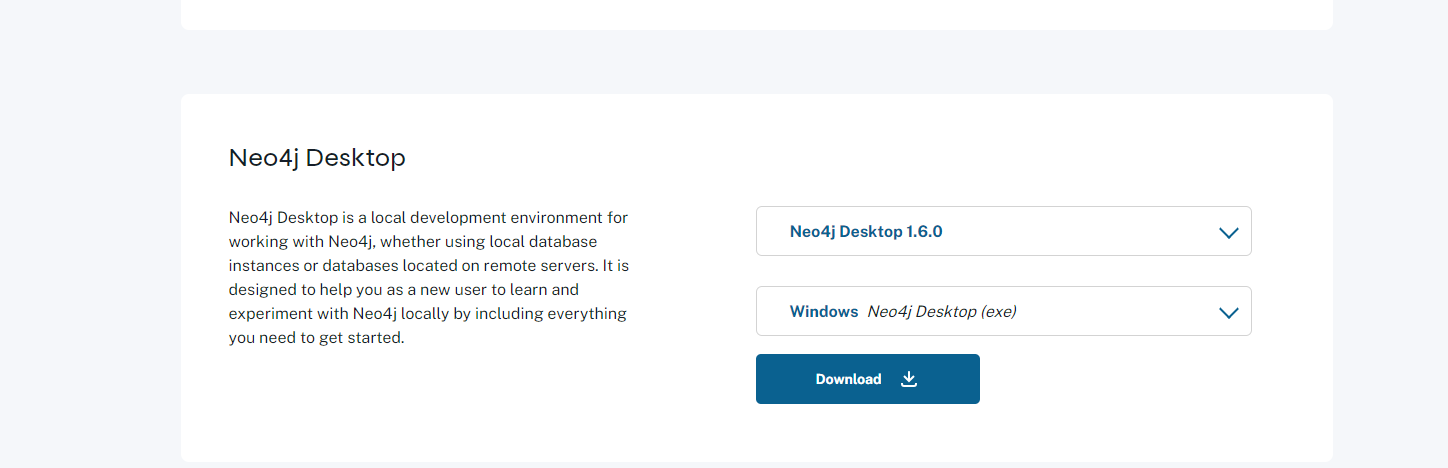
**Course Structure on Neo4j**

**You will be given few questions based on concepts related to unit 3 and unit 4 to query and find the appropriate results from the given graph stored in the neo4j database**

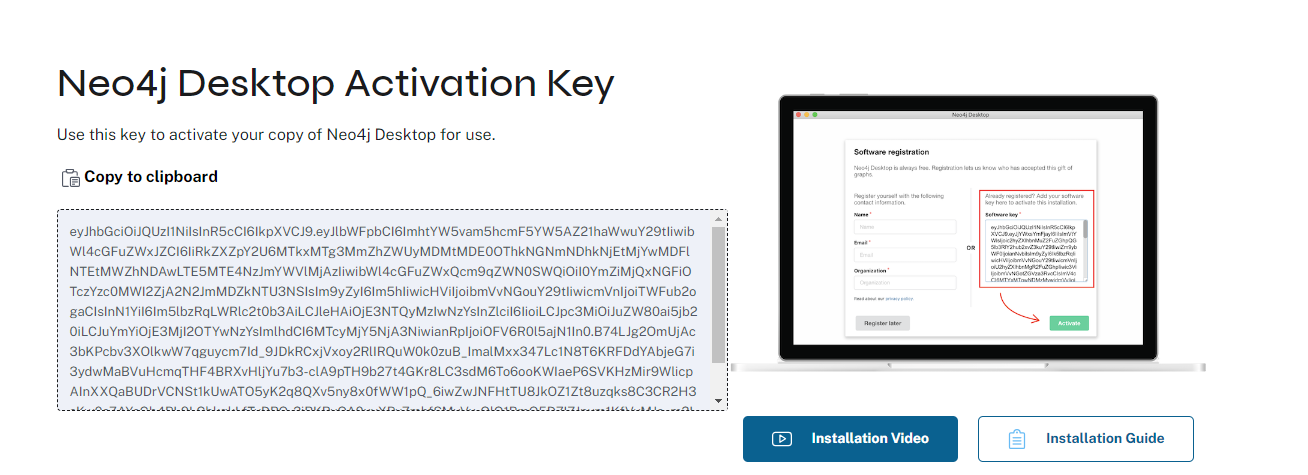
**Instructions for downloading Neo4j**

**Students should first download neo4j desktop by accessing the below link**

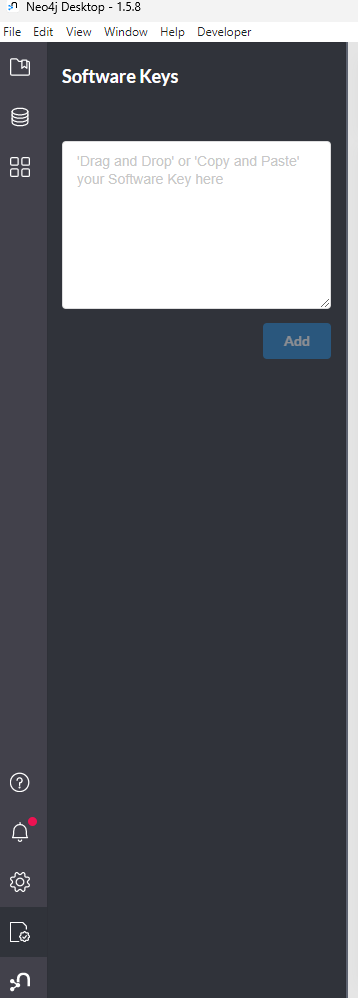
[**https://neo4j.com/deployment-center/?desktop-gdb**](https://neo4j.com/deployment-center/?desktop-gdb)

****

**Copy the activation key**

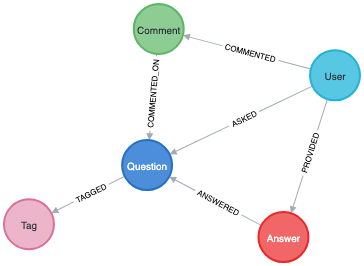
****

**After installing it paste the key and click on add**

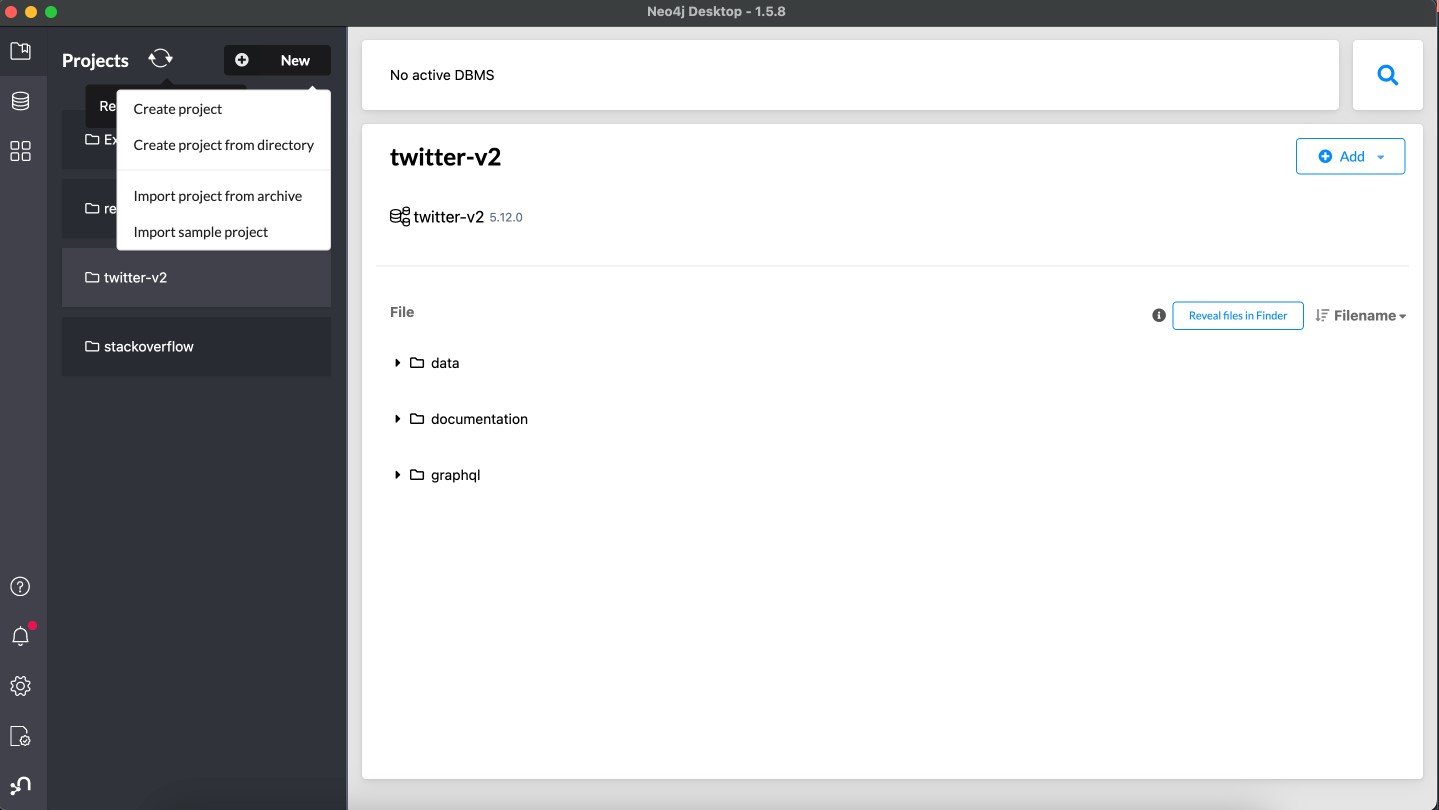
****

**Instructions for creating a sample project**

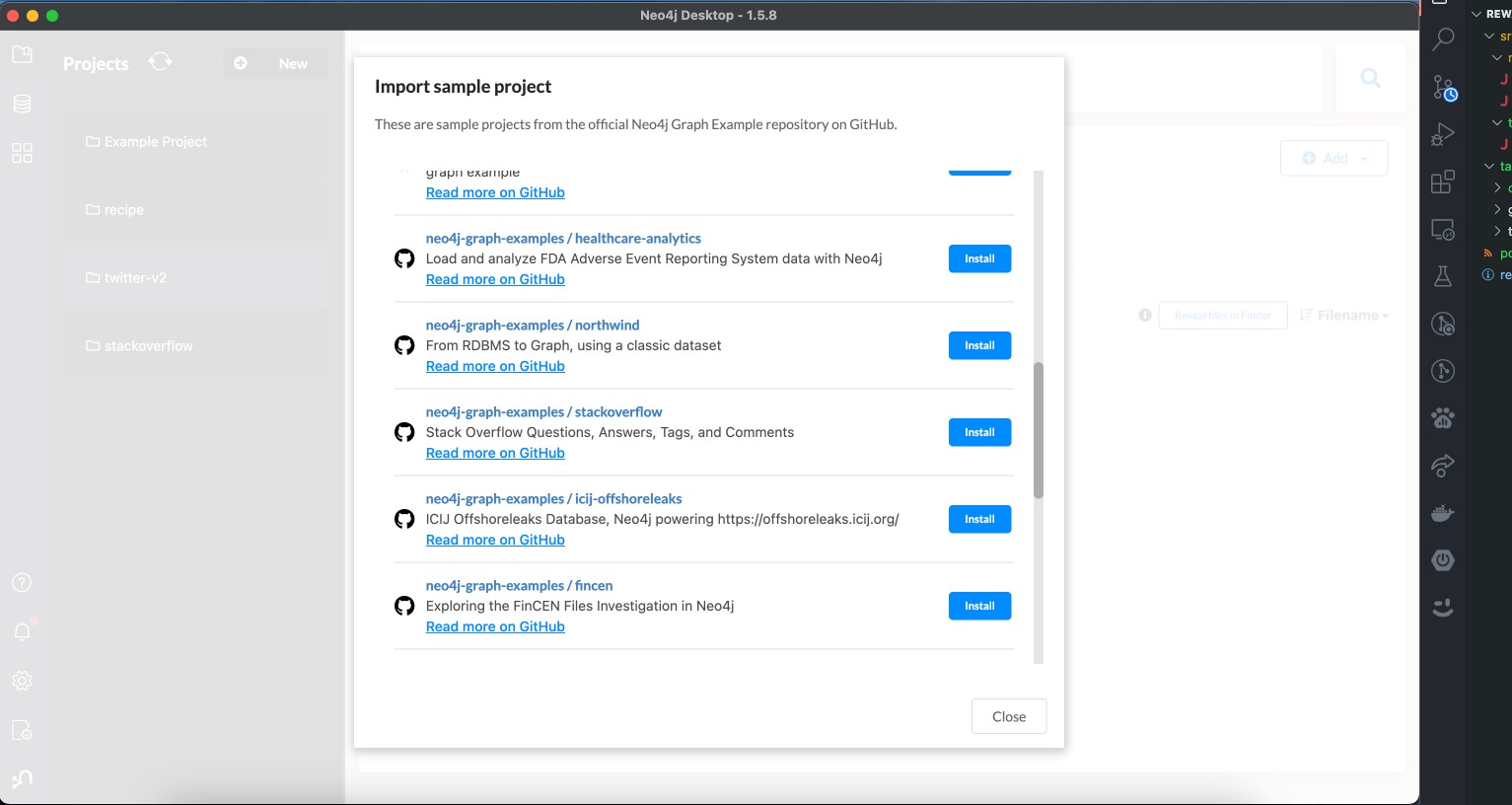
# Stack Overflow Dataset



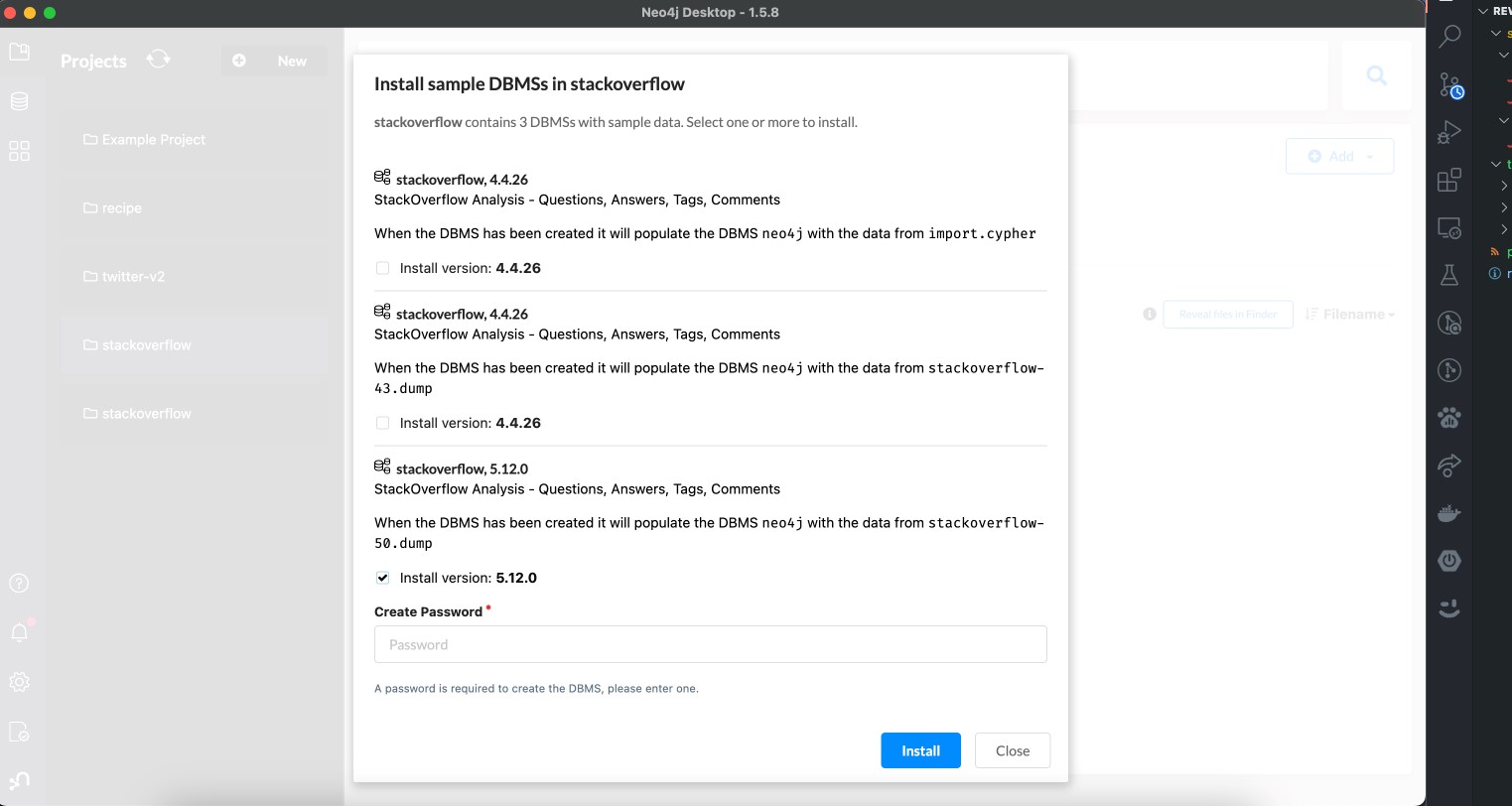
To import a new sample stackoverflow project, launch neo4j desktop, click on ***New***, followed by ***Import sample project.***



Click on ***Install*** next to ***neo4j-graph-examples/stackoverflow.***

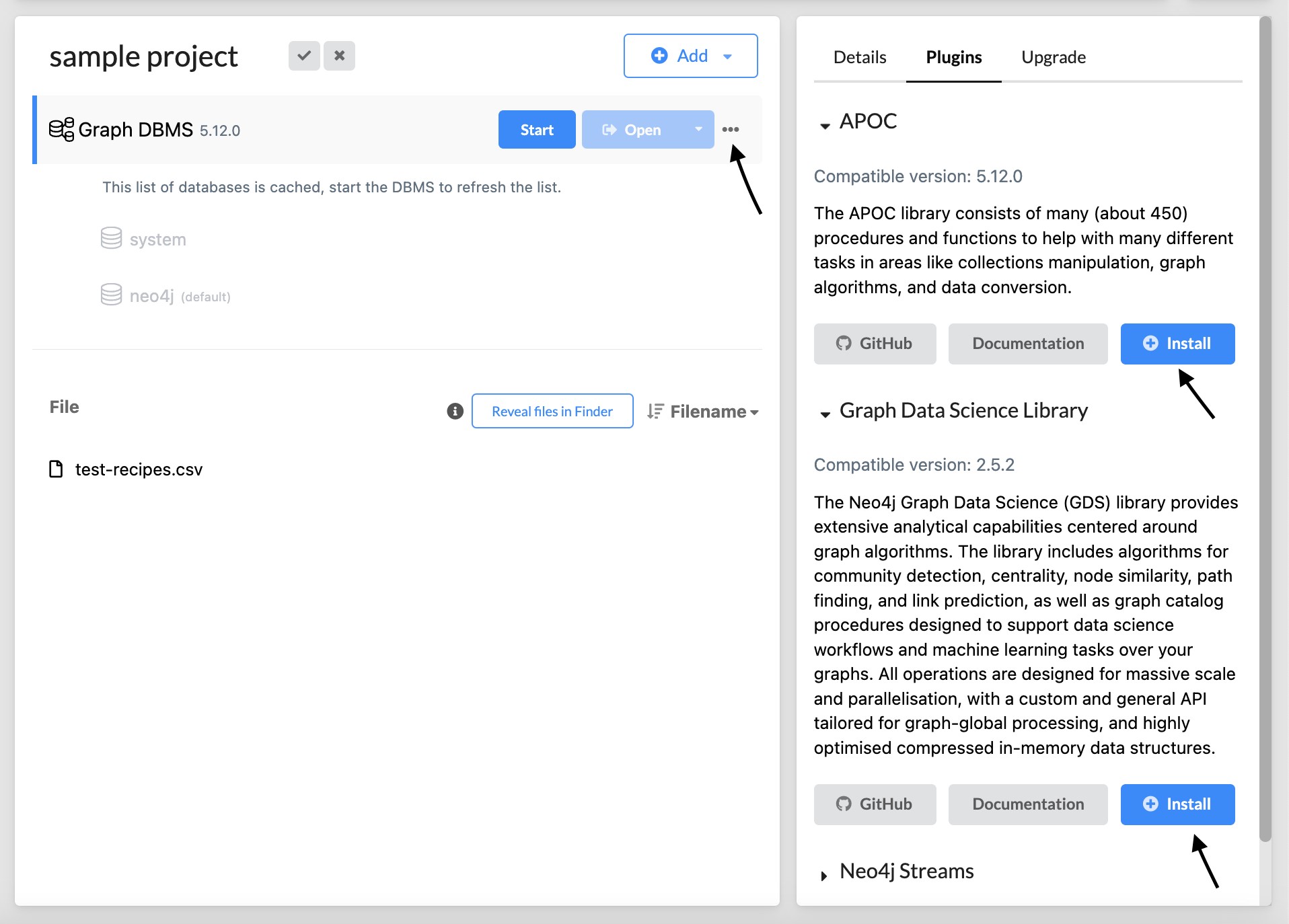


Click on ***Install*** next to ***neo4j-graph-examples/stackoverflow.***



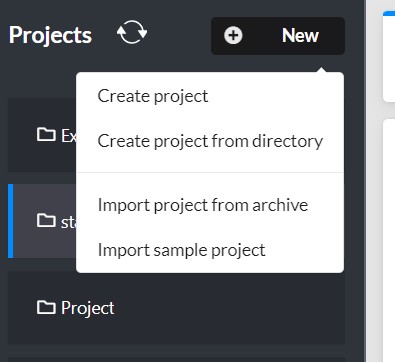
# Plugins

To make use of procedure calls in commands, install APOC and GDS library plugins by opening the three dot menu and clicking ***Install.***

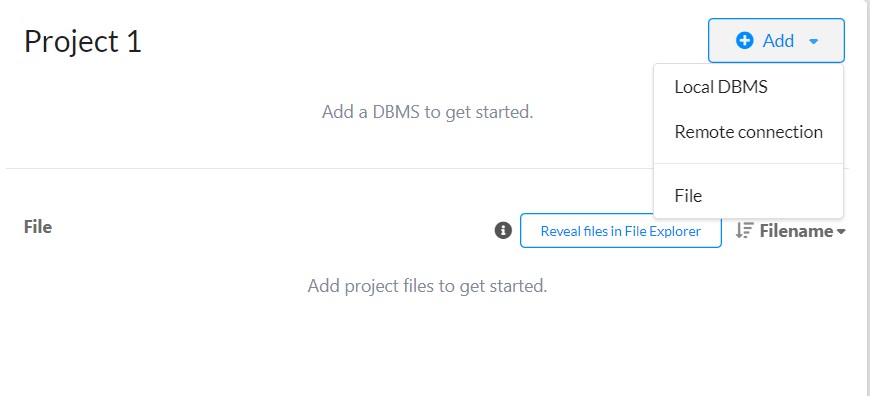


IF YOU ARE UNABLE TO IMPORT THE TWITTER DATASET, DO THE FOLLOWING:

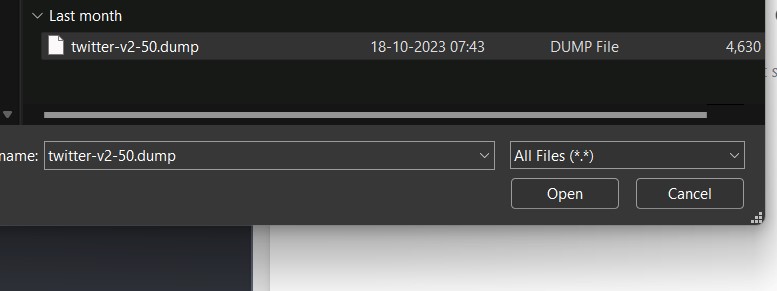
1. **Download the dump file from here :**
2. **Click on new and click on Create Project:**



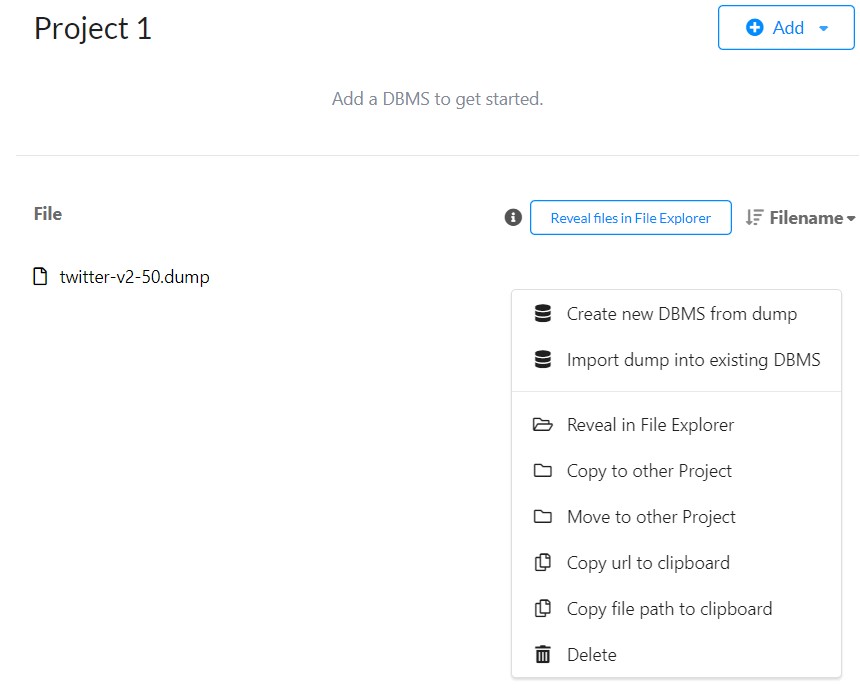
1. **Open the new project, click on Add, and choose File.**



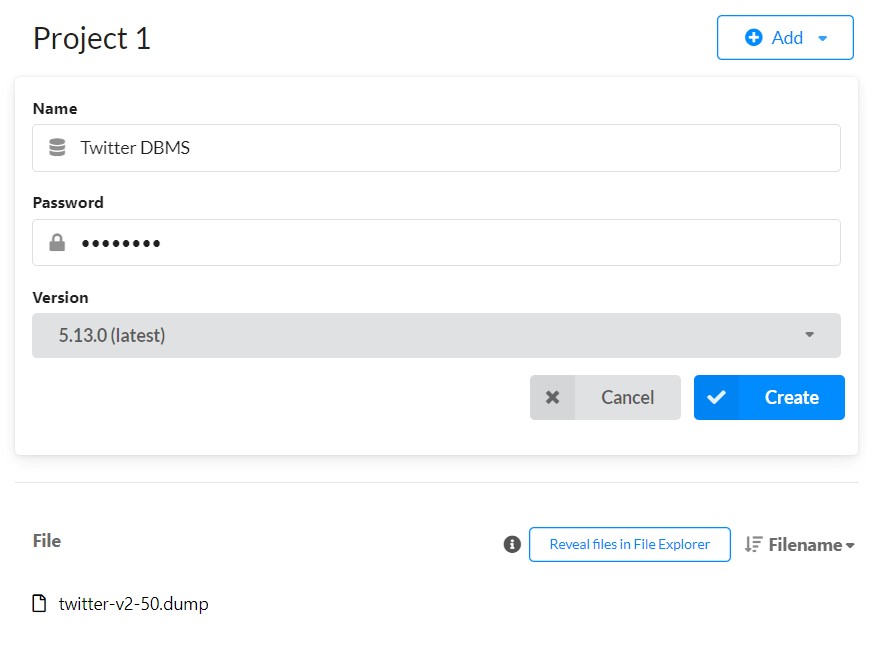
1. **Choose the downloaded dump file and click on Open.**



1. **Hover over the dump file, click on the three dots on the side, and choose Create new DBMS from dump.**



1. **Name your DBMS, add a password, and click on create.**



1. **Once the DBMS is ready, click on start, and open it with Neo4j Browser.**

**SAMPLE QUESTIONS (STACKOVERFLOW)**

Question 1: List the top 5 most popular tags and their count ' with respect to the number of questions that tag them. Return the tagname as “tag\_name” and count of tags as “count” ordered by count in non increasing order

Question 2: Determine which programming language or technology tags are at the core of discussions and widely used based on the number of questions that tag it. Return the tag name as 'Tag' and the question count as 'questionCount'

Question 3: How many users have not asked any question on stack overflow? Return the users count as 'count'

Question 4: How many questions have been viewed more than a hundred times and still been left unanswered? Return the question count as 'questionCount'.

: Question 5: Which question title has the maximum answer count? Return title as 'qtitle'.

Question 6: List the tag that co-occurs with the tag name 'docker', and has the highest frequency(the number of questions it co-occurs with) Return the tag name as 'tag\_name', frequency as 'freq'

Question 7: List the number of answers that aren't accepted for the question with maximum answers. Return the number as 'count'.

Question 8: Consider a user who has commented on and provided the answer to his/her own question. How many such comments exist? Note that a user may have left multiple comments on the question he asked and answered. Return the comment count as 'count'.

Question 9: Find the most influential tag based on page rank score, using 'TAGGED' relationship, or are all tags influential? Explain your answer, and return the score of the most influential tag(s) as 'score'

Question 10: Using Jaccard's method, compute the similarity of all tags based on the question they were TAGGED to and return the URLs of the least similar pair of tags and their score. Return tag 1's url as 't1' and tag 2's url as 't2' and Jaccard similarity score as 'score'.

**Question 11:**Identify the top 5 users with the highest eigenvector centrality score in the dataset. Return the user IDs and their scores.

**Question 12:** Determine the user with the highest closeness centrality in the dataset. Return the user ID and the centrality score.

**Question 13** Question 8: Run a Louvain community detection algorithm to identify communities within the dataset, using all nodes and all relationships, and return the count of total detected communities as 'communityCount'

**Question 14** In the Stack Overflow user interaction network, identify all the maximal sets of users where each user can reach every other user in the same set via directed interactions. Return the strongly connected component ID and the list of user IDs in each component.

**Question 15** Identify all the maximal sets of users on Stack Overflow where each user is connected to every other user in the same set if we ignore the direction of interactions (i.e., considering interactions as undirected). Return the weakly connected component ID and the list of user IDs in each component.