Configure and Manage Your Environment with Anaconda

Per the Anaconda docs:

> Conda is an open source package management system and environment management system for installing multiple versions of software packages and their dependencies and switching easily between them. It works on Linux, OS X and Windows, and was created for Python programs but can package and distribute any software.

Overview

Using Anaconda consists of the following:

1. Install `miniconda` on your computer
2. Create a new `conda` environment using this project
3. Each time you wish to work, activate your `conda` environment

Installation

Download the version of `miniconda` that matches your system. Make sure you download the version for Python 3.5.

NOTE: There have been reports of issues creating an environment using miniconda `v4.3.13`. If it gives you issues try versions `4.3.11` or `4.2.12` from here.

	Linux	Mac	Windows
64-bit	64-bit (bash installer)	64-bit (bash installer)	64-bit (exe installer)
32-bit	32-bit (bash installer)		32-bit (exe installer)

Install miniconda on your machine. Detailed instructions:

- **Linux:** http://conda.pydata.org/docs/install/quick.html#linux-miniconda-install (Linux users please also see this warning about how to resolve potential conflicts between the different Python versions installed via Anaconda and ROS.
- **Mac:** http://conda.pydata.org/docs/install/quick.html#os-x-miniconda-install (Mac users please see this thread for possible issues with Jupyter if you have recently updated OSX)
- **Windows:** http://conda.pydata.org/docs/install/quick.html#windows-miniconda-install

Setup your `RoboND` environment.

```
git clone https://github.com/udacity/RoboND-Python-StarterKit.git
cd RoboND-Python-StarterKit
```

If you are on Windows, **rename**
`meta_windows_patch.yml` to
`meta.yml`

Create RoboND. Running this command will create a new `conda` environment that is provisioned with all libraries you need to be successful in this program.
NOTE: if you get an error when you try to run this command that `conda` doesn't exist, try closing and re-opening your terminal window.

```
conda env create -f environment.yml
```

NOTE: If the above command fails due to internet issues or timed out HTTP request then remove the partially built environment using the following command (then run the above `create` command again):

```
conda env remove -n RoboND
conda env create -f environment.yml
```

Verify that the RoboND environment was created in your environments:

```
conda info --envs
```

Cleanup downloaded libraries (remove tarballs, zip files, etc):

```
conda clean -tp
```

Using Anaconda

Now that you have created an environment, in order to use it, you will need to activate the environment. This must be done **each** time you begin a new working session i.e. open a new terminal window.

Activate the `RoboND` environment:

OS X and Linux

```
$ source activate RoboND
```

Windows

Depending on shell either:

```
$ source activate RoboND
```

or

```
$ activate RoboND
```

That's it. Now all of the `RoboND` libraries are available to you.

However, If you are a Linux user with ROS installed (or planning to install ROS natively on your system) please see this warning about how to resolve potential conflicts between the different Python versions installed via Anaconda and ROS.

To exit the environment when you have completed your work session, simply close the terminal window.

Uninstalling

If you ever want to delete or remove an environment

To **delete/remove** the "RoboND" environment:

```
conda env remove -n RoboND
```