**Step 1: Already have Anaconda’s version of Python?**

If you don’t already have the Anaconda distribution of Python, you can download it here. This is easiest, simplest way to get Python up and running with the most popular libraries already loaded

<https://www.continuum.io/downloads>

**Step 2: Install Binstar in order to install psycopg2 for Python to connect to PostgreSQL**

psycopg2 is a python adapter for connecting to PostgreSQL databases

<http://initd.org/psycopg/docs/install.html>

Psycopg2 is the most popular PostgreSQL adapter for the Python programming language. At its core it fully implements the Python DB API 2.0 specifications. Several extensions allow access to many of the features offered by PostgreSQL.

Psycopg2 is a [PostgreSQL](http://www.postgresql.org/) adapter for the [Python](http://www.python.org/) programming language. It is a wrapper for the [libpq](http://www.postgresql.org/docs/current/static/libpq.html), the official PostgreSQL client library.

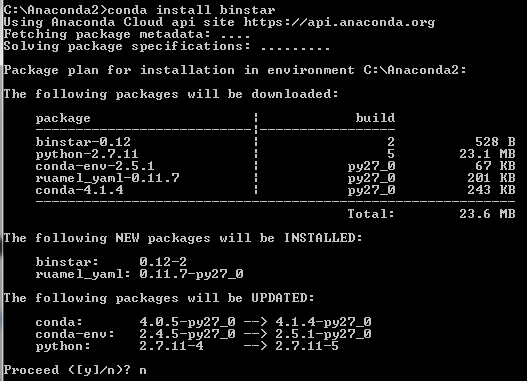
The [**psycopg2**](http://initd.org/psycopg/docs/module.html#module-psycopg2) package is the current mature implementation of the adapter: it is a C extension and as such it is only compatible with [CPython](http://en.wikipedia.org/wiki/CPython). If you want to use Psycopg2 on a different Python implementation (PyPy, Jython, IronPython) there is an experimental [porting of Psycopg for Ctypes](https://github.com/mvantellingen/psycopg2-ctypes), but it is not as mature as the C implementation yet.

The current **psycopg2** implementation supports:

* Python 2 versions from 2.5 to 2.7
* Python 3 versions from 3.1 to 3.4
* PostgreSQL versions from 7.4 to 9.4

**Step 2A: Easiest way to add ‘psycopg2’ is via the Binstar distribution method as it integrates well with Anaconda’s python**

Bring up a command line session for your operating system and bring up the C drive directory where Anaconda files are installed. Below is an example from Windows 7; command “conda install binstar” is executed in a DOS session.



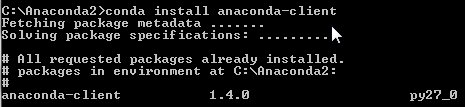
Binstar is likely NOT installed yet but your conda and python software may also be updated in the process as this example shows.

**Below is a sample output from a successful Binstar install.**

**Notice the ending message suggesting to run “conda install anaconda-client”**

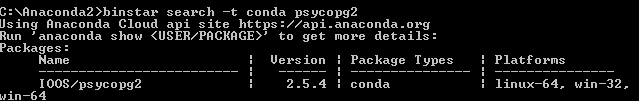


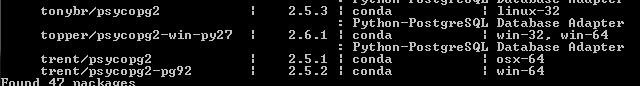
You can try running the suggested command, **“conda install anaconda-client”,**  but nothing will install if you already have the anaconda-client



**Step 2B: Search the Binstar Python distribution network for the ‘psycopg2’ install files**

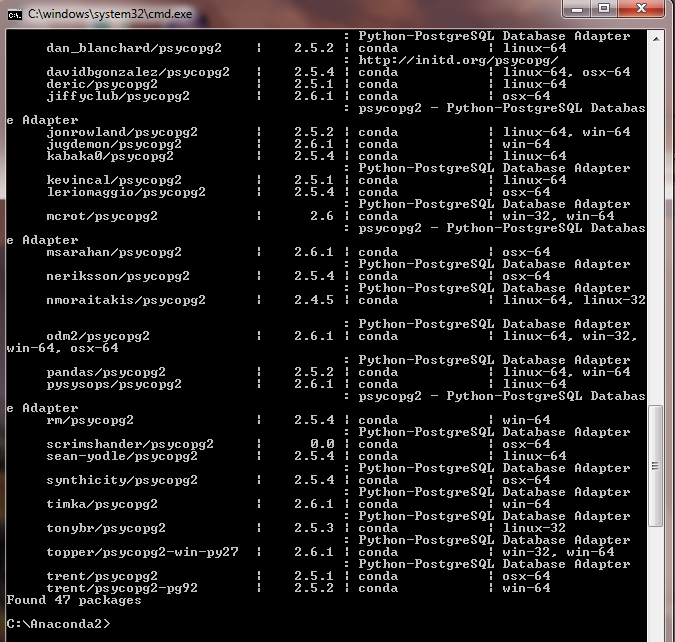
Search for conda packages ‘psychpg2’ via command: “**binstar search -t conda psycopg2**”. Below is an example for windows based install files.



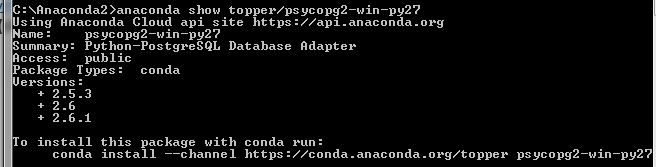


<https://dataperspective.wordpress.com/2014/11/04/installing-psycopg2-for-ipython-on-64-bits-windows7/>

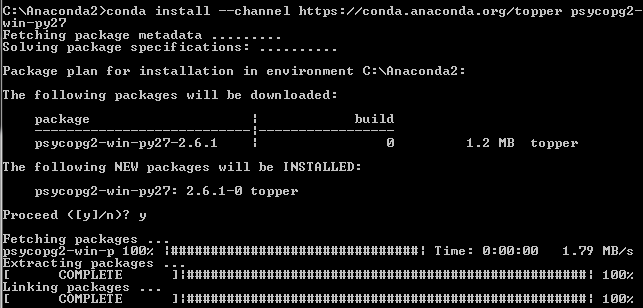
**Notice the install files topper/psycopg2-win-py27, version 2.6.1 which is compatible with win-32 and win-64. You may have to find a Mac OS version. Below the pkg ‘msarahan/psycopg2’ looks like a good version for Max OS.**



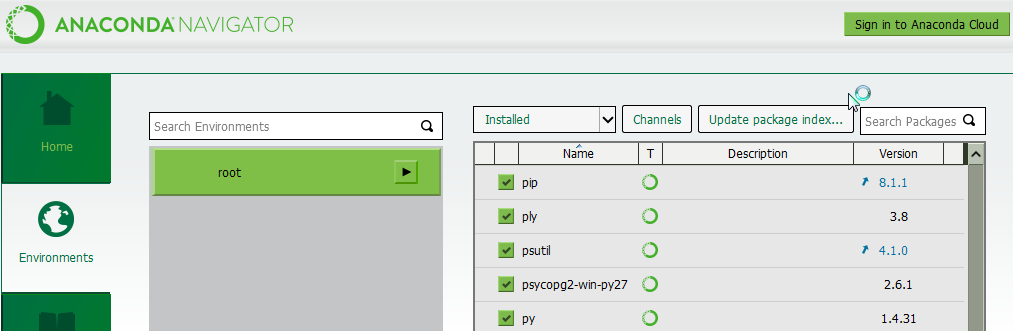
**use ‘anaconda show user/pkg’ to get more installation details on the exact command to execute and load your ‘psycopg2’ pkg. Below was the package used for windows 7.**



**Step 2C: Install the Binstar ‘psycopg2’ files appropriate for your pc**



Bring up Anaconda Navigator and you should see the psycopg2 package was installed successfully and is recognized by Anaconda



**Step 2D: Import the ‘psycopg2’ and try to use it to connect to a PostgreSQL database in a Python session of your choice**

import psycopg2

try:

conn = psycopg2.connect("dbname='get\_data' user='212569352' host='3.48.35.24' password='xxxx'")

print "Opened database get\_data successfully"

except:

print "I am unable to connect to the GPDB Dev database"

cur = conn.cursor()

try:

cur.execute("SELECT \* from public.cp\_reshapewheel\_yt")

except:

print "I can't SELECT from public.cp\_reshapewheel\_yt"

rows = cur.fetchall()

print "Operation done successfully";

i = 0

for row in rows:

print 'road number = ', row[0], 'workorder = ', row[1], 'axel txt = ', row[3], 'min rim = ', row[4], 'min flange height =', row[5]

i += 1

if i > 10:

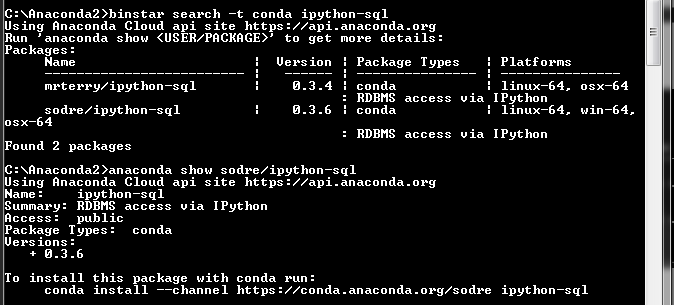
break

conn.close()

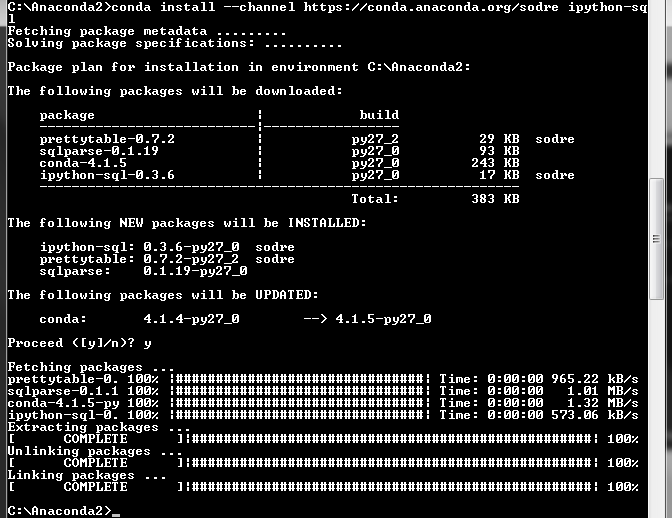
**Step 3: There is 2nd helpful library for Ipython SQL Extensions**

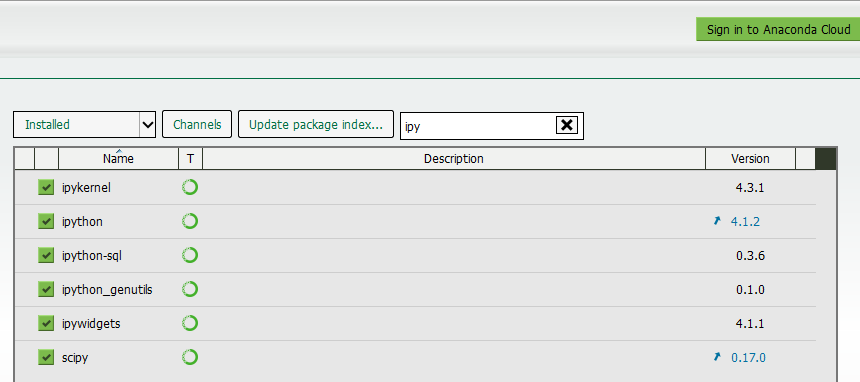
Finding and installing ipython-sql extension. There are install files for Windows, OSX and Linux

In your OS command session : **binstar search -t conda ipython-sql**



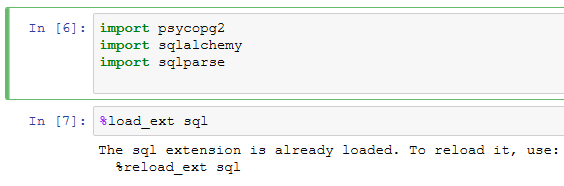
Once you find a pkg/library of files you want for your PC’s Operating System, use the SHOW command to get exact command line to install that library. ‘**anaconda show sodre/ipython-sql**’ will bring up exact INSTALL cmd below



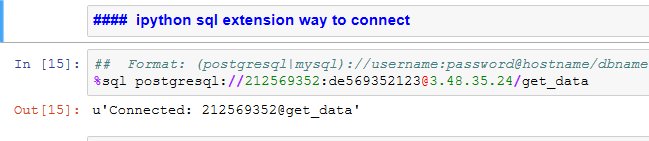


Use Anaconda Navigator and verify that ipython-sql extension is now installed

Below is the import and %load\_est sql code to begin using the ipython-sql extensions that make it easier to run SQL directly inside a ipython notebook.

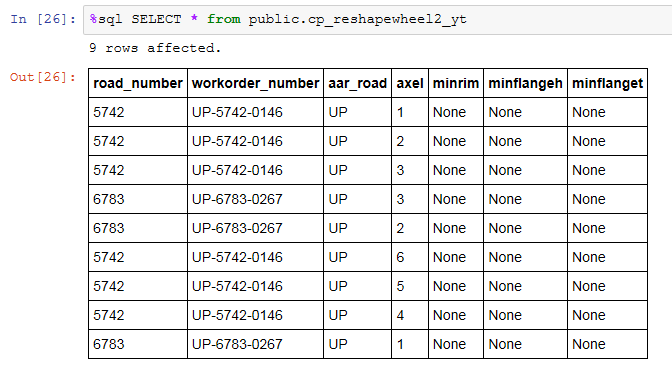


This is a simpler way to connect to PostgreSQL via the ipython-sql extensions



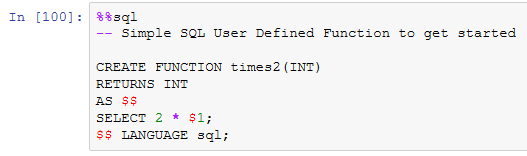
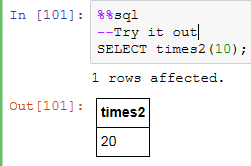
## Format: (postgresql|mysql)://username:password@hostname/dbname

%sql postgresql://212569352:de569352123@3.48.35.24/get\_data



Now you can directly execute SQL statements without using a connection cursor

**%%sql ipython magic extension to execute user defined functions in SQL with PL/python**

<https://ep2013.europython.eu/media/conference/slides/plpython-python-inside-the-postgresql-rdbms.pdf>

<http://nbviewer.jupyter.org/github/ihuston/plpython_examples/blob/master/simple_sql_example_notebook.ipynb>

<https://wiki.postgresql.org/wiki/Psycopg2_Tutorial>

<https://www.postgresql.org/docs/9.5/static/plpython.html>

<http://gpdb.docs.pivotal.io/4370/ref_guide/extensions/pl_python.html>

<http://www.tutorialspoint.com/postgresql/postgresql_python.htm>

<https://vimeo.com/79558274> (Python Powered Data Science at Pivotal - Ian Huston, Srivatsan Ramanujam)

<https://plot.ly/ipython-notebooks/survival-analysis-r-vs-python/>

<https://www.youtube.com/watch?v=Omrp4w_C5nE>