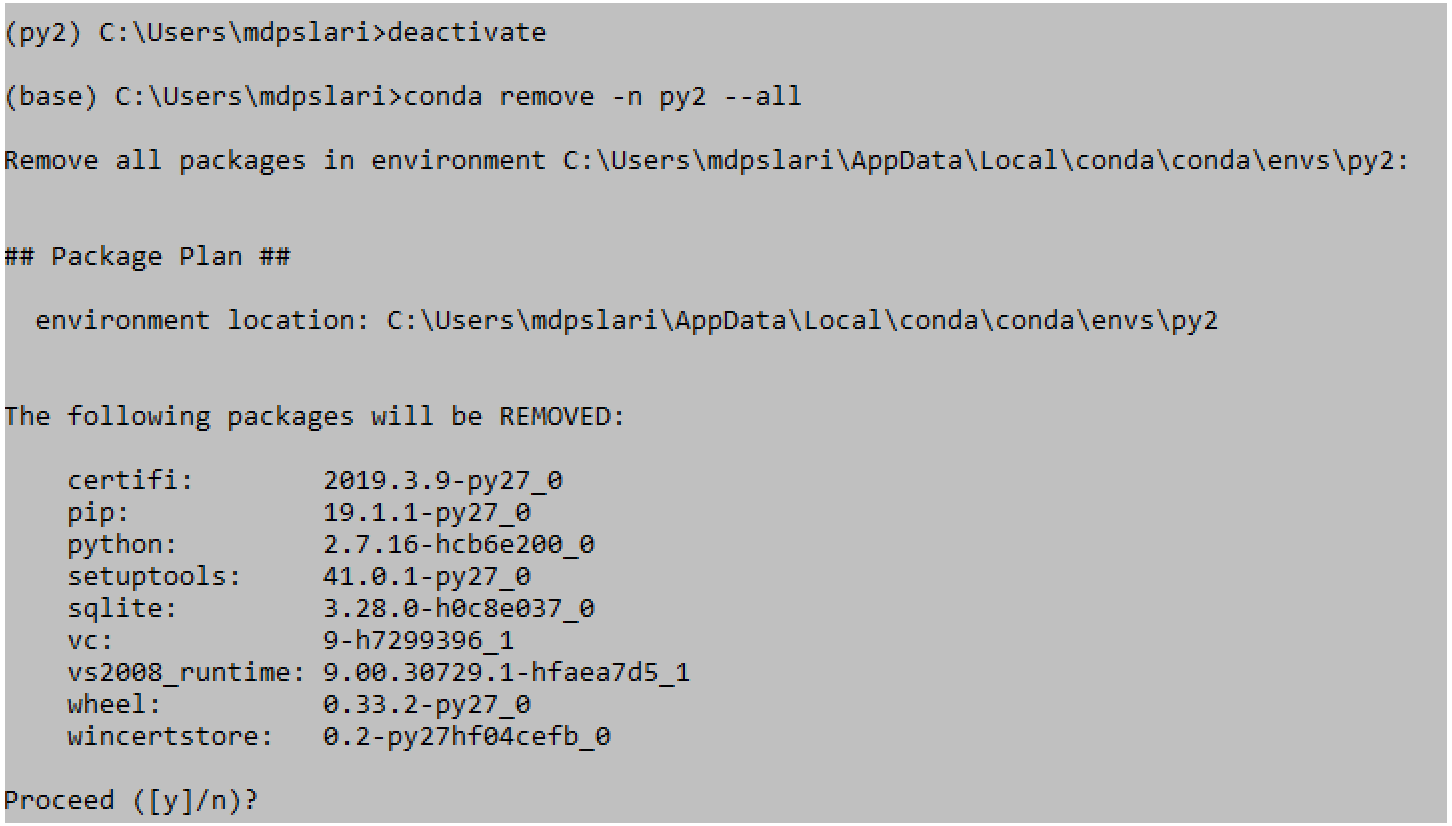
- Check python version



```
(base) C:\Users\mdpslari>conda activate py2

(py2) C:\Users\mdpslari>python
Python 2.7.16 |Anaconda, Inc.| (default, Mar 14 2019, 15:42:17) [MSC v.1500 64 bit
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()

(py2) C:\Users\mdpslari>
```

# Remove an environment

- remove an environment conda remove -n py2 --all
  conda env remove  -n py2

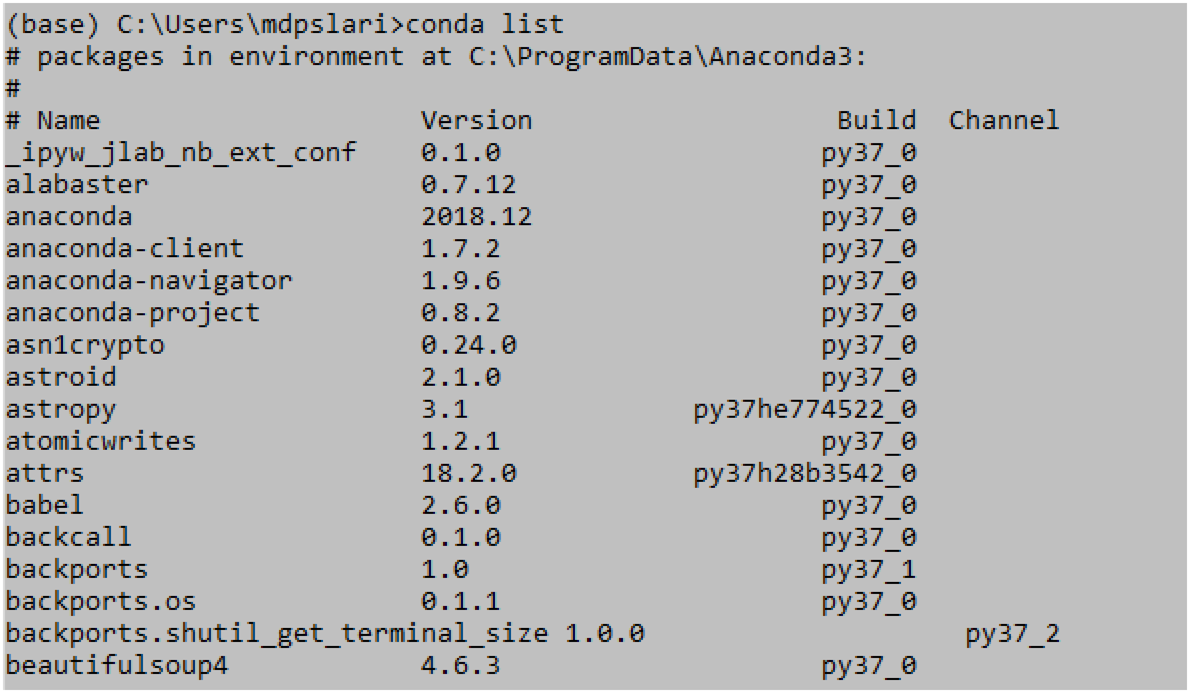


```
(py2) C:\Users\mdpslari>deactivate

(base) C:\Users\mdpslari>conda remove -n py2 --all

Remove all packages in environment C:\Users\mdpslari\AppData\Local\conda\conda\envs\py2:


## Package Plan ##

  environment location: C:\Users\mdpslari\AppData\Local\conda\conda\envs\py2


The following packages will be REMOVED:

    certifi:        2019.3.9-py27_0
    pip:            19.1.1-py27_0
    python:         2.7.16-hcb6e200_0
    setuptools:     41.0.1-py27_0
    sqlite:         3.28.0-h0c8e037_0
    vc:             9-h7299396_1
    vs2008_runtime: 9.00.30729.1-hfaea7d5_1
    wheel:          0.33.2-py27_0
    wincertstore:   0.2-py27hf04cefb_0

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

# Basic Package Management

# List existing packages

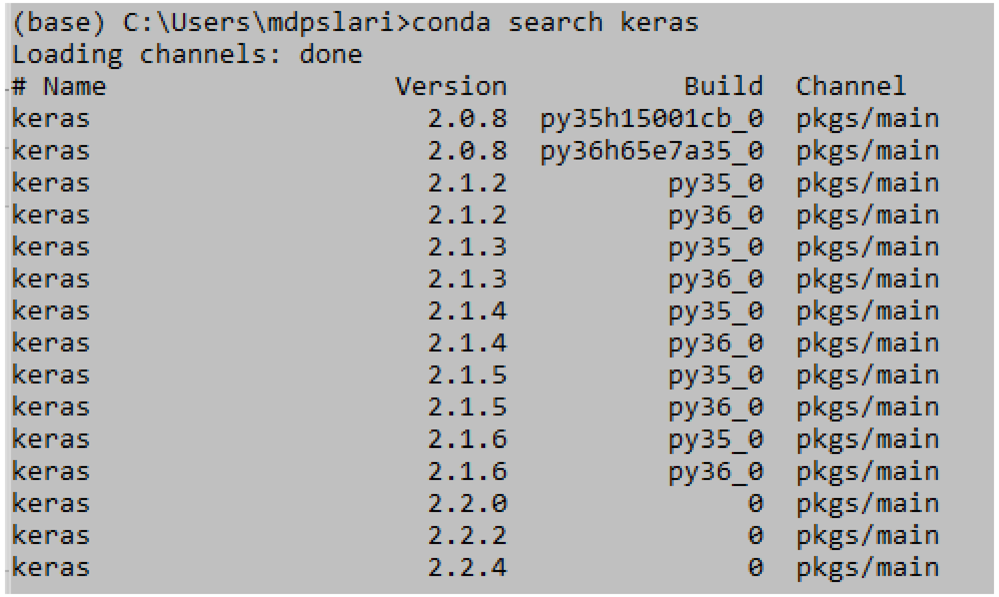
- check the list of installed packages in an environment conda list –n newEnv



```
(base) C:\Users\mdpslari>conda list
# packages in environment at C:\ProgramData\Anaconda3:
#
# Name                    Version                   Build  Channel
_ipyw_jlab_nb_ext_conf    0.1.0                    py37_0
alabaster                 0.7.12                   py37_0
anaconda                  2018.12                  py37_0
anaconda-client           1.7.2                    py37_0
anaconda-navigator        1.9.6                    py37_0
anaconda-project          0.8.2                    py37_0
asn1crypto                0.24.0                   py37_0
astroid                   2.1.0                    py37_0
astropy                   3.1              py37he774522_0
atomicwrites              1.2.1                    py37_0
attrs                     18.2.0           py37h28b3542_0
babel                     2.6.0                    py37_0
backcall                  0.1.0                    py37_0
backports                 1.0                      py37_1
backports.os              0.1.1                    py37_0
backports.shutil_get_terminal_size 1.0.0                    py37_2
beautifulsoup4            4.6.3                    py37_0
```

# Searching and Installing Packages

- search for a specific package **conda search keras**

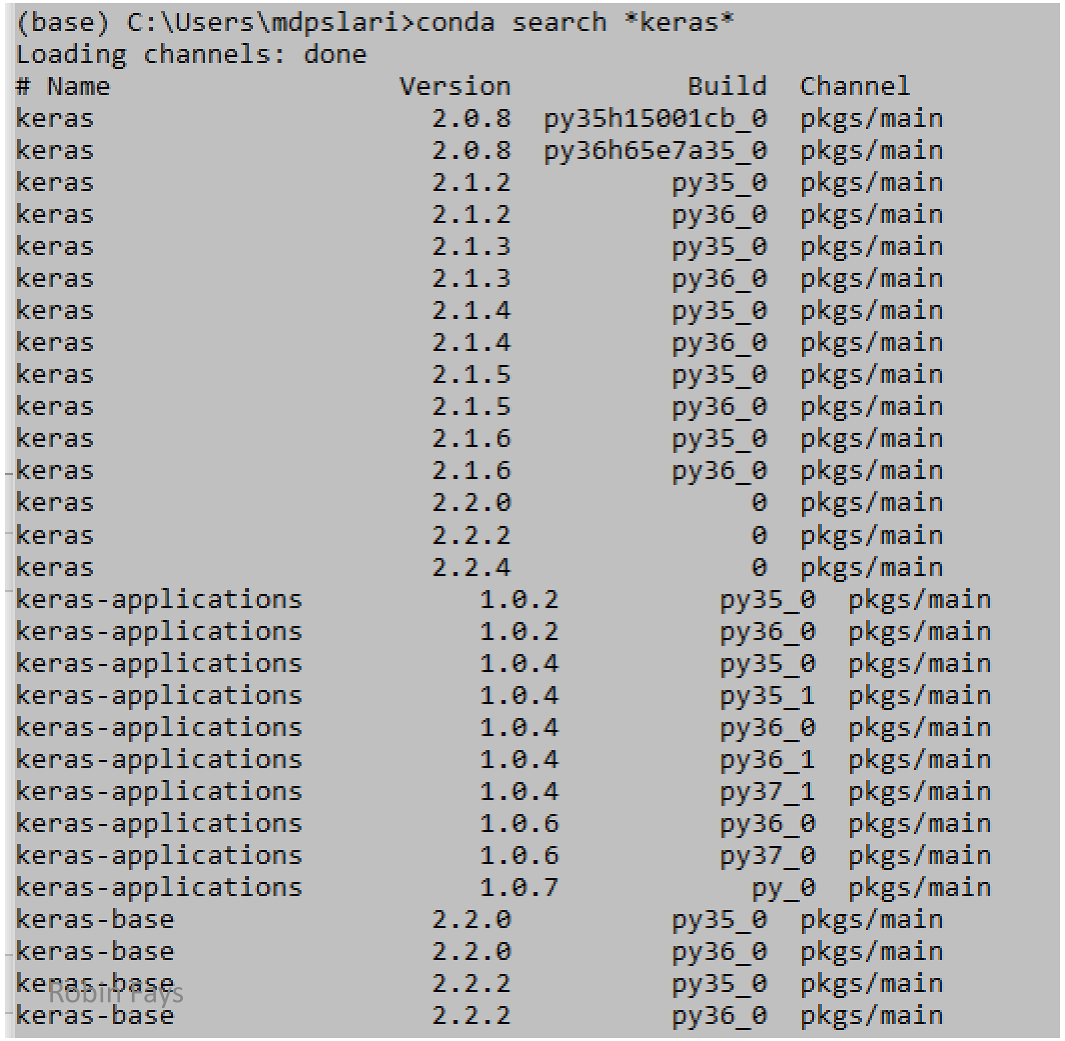


```
(base) C:\Users\mdpslari>conda search keras
Loading channels: done
# Name                       Version           Build  Channel
keras                          2.0.8  py35h15001cb_0  pkgs/main
keras                          2.0.8  py36h65e7a35_0  pkgs/main
keras                          2.1.2          py35_0  pkgs/main
keras                          2.1.2          py36_0  pkgs/main
keras                          2.1.3          py35_0  pkgs/main
keras                          2.1.3          py36_0  pkgs/main
keras                          2.1.4          py35_0  pkgs/main
keras                          2.1.4          py36_0  pkgs/main
keras                          2.1.5          py35_0  pkgs/main
keras                          2.1.5          py36_0  pkgs/main
keras                          2.1.6          py35_0  pkgs/main
keras                          2.1.6          py36_0  pkgs/main
keras                          2.2.0               0  pkgs/main
keras                          2.2.2               0  pkgs/main
keras                          2.2.4               0  pkgs/main
```

Packages are installed from repositories called **channels** by Conda!

# Searching and Installing Packages

- To perform a broader search, including all packages containing keras conda search  *keras*

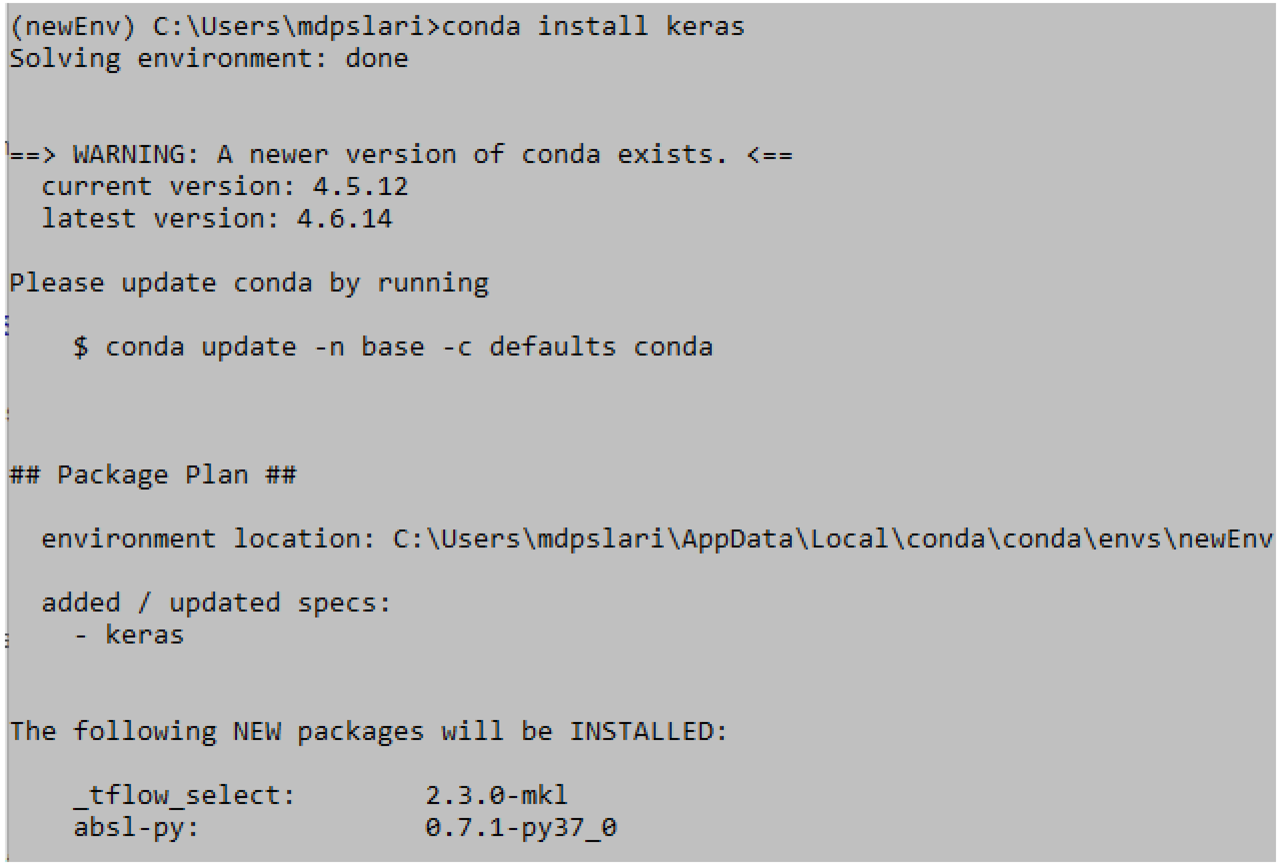


```
(base) C:\Users\mdpslari>conda search *keras*
Loading channels: done
# Name                  Version           Build  Channel
keras                     2.0.8  py35h15001cb_0  pkgs/main
keras                     2.0.8  py36h65e7a35_0  pkgs/main
keras                     2.1.2          py35_0  pkgs/main
keras                     2.1.2          py36_0  pkgs/main
keras                     2.1.3          py35_0  pkgs/main
keras                     2.1.3          py36_0  pkgs/main
keras                     2.1.4          py35_0  pkgs/main
keras                     2.1.4          py36_0  pkgs/main
keras                     2.1.5          py35_0  pkgs/main
keras                     2.1.5          py36_0  pkgs/main
keras                     2.1.6          py35_0  pkgs/main
keras                     2.1.6          py36_0  pkgs/main
keras                     2.2.0               0  pkgs/main
keras                     2.2.2               0  pkgs/main
keras                     2.2.4               0  pkgs/main
keras-applications           1.0.2          py35_0  pkgs/main
keras-applications           1.0.2          py36_0  pkgs/main
keras-applications           1.0.4          py35_0  pkgs/main
keras-applications           1.0.4          py35_1  pkgs/main
keras-applications           1.0.4          py36_0  pkgs/main
keras-applications           1.0.4          py36_1  pkgs/main
keras-applications           1.0.4          py37_1  pkgs/main
keras-applications           1.0.6          py36_0  pkgs/main
keras-applications           1.0.6          py37_0  pkgs/main
keras-applications           1.0.7            py_0  pkgs/main
keras-base                2.2.0          py35_0  pkgs/main
keras-base                2.2.0          py36_0  pkgs/main
keras-base                2.2.2          py35_0  pkgs/main
keras-base                2.2.2          py36_0  pkgs/main
```

# Searching and Installing Packages

- install the package keras in the environment newEnv

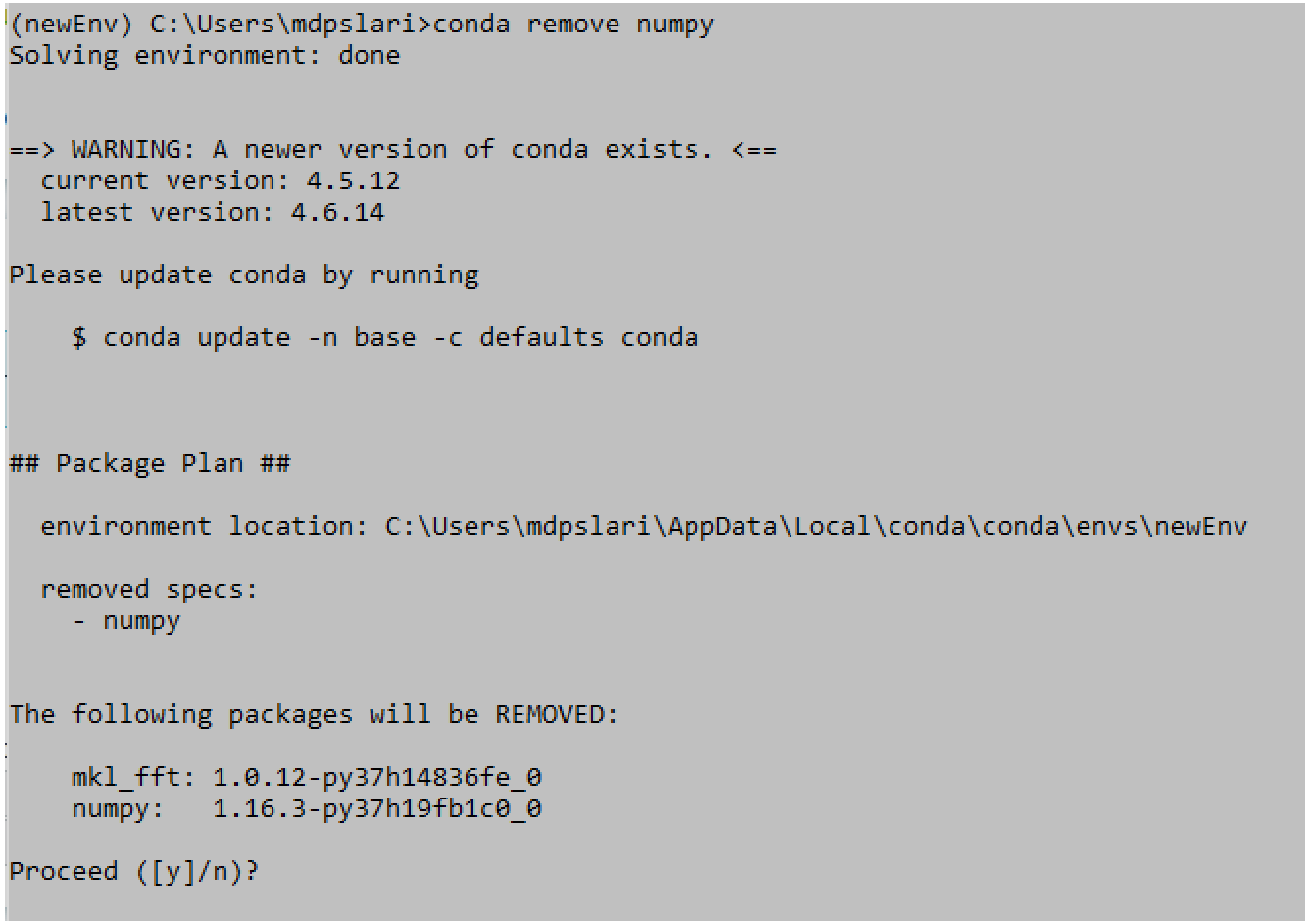
conda install keras



```
(newEnv) C:\Users\mdpslari>conda install keras
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.12
  latest version: 4.6.14

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: C:\Users\mdpslari\AppData\Local\conda\conda\envs\newEnv

  added / updated specs:
    - keras


The following NEW packages will be INSTALLED:

    _tflow_select:        2.3.0-mkl
    absl-py:              0.7.1-py37_0
```

# Updating and Removing Packages

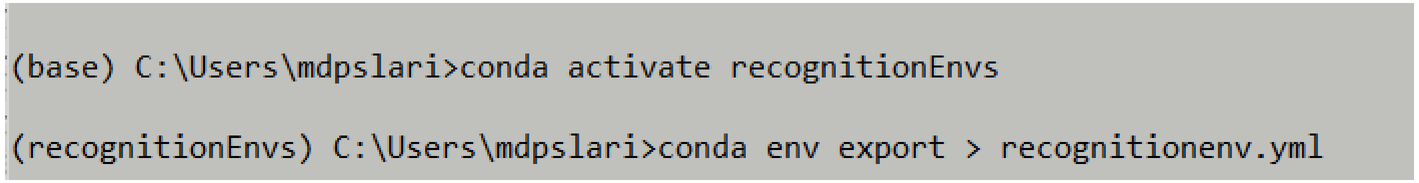
- remove a package conda remove  numpy



```
(newEnv) C:\Users\mdpslari>conda remove numpy
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.12
  latest version: 4.6.14

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: C:\Users\mdpslari\AppData\Local\conda\conda\envs\newEnv

  removed specs:
    - numpy


The following packages will be REMOVED:

    mkl_fft: 1.0.12-py37h14836fe_0
    numpy:   1.16.3-py37h19fb1c0_0

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

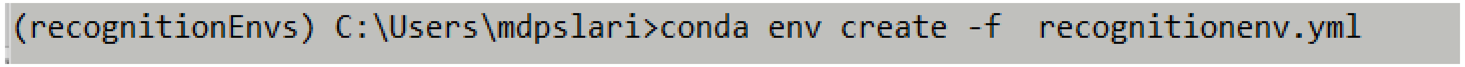
# Sharing environments with others

- Export env to yml



```
(base) C:\Users\mdpslari>conda activate recognitionEnvs

(recognitionEnvs) C:\Users\mdpslari>conda env export > recognitionenv.yml
```

In order to create it on a new system



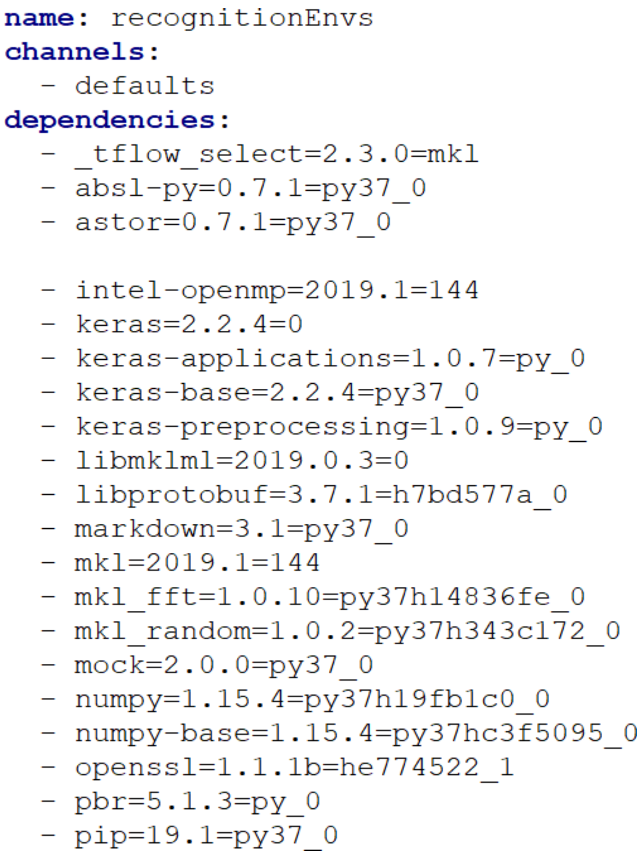
```
(recognitionEnvs) C:\Users\mdpslari>conda env create -f  recognitionenv.yml
```

conda env export > recognitionenv.yml

conda env create -n IntelligentSystems –f Intelligent24.yml

To do : create your environnement from cvcourse

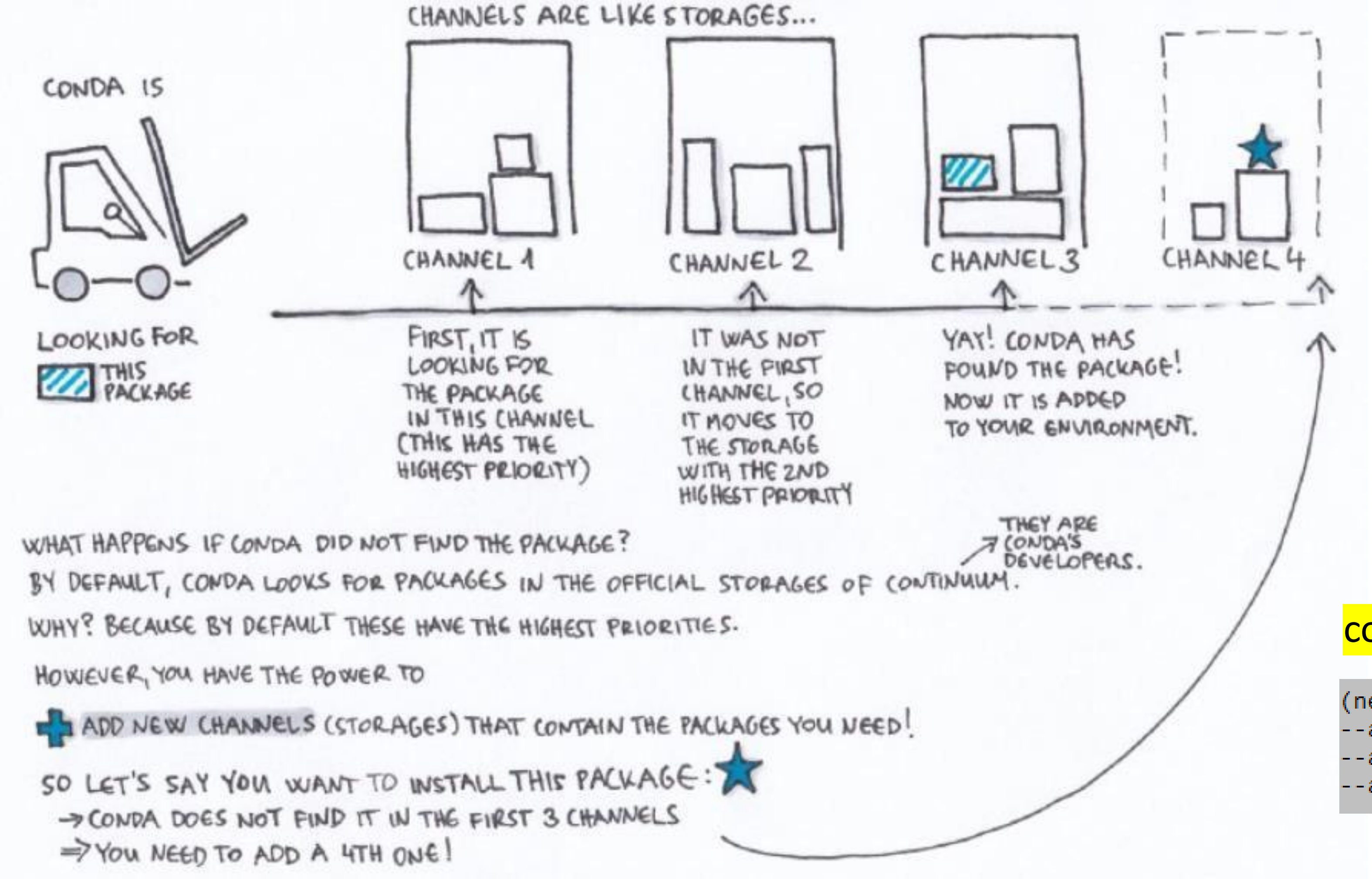
- The YAML file look like this



```
name: recognitionEnvs
channels:
  - defaults
dependencies:
  - _tflow_select=2.3.0=mkl
  - absl-py=0.7.1=py37_0
  - astor=0.7.1=py37_0

  - intel-openmp=2019.1=144
  - keras=2.2.4=0
  - keras-applications=1.0.7=py_0
  - keras-base=2.2.4=py37_0
  - keras-preprocessing=1.0.9=py_0
  - libmklml=2019.0.3=0
  - libprotobuf=3.7.1=h7bd577a_0
  - markdown=3.1=py37_0
  - mkl=2019.1=144
  - mkl_fft=1.0.10=py37h14836fe_0
  - mkl_random=1.0.2=py37h343c172_0
  - mock=2.0.0=py37_0
  - numpy=1.15.4=py37h19fb1c0_0
  - numpy-base=1.15.4=py37hc3f5095_0
  - openssl=1.1.1b=he774522_1
  - pbr=5.1.3=py_0
  - pip=19.1=py37_0
```

# Channels

# Channels?



FIRST, IT IS LOOKING FOR THE PACKAGE IN THIS CHANNEL (THIS HAS THE HIGHEST PRIORITY)

IT WAS NOT IN THE FIRST CHANNEL, SO IT MOVES TO THE STORAGE WITH THE 2ND HIGHEST PRIORITY

YAY! CONDA HAS FOUND THE PACKAGE! NOW IT IS ADDED TO YOUR ENVIRONMENT.

WHAT HAPPENS IF CONDA DID NOT FIND THE PACKAGE?

BY DEFAULT, CONDA LOOKS FOR PACKAGES IN THE OFFICIAL STORAGES OF CONTINUUM.

WHY? BECAUSE BY DEFAULT THESE HAVE THE HIGHEST PRIORITIES.

HOWEVER, YOU HAVE THE POWER TO

✚ ADD NEW CHANNELS (STORAGES) THAT CONTAIN THE PACKAGES YOU NEED!

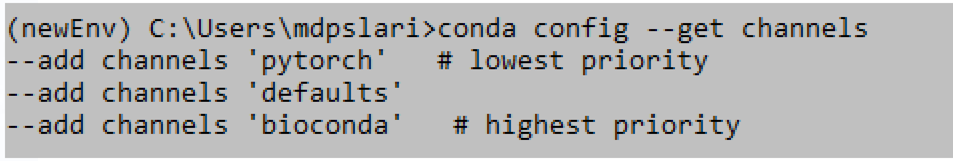
SO LET'S SAY YOU WANT TO INSTALL THIS PACKAGE: ★

→ CONDA DOES NOT FIND IT IN THE FIRST 3 CHANNELS

⇒ YOU NEED TO ADD A 4TH ONE!

Image credit: https://medium.freecodecamp.orgc

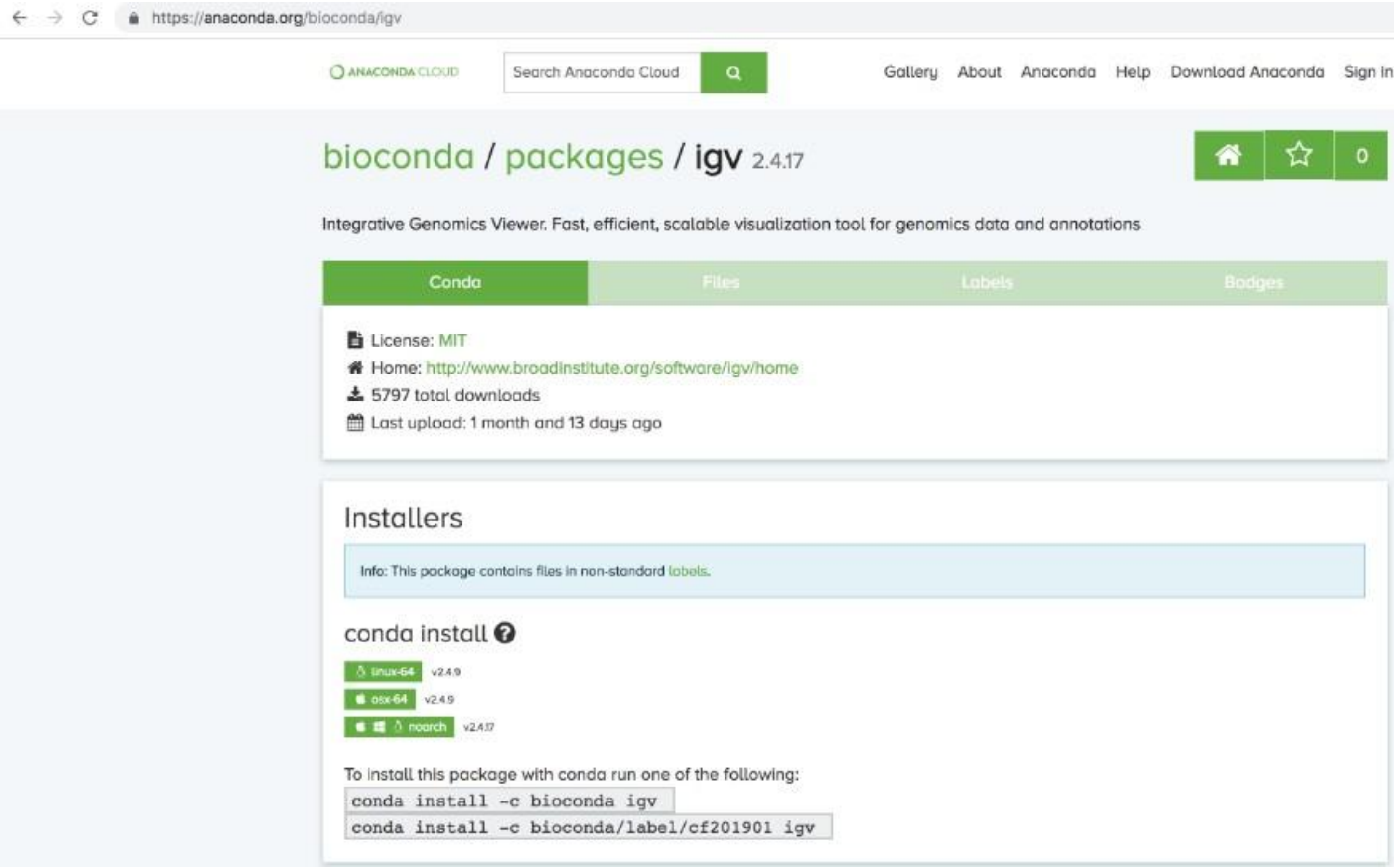
conda config --get channels



```
(newEnv) C:\Users\mdpslari>conda config --get channels
--add channels 'pytorch'   # lowest priority
--add channels 'defaults'
--add channels 'bioconda'   # highest priority
```

# Finding packages

- If you're not sure if your package is available from conda, just **google** it! E.g. "conda IGV"



bioconda / packages / igv 2.4.17

Integrative Genomics Viewer. Fast, efficient, scalable visualization tool for genomics data and annotations

License: MIT
Home: http://www.broadinstitute.org/software/igv/home
5797 total downloads
Last upload: 1 month and 13 days ago

Installers

Info: This package contains files in non-standard labels.

conda install

To install this package with conda run one of the following:

```
conda install -c bioconda igv
conda install -c bioconda/label/cf201901 igv
```

# Adding channel

- You could either install it by specifying the channel

  conda install -c bioconda igv

- Or by adding the channel to the default list of channels to search:

  conda config --add channels bioconda

- You can now install packages from bioconda without specifying:

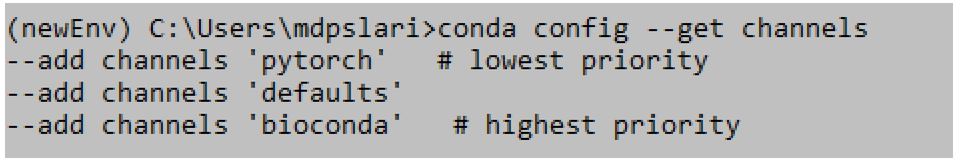
  conda install igv

- You can also add a channel using this command:

  conda config --append channels pytorch

- To check all existing channels :

  conda config --get channels



```
(newEnv) C:\Users\mdpslari>conda config --get channels
--add channels 'pytorch'   # lowest priority
--add channels 'defaults'
--add channels 'bioconda'   # highest priority
```

# Questions

# References

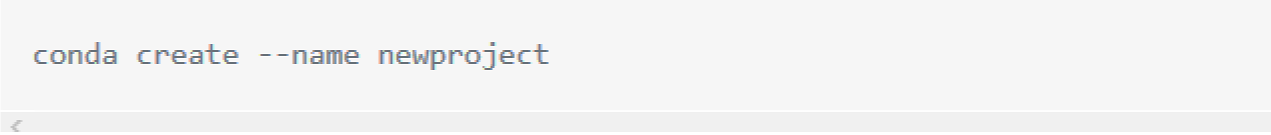
- Tutorial:  https://realpython.com/python-windows-machine-learning-setup/#using-channels
- blog post: https://medium.freecodecamp.org/why-you-need-python-environments-and-how-to-manage-them-with-conda-85f155f4353c
- Documentation on Anaconda:  https://conda.readthedocs.io/en/latest
- Main commands: https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-pkgs.html

More

# pip for virtual environments

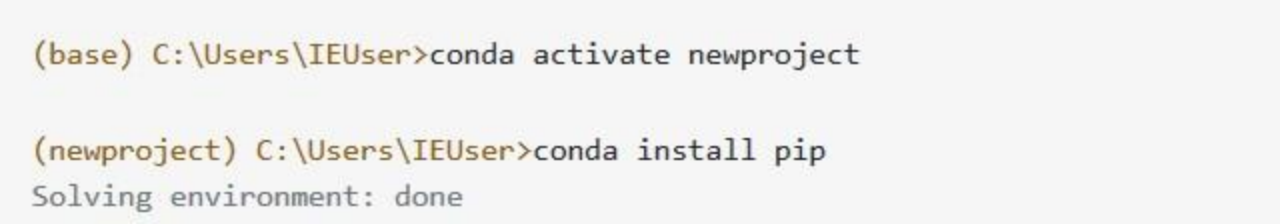
# Using pip Inside Conda Environments

- create a new environment called newproject



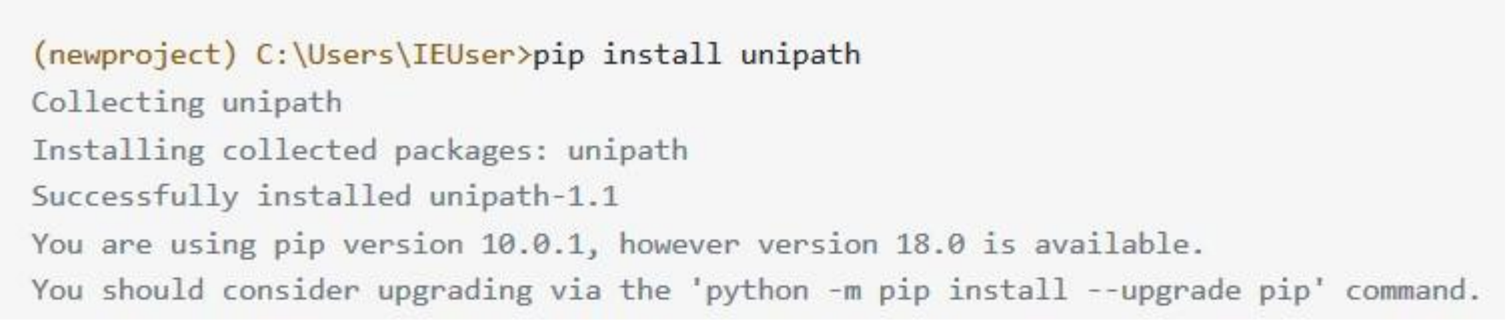
```
conda create --name newproject
```

- activate the environment and install the Conda package pip

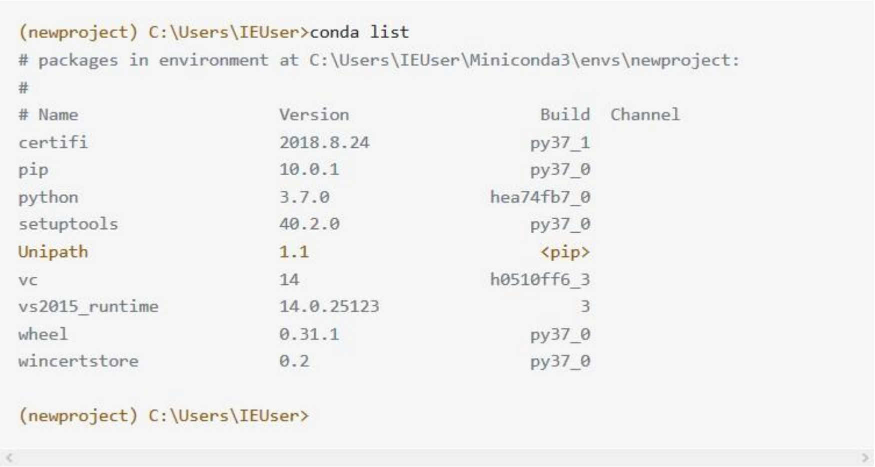


```
(base) C:\Users\IEUser>conda activate newproject

(newproject) C:\Users\IEUser>conda install pip
Solving environment: done
```

- use pip to install the package unipath



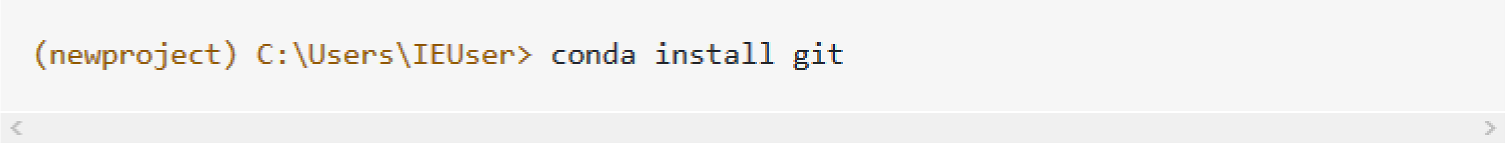
```
(newproject) C:\Users\IEUser>pip install unipath
Collecting unipath
Installing collected packages: unipath
Successfully installed unipath-1.1
You are using pip version 10.0.1, however version 18.0 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```



```
(newproject) C:\Users\IEUser>conda list
# packages in environment at C:\Users\IEUser\Miniconda3\envs\newproject:
#
# Name                    Version                   Build  Channel
certifi                   2018.8.24                py37_1
pip                       10.0.1                   py37_0
python                    3.7.0                hea74fb7_0
setuptools                40.2.0                   py37_0
Unipath                   1.1                       <pip>
vc                        14                   h0510ff6_3
vs2015_runtime            14.0.25123                    3
wheel                     0.31.1                   py37_0
wincertstore              0.2                      py37_0

(newproject) C:\Users\IEUser>
```

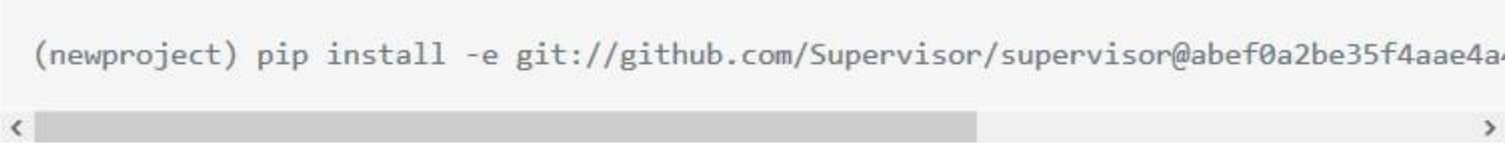
# Install packages from a version control system (VCS) using pip

- As Git is not installed in the newproject environment, you should install it first:



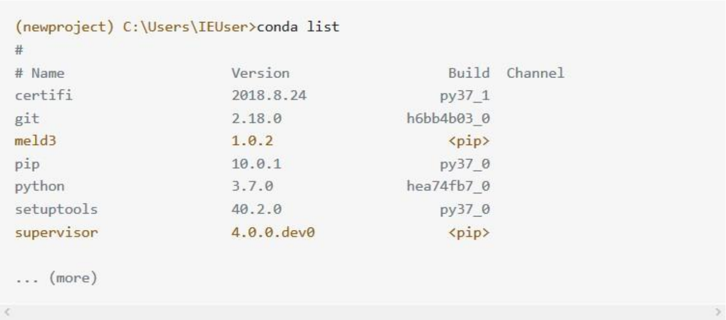
```
(newproject) C:\Users\IEUser> conda install git
```

- Then, install supervisor, using pip to install it from the Git repository:



```
(newproject) pip install -e git://github.com/Supervisor/supervisor@abef0a2be35f4aae4a
```

- After the installation finishes, you can see that supervisor is listed in the installed packages list:

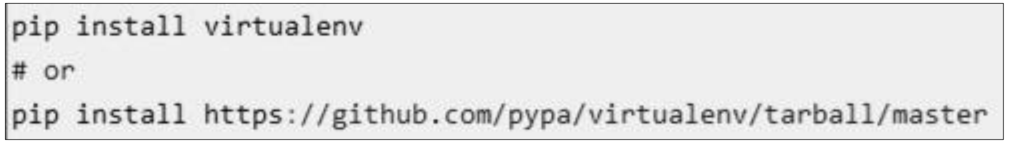


```
(newproject) C:\Users\IEUser>conda list
#
# Name                    Version                    Build  Channel
certifi                   2018.8.24                 py37_1
git                       2.18.0                h6bb4b03_0
meld3                     1.0.2                      <pip>
pip                       10.0.1                    py37_0
python                    3.7.0                 hea74fb7_0
setuptools                40.2.0                    py37_0
supervisor                4.0.0.dev0                 <pip>

... (more)
```

# Install virtualenv on Windows

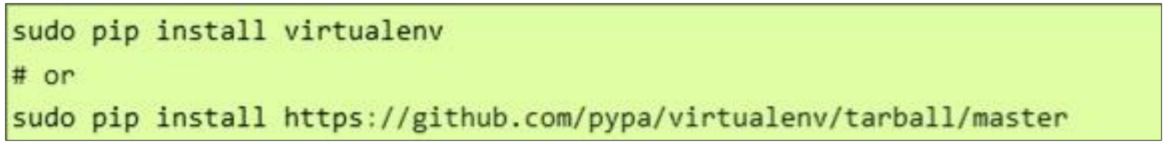
- ➢ If you don't installed pip then visit [https://learnbatta.com/blog/install-pip-on-windows-16/ ](https://learnbatta.com/blog/install-pip-on-windows-16/)and install PIP

- ➢ On Windows, to install virtualenv execute the below commands



```
pip install virtualenv
# or
pip install https://github.com/pypa/virtualenv/tarball/master
```

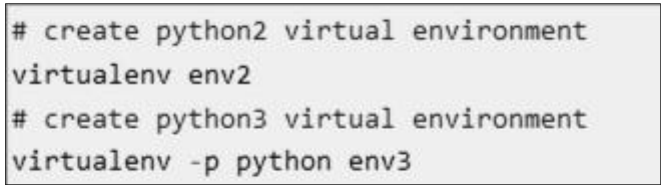
- ➢ On Ubuntu



```
sudo pip install virtualenv
# or
sudo pip install https://github.com/pypa/virtualenv/tarball/master
```

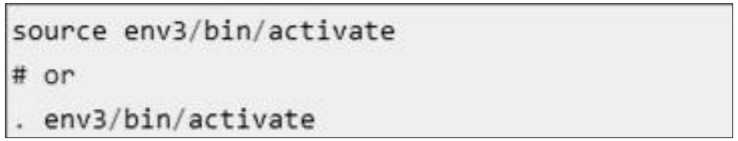
# Usage of virtualenv

- How to create virtual environment?



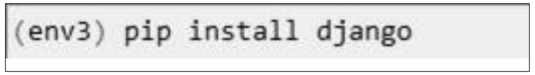
```
# create python2 virtual environment
virtualenv env2
# create python3 virtual environment
virtualenv -p python env3
```

- How to activate virtual environment ?



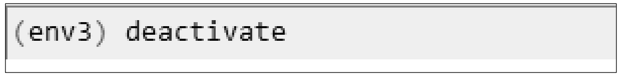
```
source env3/bin/activate
# or
. env3/bin/activate
```

- How to install a python package in virtualenv ?



```
(env3) pip install django
```

- How to deactivate virtual environment ?



```
(env3) deactivate
```