My code as of now has the following structure:

- dbutils.py (same as professors code)
- signature.py (interface for user to interact with the TwitterAPI)
- twitter_api_mysql (Twitter API with MYSql on the backend)
- driver.py (performance tester).

My code results in 533.425 timelines per second and 1030.169 tweets per second. I think getting the tweets to load faster is easier than getting the timelines to load. I was having issues with loading the timelines when I was using the full dataset rather than the sample. Get timelines was the one I struggled the most with. Despite having indexes I could not output any timelines when using the full tweets.csv. Tweets_sample.csv and follows.csv worked, as well as tweets_sample.csv and follows_sample.csv.

I lost connection trying to drop the tables so I think my computer is just very very slow