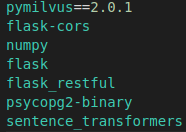
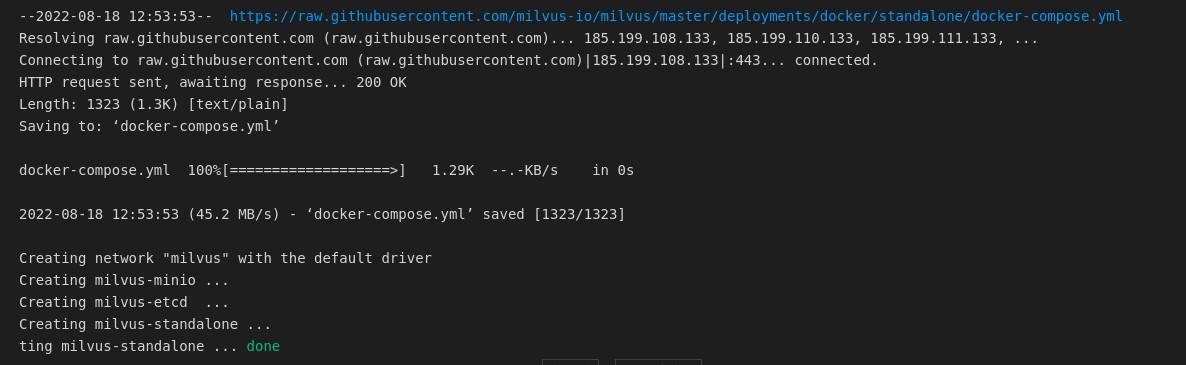
**Step–1 :-** Install the following libraries using pip install.



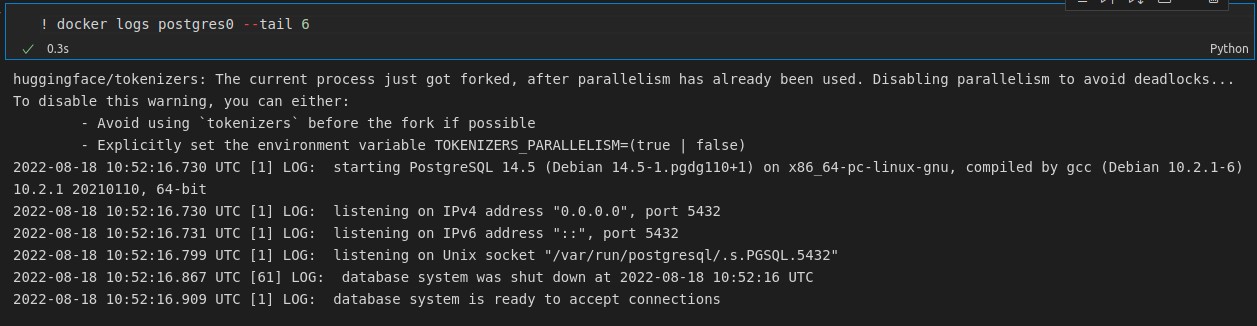
**Step-2 :-** Download the docker-compose(wget https://raw.githubusercontent.com/milvus-io/milvus/master/deployments/docker/ standalone/docker-compose.yml -O docker-compose.yml) and run command sudo docker-compose up -d.



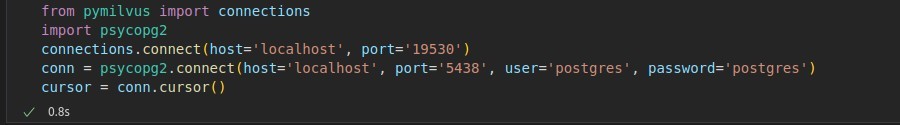
**Step-3 :-** Start the Postgres Server by command(!docker run --name postgres0 -d -p 5438:5432 -e POSTGRES\_HOST\_AUTH\_METHOD=trust postgres)



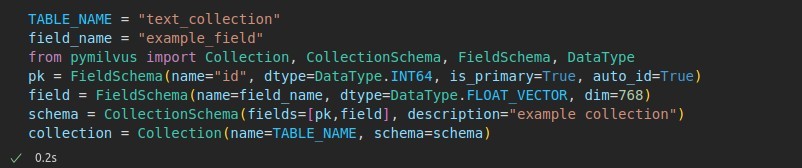
**Step-4 :-** See the docker logs.



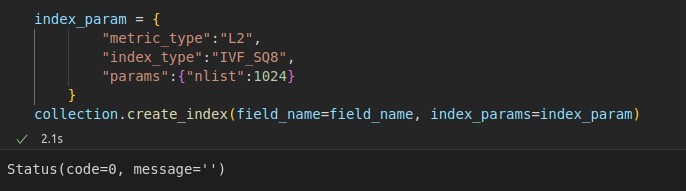
**Step-5 :-** Connect to Docker and postgres server.

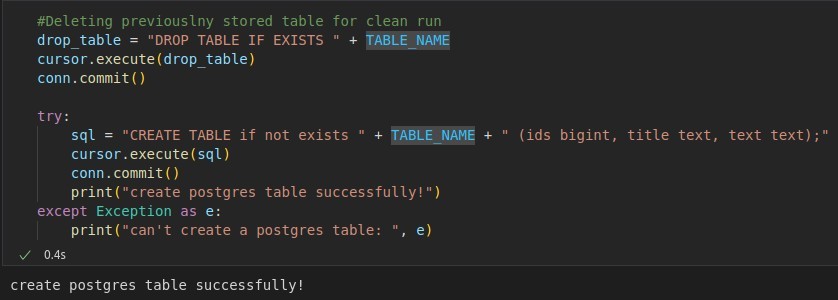


**Step-6 :-** Create the collection by giving the name of the collection and dimension of the vector.

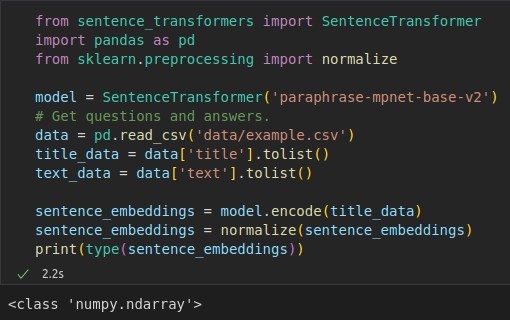


**Step-7 :-** Set the index. This can be done before or after inserting the data. If done before, indexes will be made as data comes in and fills the data segments.

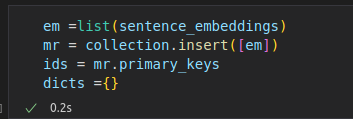


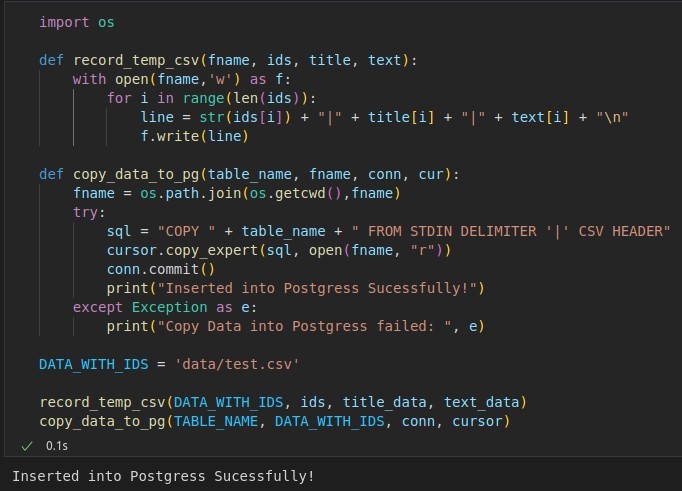
**Step-8 :-** Create the new table in PostgresSQL.**S**

**Step-9 :-** Generate the embeddings, in this we are using the sentence\_transformer library to encode the sentence into vectors. This library uses the modified BERT model to generate the embeddings.

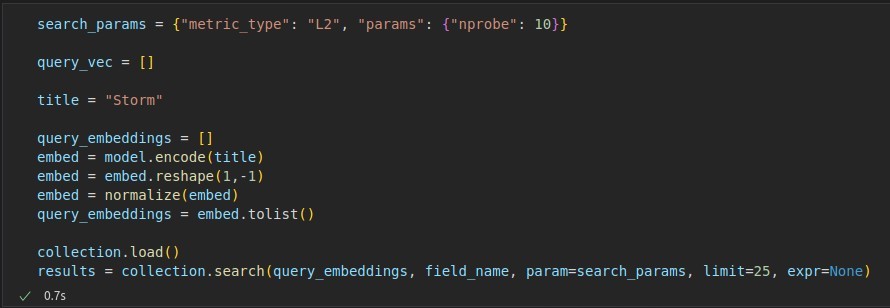


**Step-10 :-** Insert the vectors into Milvus.

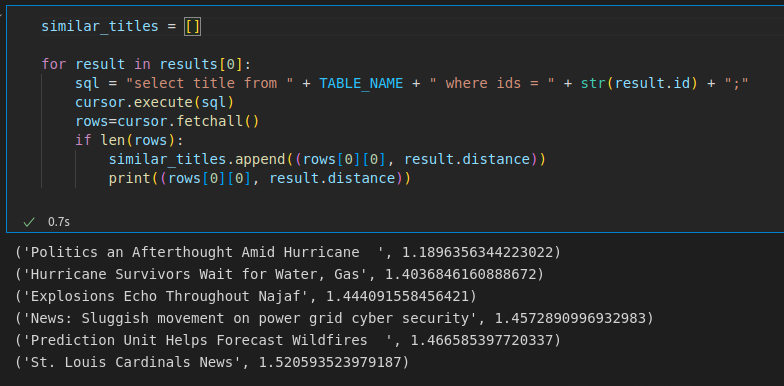


**Step-11 :-** Insert the data(ID, Title and Text) into PostgreSQL.

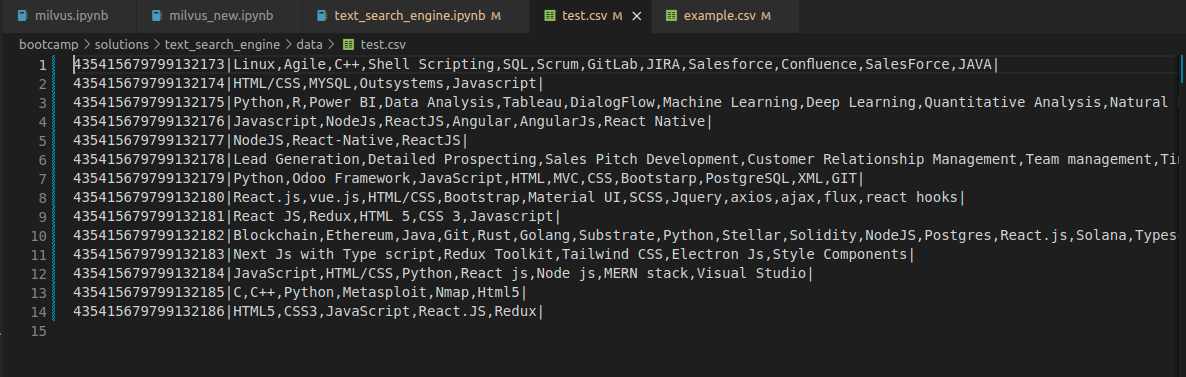
**Step-12 :-** Now, we will generate the embedding we want to search and then will search for the similiar Embeddings in the Milvus.



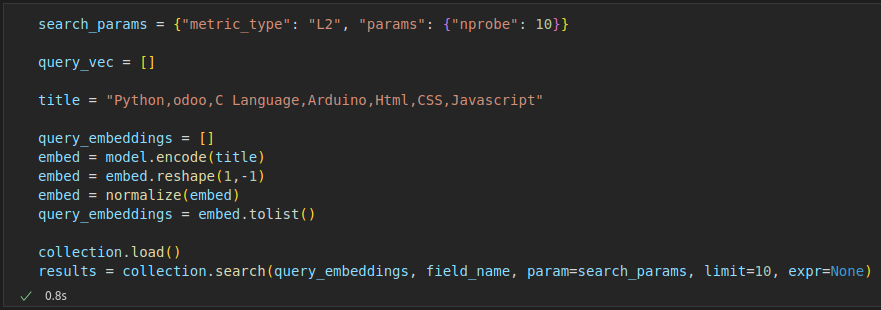
**Step-13 :-** Get the most closest Titles.



**Step-14 :-** We have prepared our Dataset.



**Step-15 :-** Now, get the similar skills. We have searched skills :-Python,odoo,C Language,Arduino,Html,CSS,Javascript



**Step-16 :-** Get the closest titles.

