Conda user cheat sheet

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

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

Managing environments

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

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

Managing .condarc configuration

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

Managing packages, including Python

`conda list`	View list of packages and versions installed in active environment
`conda search beautiful-soup`	Search for a package to see if it is available to conda install
`conda install -n bunnies beautiful-soup`	Install a new package

NOTE: *If you do not include the name of the new environment (`-n bunnies`) it will install in the current active environment.*

TIP: *To view list of all packages available through* **conda install**, *visit http://docs.continuum.io/anaconda/pkg-docs*

`conda update beautiful-soup`	Update a package in the current environment
`conda search --override-channels -c pandas bottleneck`	Search for a package in a specific location (the pandas channel on Anaconda.org)

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.*

`conda install -c pandas bottleneck`	Install a package from a specific channel
`conda search --override-channels -c defaults beautiful-soup`	Search for a package to see if it is available from the Anaconda repository
`source activate bunnies` *(Linux, OS X)* `activate bunnies` *(Windows)* `pip install see`	Activate the environment where you want to install a package and install it with pip (included with Anaconda and Miniconda)
`conda install iopro accelerate`	Install commercial Continuum packages
`conda skeleton pypi pyinstrument` `conda build pyinstrument`	Build a Conda package from a Python Package Index (PyPI) Package

Remove packages, environments, or channels

`conda remove --name bunnies beautiful-soup`	Remove one package from any named environment
`conda remove beautiful-soup`	Remove one package from the active environment
`conda remove --name bunnies beautiful-soup astroid`	Remove multiple packages from any environment
`conda remove --name snakes --all`	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://www.continuum.io/training
Continuum consulting services	http://continuum.io/consulting/