**ERROR 1:**

ERROR: Could not install packages due to an EnvironmentError: [Errno 2] No such file or directory:

Common Error on Windows.

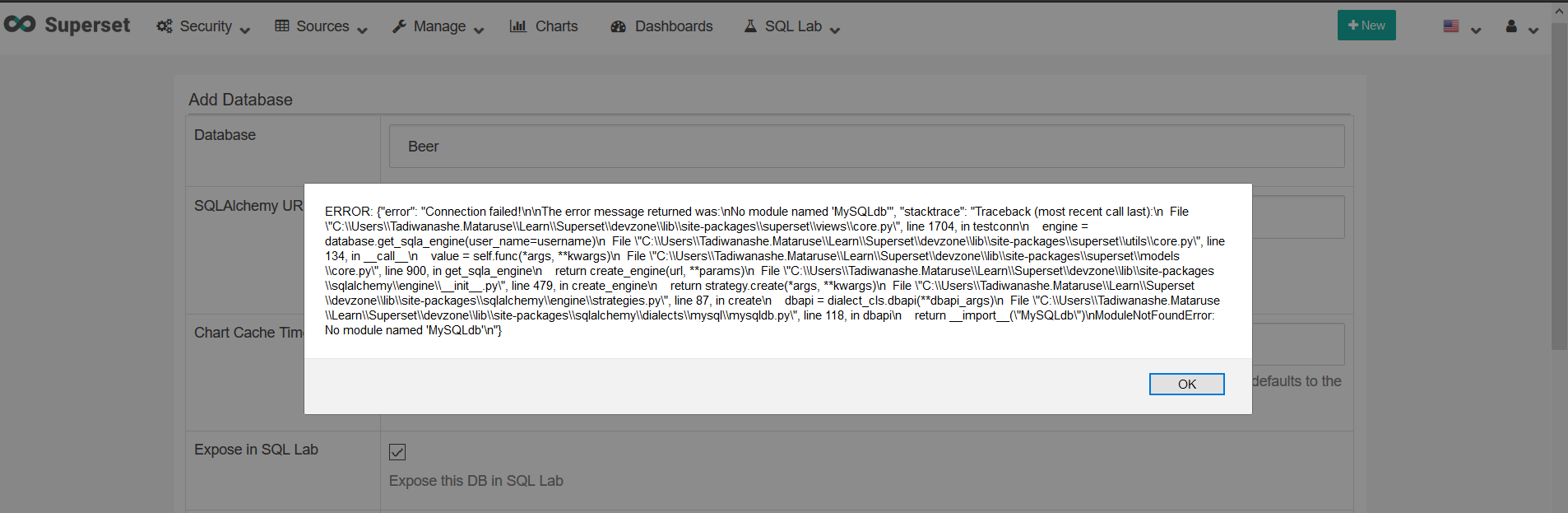
**Solution:** Make sure you Disable python Max path length limit.

**ERROR 3:**

Error: Could not locate a Flask application. You did not provide the "FLASK\_APP" environment variable, and a "wsgi.py" or "app.py" module was not found in the current directory.

**Solution**: On Windows use **SET** instead of **Export**

**Error:** Failing to connect to MySQL Database.



**Solution:** Do a **pip list**

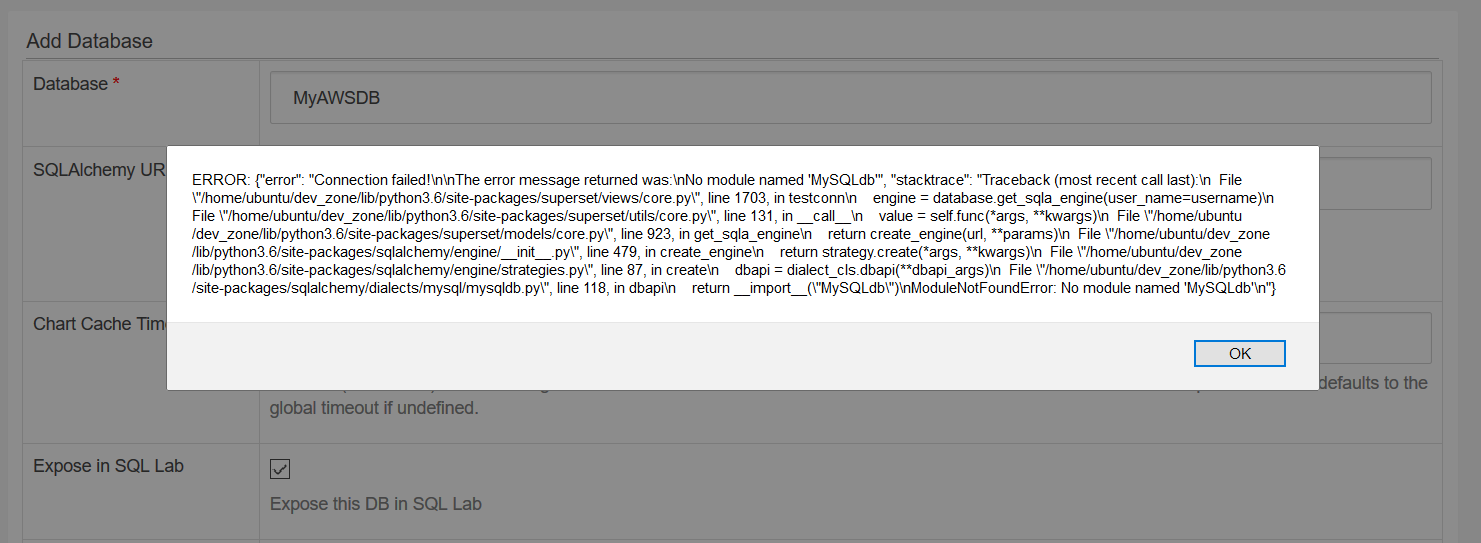
Check to see if MySQL driver is available or not. Most probably the driver will be missing, so run the following command: **pip install mysqlclient**

**ERROR: module 'signal' has no attribute 'SIGALRM'**

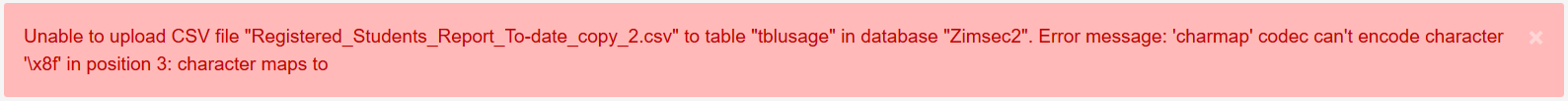
**Solution:** This is because Windows does not have a signal alarm.

**ERROR: 'charmap' codec can't encode character '\u0107' in position 8: character maps to <undefined>**

**Solution:**

**Error: Insert**

**Error No. 2003. Can't connect to mysql server**

**ERROR: CSV UPLOAD**

**Solution:**

Usually this error comes because you will have either installed Superset with root privileges, “sudo” OR

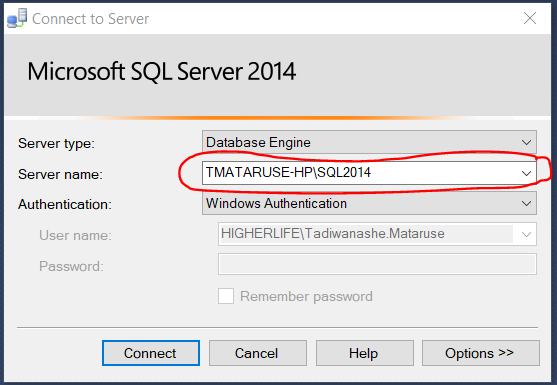
Generally your CSV file will be too big for Superset to upload before timeout.

**MICROSOFT SQL SERVER CONNECTION**

Check the Github Repository README file for the initial processes. This section of the document assumes that you have gone through all the other stages and now only want to fix the SQL Server Configurations.

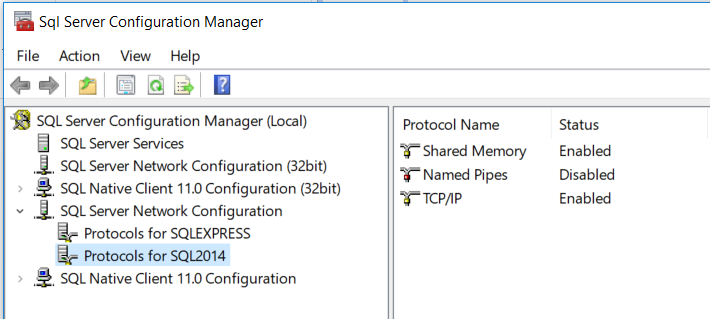
**Step 1: Launch SQL Server Management Studio – In my case I am using SQL Server 2014**

**Step 2: Take note of the actual name of your SQL Server instance:**

****

Make sure you record the SQL Instance name, in my case it is **SQL2014.**

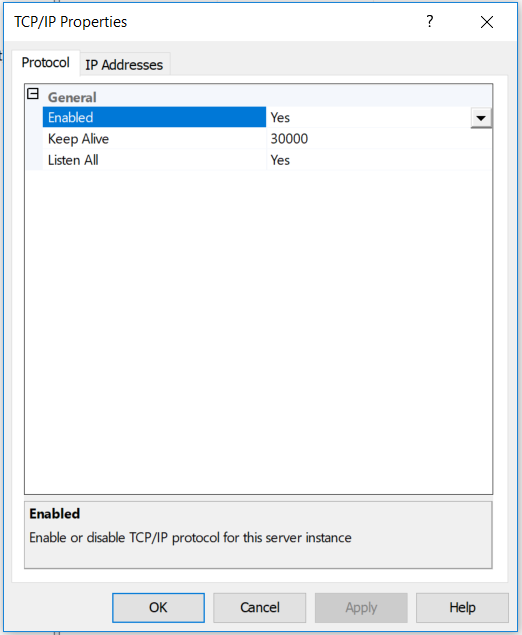
**Step 3: Launch SQL Server Configuration Manager:**



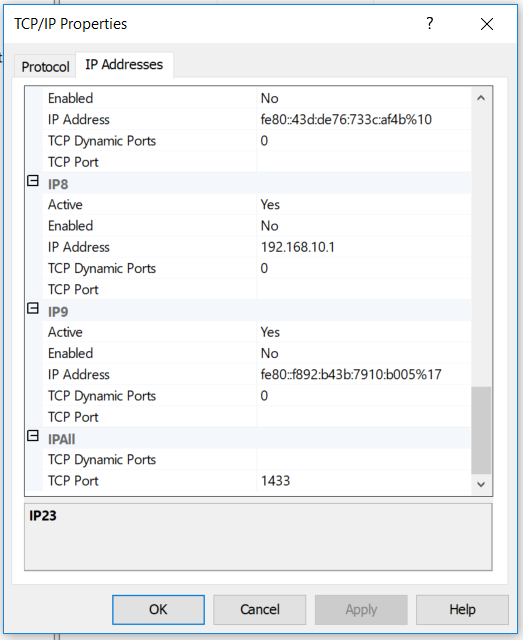
Look for SQL Server Network Configuration, then collapse it and look for Protocols that match your named SQL Server Instance, mine in this case is **Protocols for SQL2014**.

After that, Double Click the matching TCP/IP Protocol name and check the status and make sure that it is **enabled**. Either way, Double Click on the **Protocol Name TCP/IP** as shown in the following screen.

**Step 4: Protocol Name Properties**



Make sure that you change **Enabled to Yes** as on the above screen. After that Click on Apply and read through the pop-up that comes.

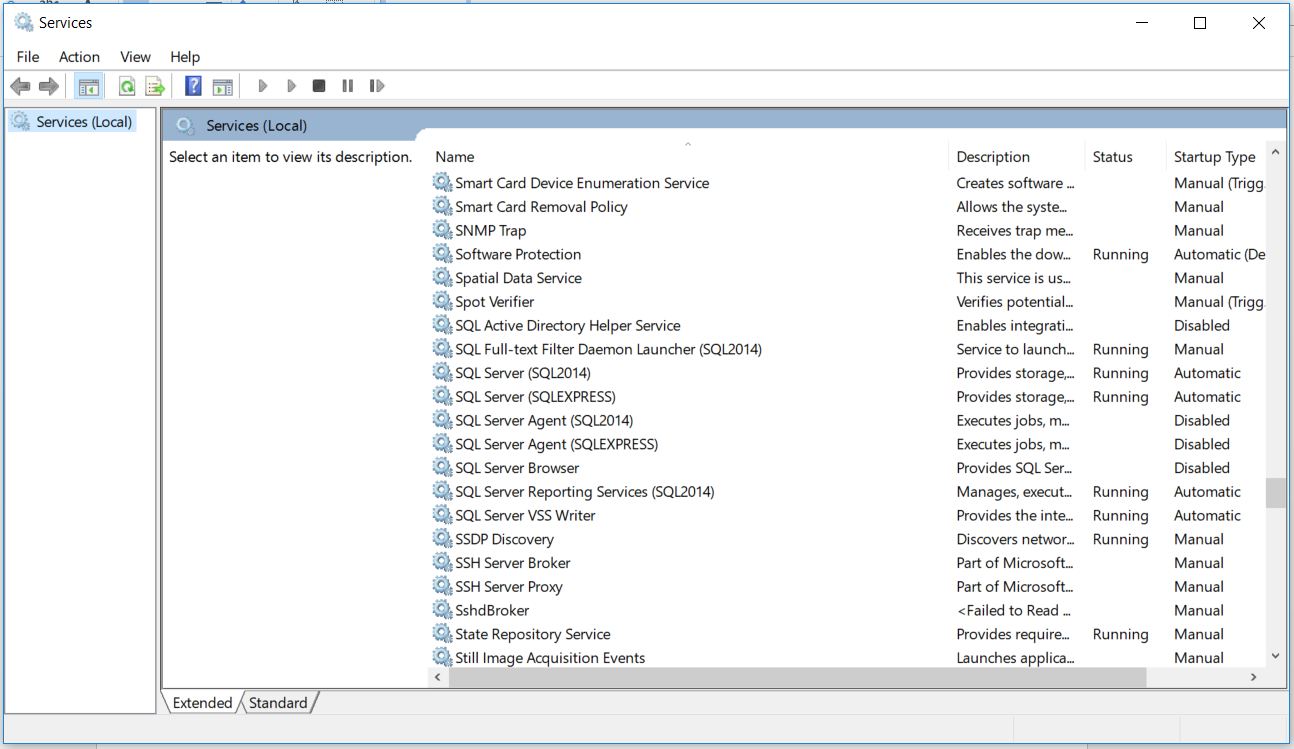


IP Addresses Tab

Make sure you later then select the IP Addresses Tab on top as shown. Scroll all the way down to **IPAll** and **CLEAR THE TCP Dynamic Ports, REMOVE WHATEVER IS WRITTEN IN THERE**.

* Proceed to then enter the port number and the default one is **1433,** in the TCP Port text field. Click on **Apply** and another POP UP dialog box will show.
* All these changes we made in this Step, remember will only take effect after we have restarted our named SQL Server Instance service.
* Look for Services, either click **Start Menu** then Type **Services** or press **WIN\_KEY + R** then type **services.msc**

The screen below shows the Services screen that opens up after.



* Locate SQL Server (Instance name) and in case the service name is **SQL Server (SQL 2014)** and restart the service.

After all this, proceed with what is mentioned on the Github repo.