# Stack Overflow Set 175

Question 1: Return the degree distributions of the top 20 most tagged topics. Return tag name as 'tag' and degree of that particular tag as 'degree' ordered by degree in non increasing order.

Enter answer query as text:

Screenshot of query output:

Question 2: Find the degree distributions of the tags co-occurring with tag name 'ruby' (limit to 10), in descending order of frequency. Return tag name as 'tag\_name' , their frequency as 'freq' and their degree as 'degree' .

Enter answer query as text:

Screenshot of query output:

Question 3: Which question has the maximum views?Return the question title as 'question\_title' , number of views on that particular question as 'view\_count' .

Enter answer query as text:

Screenshot of query output:

Question 4: 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

Enter answer query as text:

Screenshot of query output:

Question 5: What are the least similar questions based on its TAGGED relationship? Return question titles as 'q1' and 'q2' and its node similarity score as 'similarity'.

Enter answer query as text:

Screenshot of query output:

Question 6: List the number of answers that aren't accepted for the question with maximum answers. Return the number as 'count'.

Enter answer query as text:

Screenshot of query output:

Question 7: 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'.

Enter answer query as text:

Screenshot of query output:

Question 8: List the tag that co-occurs with the tag name 'java', and has the highest frequency(the number of questions it co-occurs with) Return the tag name as 'tag\_name', frequency as 'freq'

Enter answer query as text:

Screenshot of query output:

Question 9: 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'

Enter answer query as text:

Screenshot of query output:

Question 10: 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'.

Enter answer query as text:

Screenshot of query output: