

Take a conda test drive at bit.ly/tryconda

For full documentation of any command, type the command followed by `--help`.

`conda create --help`

TIP: Many options after two dashes (-) have shortcuts.

`conda create --help` or `conda create -h`

Managing conda and anaconda

<code>conda info</code>	Verify conda is installed, check version #
<code>conda update conda</code>	Update conda package and environment manager to current version
<code>conda update anaconda</code>	Update the anaconda meta package (the library of packages ready to install with <code>conda</code> command)

Managing environments

<code>conda info --envs</code> or <code>conda info -e</code>	Get a list of all my environments, active environment shown with *
<code>conda create --name snowflakes biopython</code> or <code>conda create -n snowflakes biopython</code>	Create an environment and install program(s) <i>TIP: To avoid dependency conflicts, install all programs in the environment (snowflakes) at the same time.</i> <i>TIP: Environments install by default into the envs directory in your conda directory. You can specify a different path; see <code>conda create --help</code> for details.</i>
<code>source activate snowflakes</code> (Linux, Mac) <code>activate snowflakes</code> (Windows)	Activate the new environment to use it <i>TIP: Activate prepends the path to the snowflakes environment.</i>
<code>conda create -n bunnies python=3.4 astroid</code>	Create a new environment, specify Python version

`conda create -n flowers --clone snowflakes` Make exact copy of an environment

`conda remove -n flowers --all` Delete an environment

`conda env export > puppies.yml`
`conda env create -f puppies.yml`
Save current environment to a file
Load environment from a file

Managing Python

<code>conda search --full-name python</code> or <code>conda search -f python</code>	Check versions of Python available to install
<code>conda create -n snakes python=3.4</code>	Install different version of Python in new environment
<code>source activate snakes</code> (Linux, Mac) <code>activate snakes</code> (Windows)	Switch to the new environment that has a different version of Python <i>TIP: Activate prepends the path to the snakes environment.</i>

Managing .condarc configuration

<code>conda config --get</code>	Get all keys and values from my .condarc file
<code>conda config --get channels</code>	Get value of the key channels from .condarc file
<code>conda config --add channels pandas</code>	Add a new value to channels so conda looks for packages in this location

Managing packages, including Python

<code>conda list</code>	View list of packages and versions installed in active environment
<code>conda search beautiful-soup</code>	Search for a package to see if it is available to <code>conda install</code>
<code>conda install -n bunnies beautiful-soup</code>	Install a new package <i>NOTE: If you do not include the name of the new environment (<code>-n bunnies</code>) it will install in the current active environment.</i> <i>TIP: To view list of all packages available through <code>conda install</code>, visit http://docs.continuum.io/anaconda/pkg-docs.html.</i>
<code>conda update beautiful-soup</code>	Update a package in the current environment
<code>conda search --override-channels -c pandas bottleneck</code>	Search for a package in a specific location (i.e. the pandas channel on Anaconda.org) <i>NOTE: Or go to Anaconda.org in the browser and search by package name. This will show the specific channel (owner) through which it is available.</i>
<code>conda install -c pandas bottleneck</code>	Install a package from a specific channel
<code>conda search --override-channels -c defaults beautiful-soup</code>	Search for a package to see if it is available from the Anaconda repository
<code>source activate bunnies</code> (Linux, Mac) <code>activate bunnies</code> (Windows) <code>pip install see</code>	Activate the environment where you want to install a package and install it with pip (included with Anaconda and Miniconda)
<code>conda install iopro accelerate</code>	Install commercial Continuum packages
<code>conda skeleton pypi pyinstrument</code> <code>conda build pyinstrument</code>	Build a Conda package from a Python Package Index (PyPI) Package

Remove packages, environments, or channels

<code>conda remove --name bunnies beautiful-soup</code>	Remove one package from any named environment
<code>conda remove beautiful-soup</code>	Remove one package from the active environment
<code>conda remove --name bunnies beautiful-soup astroid</code>	Remove multiple packages from any environment
<code>conda remove --name snakes --all</code>	Remove an environment

More resources

Free community support	https://groups.google.com/a/continuum.io/forum/#!forum/anaconda
Online documentation	http://conda.pydata.org/docs/
Paid support options	http://continuum.io/support
Continuum onsite training courses	https://store.continuum.io/cshop/training/
Continuum consulting services	http://continuum.io/consulting/