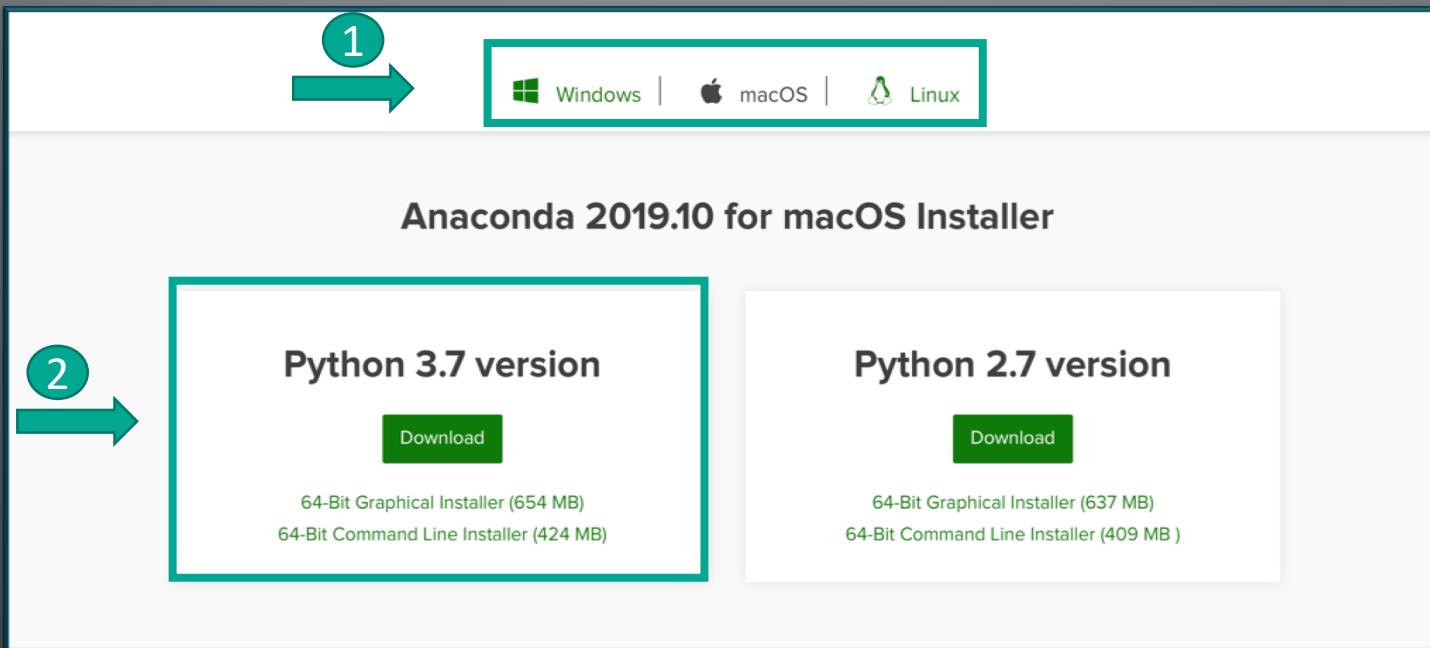


01

Anaconda

- Visit the Anaconda downloads page:
<https://www.anaconda.com/distribution/>



- Open and run the installer.

Install Anaconda3

This package will run a program to determine if the software can be installed.

To keep your computer secure, you should only run programs or install software from a trusted source. If you're not sure about this software's source, click Cancel to stop the program and the installation.

Continue

Welcome to the Anaconda3 Installer

You will be guided through the steps necessary to install this software.

- Introduction
- Read Me
- License
- Destination Select
- Installation Type
- Installation
- PyCharm IDE
- Summary

Go Back Continue

Important Information

Anaconda is the most popular Python data science platform. See <https://www.anaconda.com/downloads/>.

By default, this installer modifies your bash profile to activate the base environment of Anaconda3 when your shell starts up. To disable this, choose "Customize" at the "Installation Type" phase, and disable the "Modify PATH" option. If you decline this option, the executables installed by this installer will not be available on PATH. You will need to use the full executable path to run commands, or otherwise initialize the base environment of Anaconda3 on your own.

To install to a different location, select "Change Install Location..." at the "Installation Type" phase, then choose "Install on a specific disk...", choose the disk you wish to install on, and click "Choose Folder...". The "Install for me only" option will install Anaconda3 to the default location, ~/anaconda3.

The packages included in this installation are:

- alabaster 0.7.12

Print... Save... Go Back Continue

To continue installing the software you must agree to the terms of the software license agreement.

Click Agree to continue or click Disagree to cancel the installation and quit the Installer.

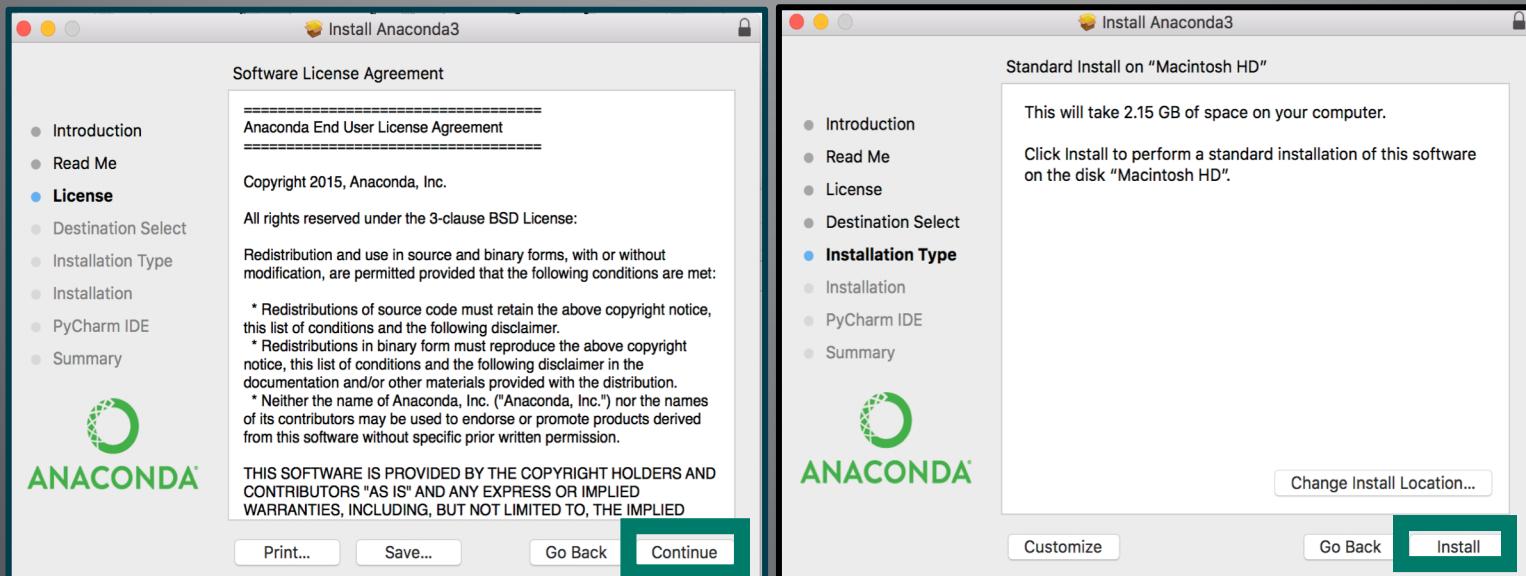
Read License Disagree Agree

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

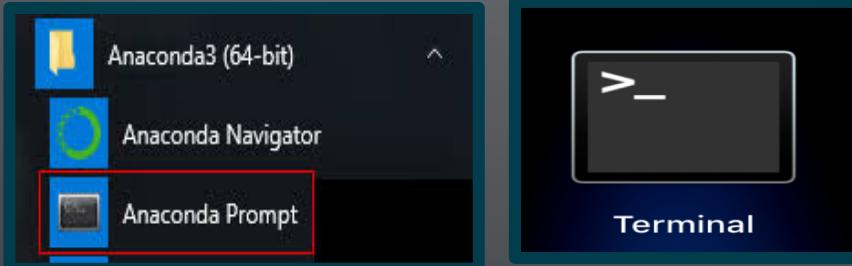


01

Anaconda



- Open the Anaconda Prompt(Windows) or terminal(macOS & Linux) and run some Python code .

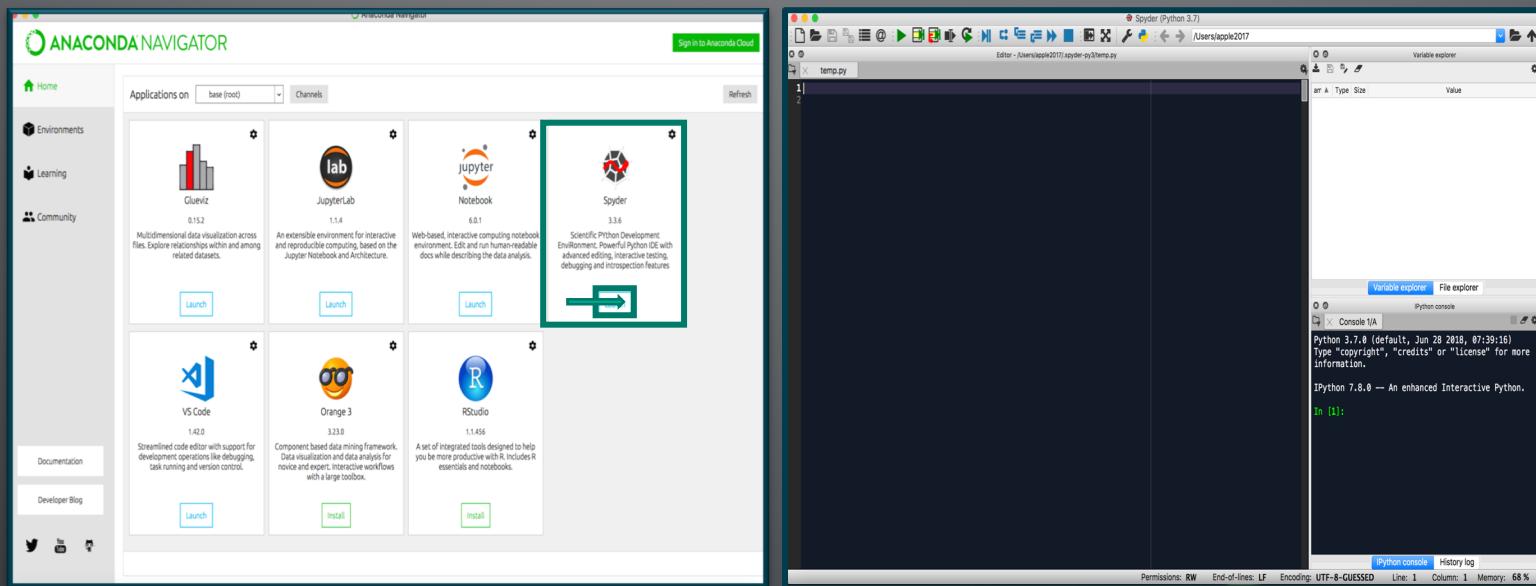




01

Anaconda

- Use Anaconda Navigator to launch an application. Then, create and run a simple Python program with Spyder and Jupyter Notebook.
- On Navigator's Home tab, in the Applications pane on the right, scroll to the Spyder tile and click the Install button to install Spyder then you can jump right to the Launch step..





01

Anaconda

Environment :

- Create separate environments for each purpose of Python and each of these purposes requires specific packages (These packages are not related at all).
- You might have multiple project for multiple purposes and you have all these packages installed globally on your machine now this can cause some problems or at least inconveniences for that we can use so-called virtual environments in python.

Creating an environment with commands :

- Use the terminal or an Anaconda Prompt for the following steps :

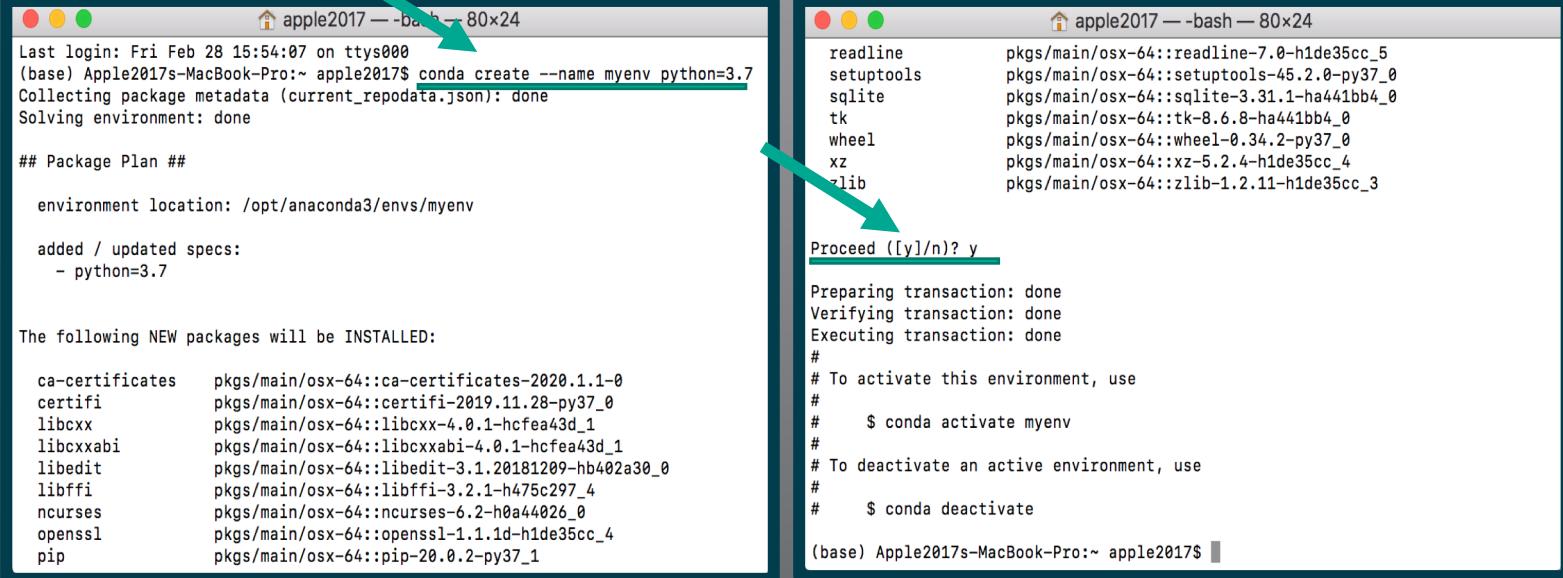
conda create --name myenv

(Replace myenv with the environment name.)

To create an environment with a specific version of Python :

conda create --name myenv python=3.7

When conda asks you to proceed (proceed ([y]/n)?), type y



The image shows two adjacent terminal windows on a Mac OS X system. Both windows have a title bar 'apple2017 — bash — 80x24'.

Terminal Window 1 (Left):

```
Last login: Fri Feb 28 15:54:07 on ttys000
(base) Apple2017s-MacBook-Pro:~ apple2017$ conda create --name myenv python=3.7
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

environment location: /opt/anaconda3/envs/myenv

added / updated specs:
- python=3.7

The following NEW packages will be INSTALLED:

ca-certificates      pkgs/main/osx-64::ca-certificates-2020.1.1-0
certifi               pkgs/main/osx-64::certifi-2019.11.28-py37_0
libcxx                pkgs/main/osx-64::libcxx-4.0.1-hcfea43d_1
libcxxabi              pkgs/main/osx-64::libcxxabi-4.0.1-hcfea43d_1
libedit                pkgs/main/osx-64::libedit-3.1.20181209-hb402a30_0
libffi                 pkgs/main/osx-64::libffi-3.2.1-h475c297_4
ncurses                pkgs/main/osx-64::ncurses-6.2-h0a44026_0
openssl                pkgs/main/osx-64::openssl-1.1.1d-h1de35cc_4
pip                     pkgs/main/osx-64::pip-20.0.2-py37_1
```

Terminal Window 2 (Right):

```
readline          pkgs/main/osx-64::readline-7.0-h1de35cc_5
setuptools        pkgs/main/osx-64::setuptools-45.2.0-py37_0
sqlite            pkgs/main/osx-64::sqlite-3.31.1-ha441bb4_0
tk                 pkgs/main/osx-64::tk-8.6.8-ha441bb4_0
wheel             pkgs/main/osx-64::wheel-0.34.2-py37_0
xz                 pkgs/main/osx-64::xz-5.2.4-h1de35cc_4
zlib              pkgs/main/osx-64::zlib-1.2.11-h1de35cc_3

Proceed ([y]/n)? y

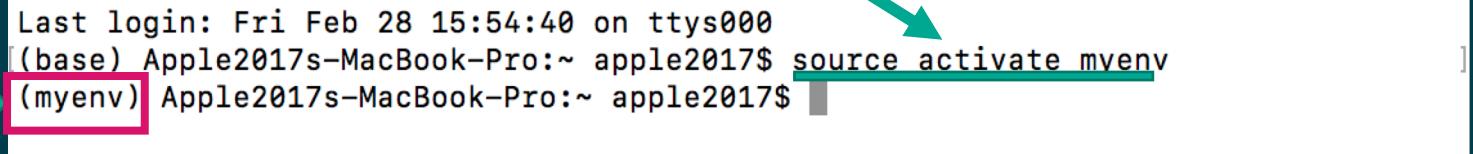
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate myenv
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) Apple2017s-MacBook-Pro:~ apple2017$ ]
```

Activate the new environment to use it:

Windows: **activate myenv**

Linux, macOS: **source activate myenv**



The image shows a single terminal window with a title bar 'apple2017 — bash — 80x24'.

```
Last login: Fri Feb 28 15:54:40 on ttys000
(base) Apple2017s-MacBook-Pro:~ apple2017$ source activate myenv
(myenv) Apple2017s-MacBook-Pro:~ apple2017$ ]
```

A green arrow points from the text 'activated' at the bottom left to the word 'activated' in the terminal output, which is highlighted with a red box.

Installing packages:

- Install a package :

1-pip install PackageName

2-conda install PackageName

```
(myenv) Apple2017s-MacBook-Pro:~ apple2017$ pip install keras
Collecting keras
  Using cached Keras-2.3.1-py2.py3-none-any.whl (377 kB)
Collecting h5py
  Downloading h5py-2.10.0-cp37-cp37m-macosx_10_6_intel.whl (3.0 MB)
    |█████████████████████| 1.4 MB 452 kB/s eta 0:00:04
```



Install a package from a specific channel :

- conda install --channel ChannleName PackageName**

Get a list of all environments(active environment is shown with *):

- conda env list**

List of all packages and versions installed in active environment :

- **conda list**

Show version information for the current active Python :

- **python --version**

Deactivate the current environment :

- Windows: **deactivate**
Linux, macOS: **source deactivate**

You can remove one or more packages from a specific environment :

- **conda remove --name EnvironmentName PackageName**
conda remove --name EnvironmentName PackageName1 PackageName2



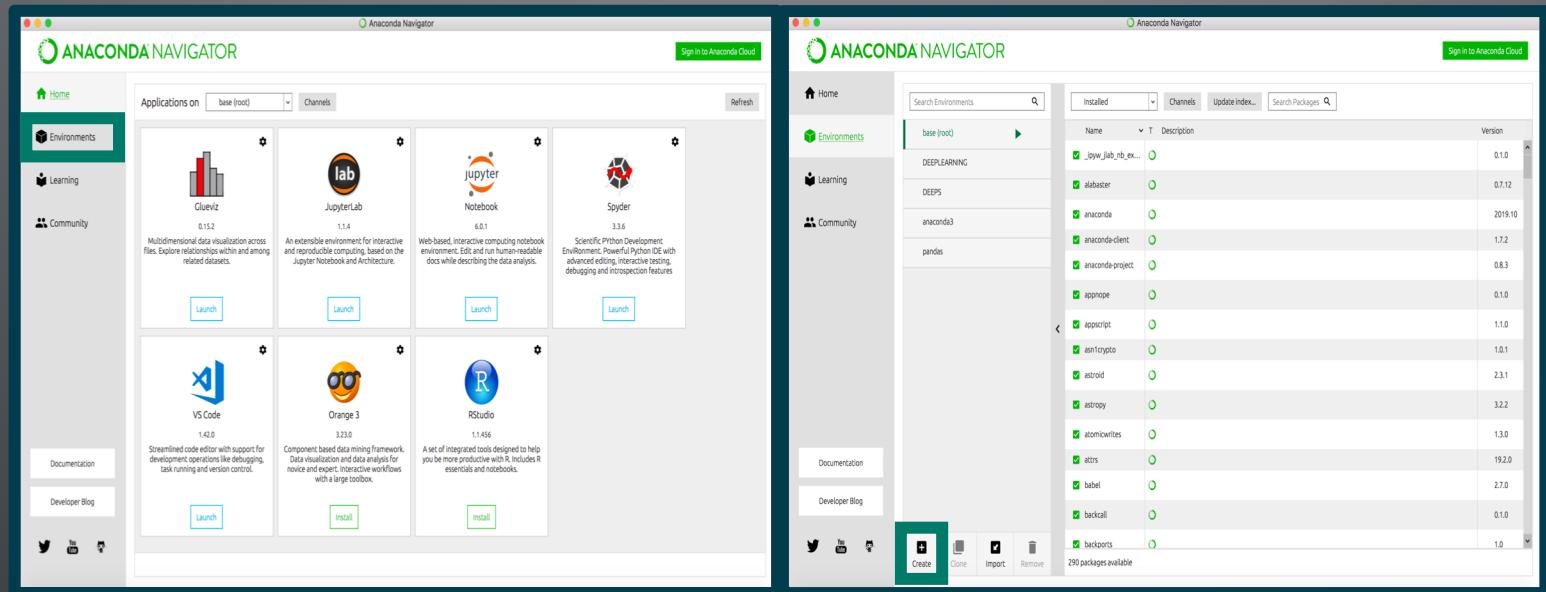
01

Anaconda

You can delete an environment and everything in it :

conda env remove --name EnvironmentName

Creating an environment with Anaconda Navigator :



The image displays two side-by-side screenshots of the Anaconda Navigator application.

Left Screenshot: Shows the "Applications on base (root)" page. It features a grid of icons for various data science tools: Glueviz, JupyterLab, Notebook, Spyder, VS Code, Orange 3, and RStudio. Each tool has a brief description and "Launch" or "Install" buttons.

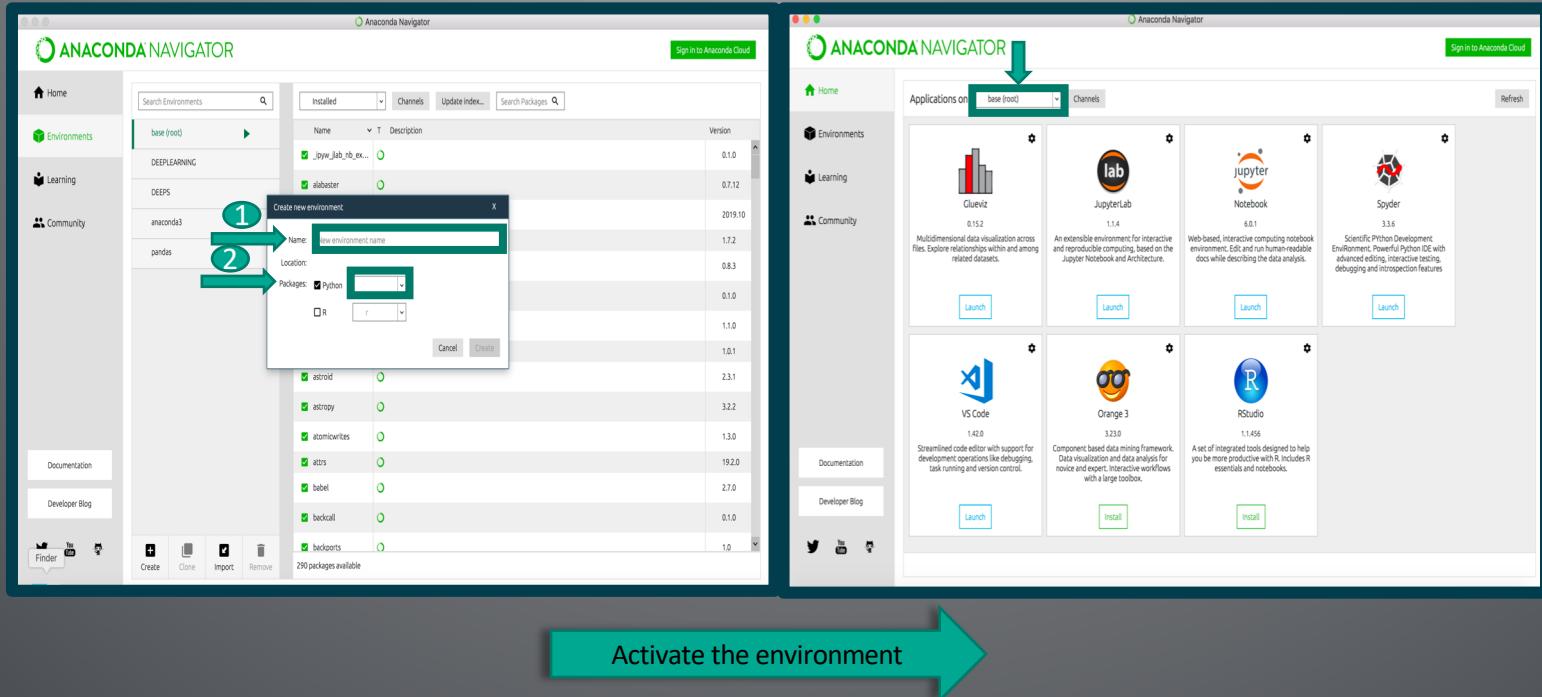
Right Screenshot: Shows the "Environments" page. It lists environments under "base (root)", "DEEPLARNING", "DEEPS", and "anaconda3". Below this is a detailed list of installed packages:

Name	Description	Version
_jupyter_nb_ext		0.1.0
alabaster		0.7.12
anaconda		2019.10
anaconda-client		1.7.2
anaconda-project		0.8.3
apnope		0.1.0
asport		1.1.0
astropy		1.0.1
astroid		2.3.1
astropy		3.2.2
atomicwrites		1.3.0
attrs		19.2.0
babel		2.7.0
backcall		0.1.0
beckports		1.0

At the bottom of the right screenshot, there are buttons for "Create", "Clone", "Import", and "Remove".

01

Anaconda



The image shows two screenshots of the Anaconda Navigator interface.

Left Screenshot: The "Create new environment" dialog box is open. Step 1 highlights the "Name" field where "new_environment_name" is typed. Step 2 highlights the "Packages" dropdown menu, which is currently set to "Python".

Right Screenshot: The "Applications on base (root)" dashboard is displayed. It lists several applications with their icons, versions, and descriptions. A green arrow points from the top of the right screenshot towards the "Launch" button for the Jupyter Notebook application.

Bottom Center: A large green arrow points to the right, containing the text "Activate the environment".

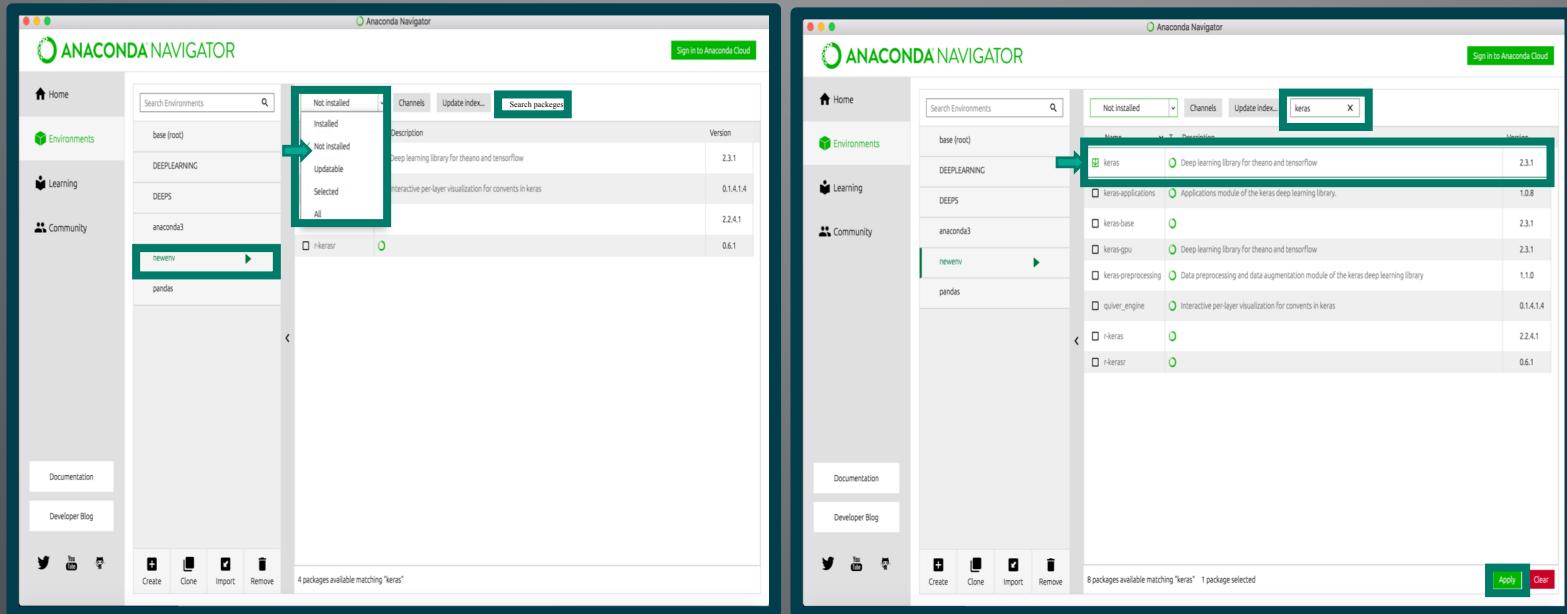
you also might use different versions of Python itself for example you might use Python 3.7 for one project and Python 2.7 for another project.



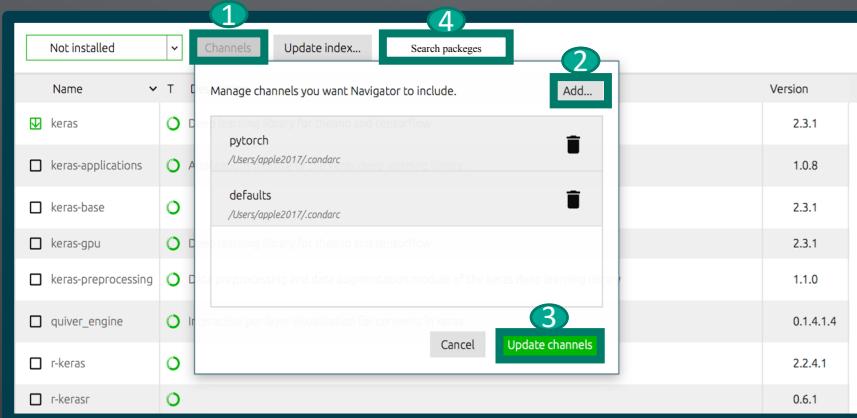
01

Anaconda

Installing packages:



Install a package from a specific channel :



NumPy:

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

PYTORCH:

It's a Python-based scientific computing package targeted at two sets of audiences:

- A replacement for NumPy to use the power of GPUs.
- a deep learning research platform that provides maximum flexibility and speed.

TensorFlow:

- TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embeddings, recurrent neural networks, sequence-to-sequence models for machine translation, natural language processing, and PDE (partial differential equation) based simulations. Best of all, TensorFlow supports production prediction at scale, with the same models used for training.
- TensorFlow allows developers to create dataflow graphs—structures that describe how data moves through a graph, or a series of processing nodes. Each node in the graph represents a mathematical operation, and each connection or edge between nodes is a multidimensional data array, or tensor.

Keras:

- It is capable of running on top of TensorFlow. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.
- Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code.
- In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling.

Scikit-learn:

- Scikit-learn is a free software machine learning library for the python programming language. It features various classification, regression and clustering algorithms including support vector machines, gradient boosting and k-means, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

SciPy:

- SciPy contains modules for optimization, linear algebra, integration ,interpolation algebra,integration ,interpolation , special function, FFT, signal and image processing , ODE solvers and other tasks common in science and engineering. SciPy builds on the Numpy array object and is part of the NumPy stack which includes tools like Matplotlib.

Matplotlib:

- Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.