The first thing you want to do is download Python. The latest (as of this writing, at least) is 3.9.5. You can download it from here:

https://www.python.org/downloads/

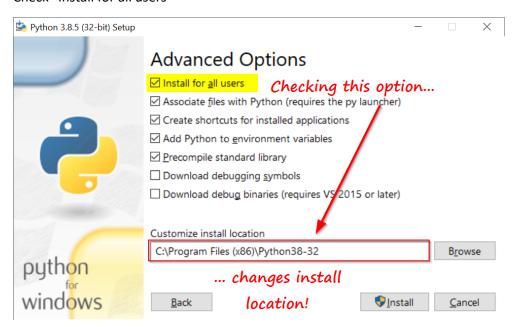
Download Python.

Sample Screenshot:



Check the circled box, and choose "Customize Installation"

Check "Install for all users"



Open a command terminal (DOS prompt)

Run:

C:\ pip3 install verticapy

Note: this can fail if it can't find 'pip3' in the PATH. If that happens, this option should work (and accomplishes the same thing):

C:\ py -3 -mpip install verticapy

```
Sample Screenshot:
 Select Command Prompt
                                                                                                                                                         :\Windows\System32>py -3 -mpip install verticapy
 Oblecting verticapy

Downloading verticapy-0.6.1-py3-none-any.whl (1.2 MB)

1.2 MB 3.3 MB/s
   Downloading matplotlib-3.4.2-cp39-cp39-win_amd64.whl (7.1 MB)
                                                   7.1 MB ...
 ollecting vertica-python>=0.11.0

Downloading vertica_python-1.0.1-py2.py3-none-any.whl (168 kB)
  Downloading numpy-1.20.3-cp39-cp39-win_amd64.whl (13.7 MB)
  ollecting scipy>=1.0.0
Downloading scipy-1.6.3-cp39-cp39-win_amd64.whl (32.7 MB)
                                                   32.7 MB 6.8 MB/s
  Downloading Pillow-8.2.0-cp39-cp39-win_amd64.whl (2.2 MB)
  ollecting python-dateutil>=2.7

Downloading python_dateutil-2.8.1-py2.py3-none-any.whl (227 kB)
 Collecting kiwisolver>=1.0.1
Down<u>loading kiwisolver-1.3.1-cp39-cp39-win</u>amd64.whl (51 kB)
                                                   | 51 kB 201 kB/s
  Downloading cycler-0.10.0-py2.py3-none-any.whl (6.5 kB) ollecting pyparsing>=2.2.1
Downloading pyparsing-2.4.7-py2.py3-none-any.whl (67 kB)
```

It may be possible to get an error here regarding the "scripts" folder not being in your path, as below:

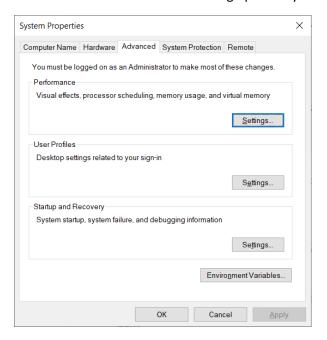
```
Downloading numpy-1.20.3-cp39-cp39-win_amd64.whl (13.7 MB)

| 13.7 MB 6.4 MB/s |
| 13.7 MB 6.4 MB/s |
| 13.7 MB 6.4 MB/s |
| 13.7 MB 6.8 MB/s |
| 13.7 MB 6.
```

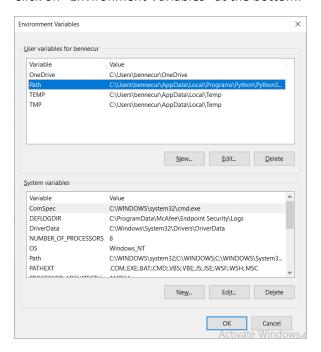
This is apparently due to multiple registry keys that contain trailing nulls in the Windows machine that cause searches to fail. Regardless of the convoluted reason, you can add the scripts folder to your path manually:

Adding Scripts to the PATH:

Go to Start and run, and run "environment variables" or load Control Panel, and search for "environment variables". It will bring up the "System Properties" dialog.



Click on "Environment Variables" at the bottom.



In the top box, click on "Edit" and then add the necessary path(s).

This shouldn't require a restart – but you will need to close the instance of the CMD prompt, and open a new one.

Install JupyterLab. You need not download this – this function works kind of like a "git clone" since it finds the project on Github, and downloads it automatically.

C:\ py -3 -mpip install jupyterlab

Or

C:\ pip3 install jupyterlab

```
Sample Screenshot:

Select Command Prompt

C:\Windows\System32>py -3 -mpip install jupyterlab
Collecting jupyterlab-3.0.15-py3-none-any.whl (8.2 MB)

Downloading jupyterlab-3.0.15-py3-none-any.whl (8.2 MB)

B.2 MB 6.8 MB/s

Collecting jupyter-server~=1.4

Downloading jupyter-server~=1.4

Downloading jupyter-server~=1.4

Downloading jinja2-2.1

Downloading jinja2-2.1

Downloading jinja2-2.1

Downloading jinja2-2.11.3-py2.py3-none-any.whl (125 kB)

| 125 kB ...

Collecting inja2>-2.1

Downloading nbclassic~0.2

Downloading nbclassic~0.2.8-py3-none-any.whl (17 kB)

Collecting tornado-6.1.0

Downloading jupyterlab-server~2.3

Downloading jupyterlab-server~2.5

| 422 kB 6.8 MB/s

Collecting jupyterlab-server~2.5

| 49 kB 2.5 MB/s

Collecting packaging

Downloading packaging-20.9-py2.py3-none-any.whl (40 kB)

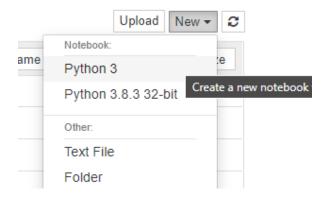
| 40 kB 2.5 MB/s

Collecting MarkupSafe>-0.23
```

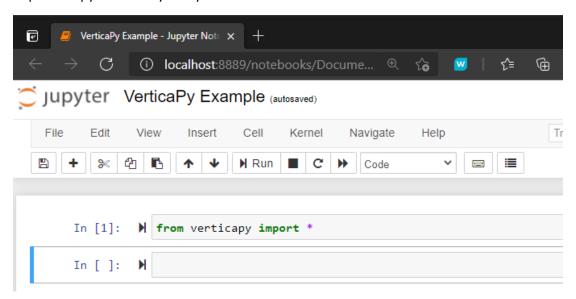
Once Jupyter is installed, Navigate to a suitable directory (Documents, Desktop, etc) and start a notebook with the following command:

```
Administrator: Command Prompt - jupyter notebook
Microsoft Windows [Version 10.0.17763.1879]
(c) 2018 Microsoft Corporation. All rights reserved.
C:\windows\system32>jupyter notebook
[I 15:24:11.570 NotebookApp] The port 8888 is already in use, trying another port.
[W 15:24:11.583 NotebookApp] Terminals not available (error was No module named 'winpty.cyw.
[I 15:24:11.637 NotebookApp] [jupyter_nbextensions_configurator] enabled 0.4.1
[I 15:24:11.826 NotebookApp] JupyterLab extension loaded from c:\users\psingh5\appdata\loca
\python38-32\lib\site-packages\jupyterlab
[I 15:24:11.826 NotebookApp] JupyterLab application directory is c:\users\psingh5\appdata\l
hon\python38-32\share\jupyter\lab
[I 15:24:11.829 NotebookApp] Serving notebooks from local directory: C:\windows\system32
[I 15:24:11.829 NotebookApp] The Jupyter Notebook is running at:
[I 15:24:11.829 NotebookApp] http://localhost:8889/?token=485ebfd5c8ef594ff1ab1beac9554489f
[I 15:24:11.829 NotebookApp] or http://127.0.0.1:8889/?token=485ebfd5c8ef594ff1ab1beac9554
[I 15:24:11.830 NotebookApp] Use Control-C to stop this server and shut down all kernels (t
irmation).
[C 15:24:11.873 NotebookApp]
      To access the notebook, open this file in a browser: file:///C:/Users/PSingh5/AppData/Roaming/jupyter/runtime/nbserver-48304-open.html
      Or copy and paste one of these URLs:
            http://localhost:8889/?token=485ebfd5c8ef594ff1ab1beac9554489fe1ab65aabc9dd4f
       or http://127.0.0.1:8889/?token=485ebfd5c8ef594ff1ab1beac9554489fe1ab65aabc9dd4f
```

The Jupyter hub will open in your browser... If not copy the notebook link from the terminal Create a new notebook from the dropdown towards the top right



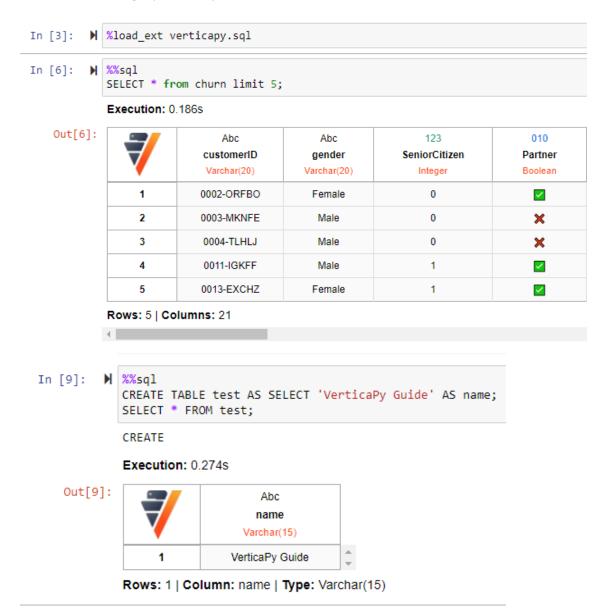
Import the python library into your environment



Connect to your Vertica Database. You will need the host, port, username, password, and database name to connect. An example of how to make a database connection:

For more please visit: https://www.vertica.com/python/workshop/introduction/installation/index

Now we can push queries directly to the database. One way to do this is to use the built in SQL Extension: <u>SQL Magic | VerticaPy</u>



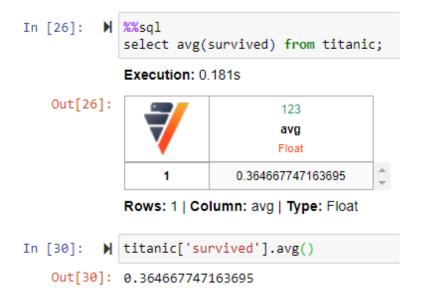
VerticaPy has many built in datasets to play around with, one of which being the titanic dataset, containing information on the passengers aboard the ship.

t	<pre>from verticapy.datasets import load_titanic titanic = load_titanic()</pre>							
n [28]: ▶ t	titanic							
Out[28]:	7	123 pclass Int	123 survived Int	Abc name Varchar(164)	Abc sex Varchar(20)	123 age Numeric(6,3)		
	1	1	0	Allison, Miss. Helen Loraine	female	2.0		
	2	1	0	Allison, Mr. Hudson Joshua Creighton	male	30.0		
	3	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25.0		
	4	1	0	Andrews, Mr. Thomas Jr	male	39.0		
	5	1	0	Artagaveytia, Mr. Ramon	male	71.0		
	6	1	0	Astor, Col. John Jacob	male	47.0		
	7	1	0	Baumann, Mr. John D	male	[null]		
	8	1	0	Baxter, Mr. Quigg Edmond	male	24.0		
	9	1	0	Beattie, Mr. Thomson	male	36.0		
	10	1	0	Birnbaum, Mr. Jakob	male	25.0		

Titanic is now a virtual dataframe (learn more: <u>vDataFrame | VerticaPy</u>)

To calculate the average of a column, one can use both SQL or pandas-like functions in Python. In both cases, VerticaPy pushes the query and computation to the database, and displays the results.

In the first example, the SQL is pushed to the Database. In the second example, the python is converted to SQL first before being pushed to the database for execution. In both cases, the query ran by the database is the same, and the result is the same!



A basic data exploration exercise may be to calculate the correlations between different variables. VerticaPy can plot a heatmap using the corr() function

In [20]: Ŋ titanic.corr()

Out[2**0]:**

7	"pclass"	"survived"	
"pclass"	1	-0.335856950271864	-0.400
"survived"	-0.335856950271864	1	-0.0422
"age"	-0.400828642351015	-0.0422446185581737	
"sibsp"	0.06	-0.018	
"parch"	0.009	0.087	
"fare"	-0.561687581153705	0.264150360783869	0.1785
"body"	-0.0472355333131433	nan	0.0581

Rows: 1-7 | Columns: 8

