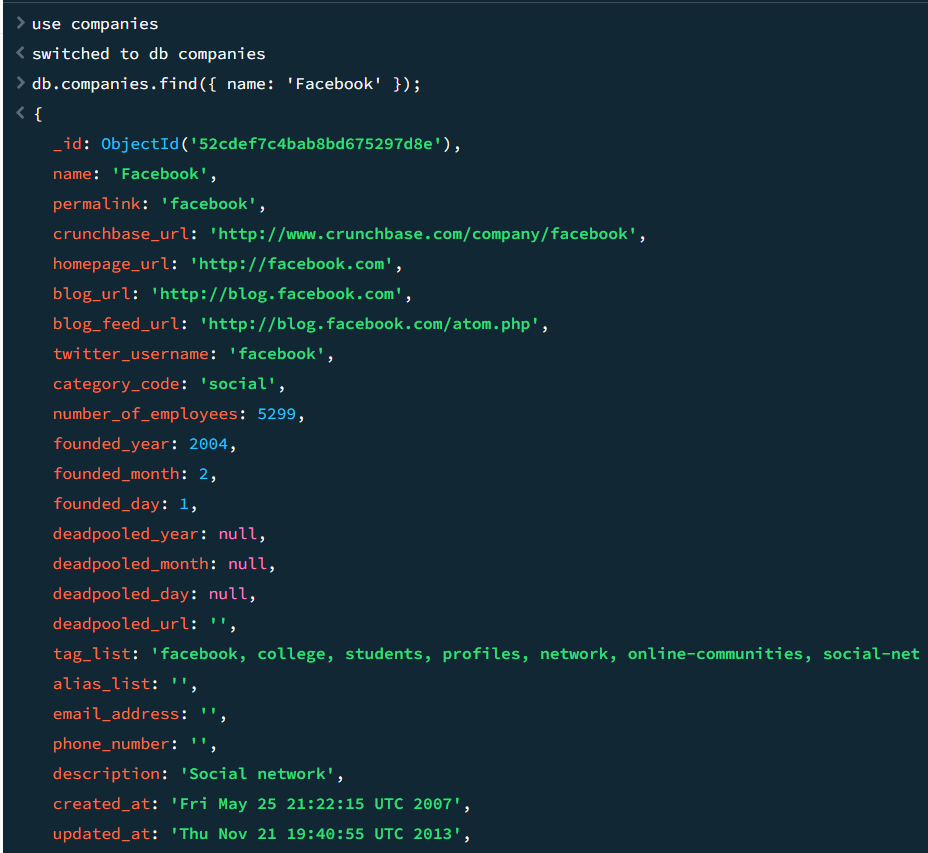
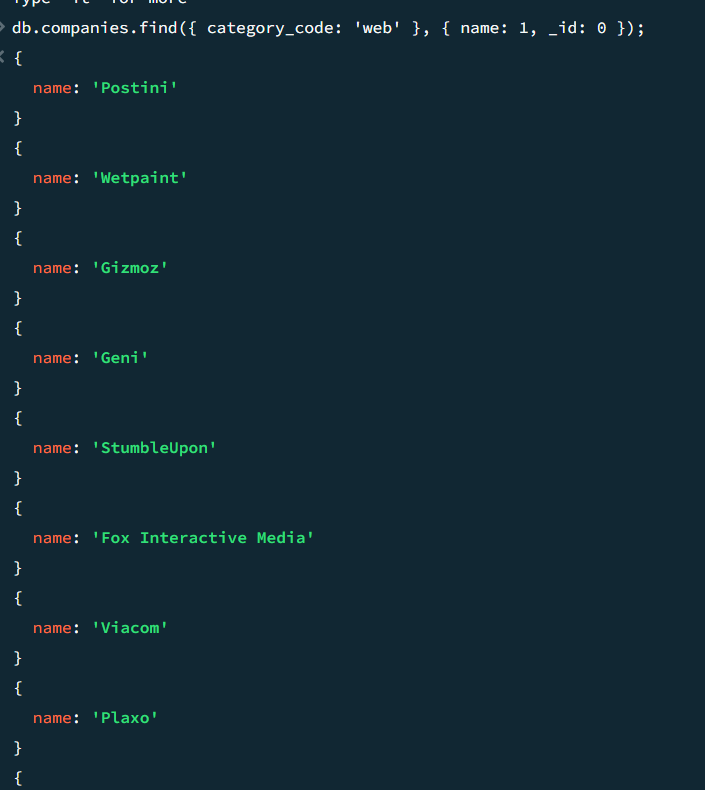


Task 1



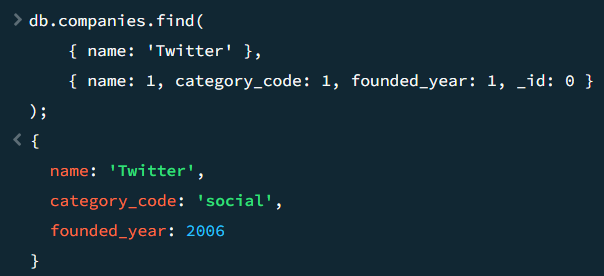
This query searches the companies collection for documents where the field name is equal to 'Facebook'.

Task 2



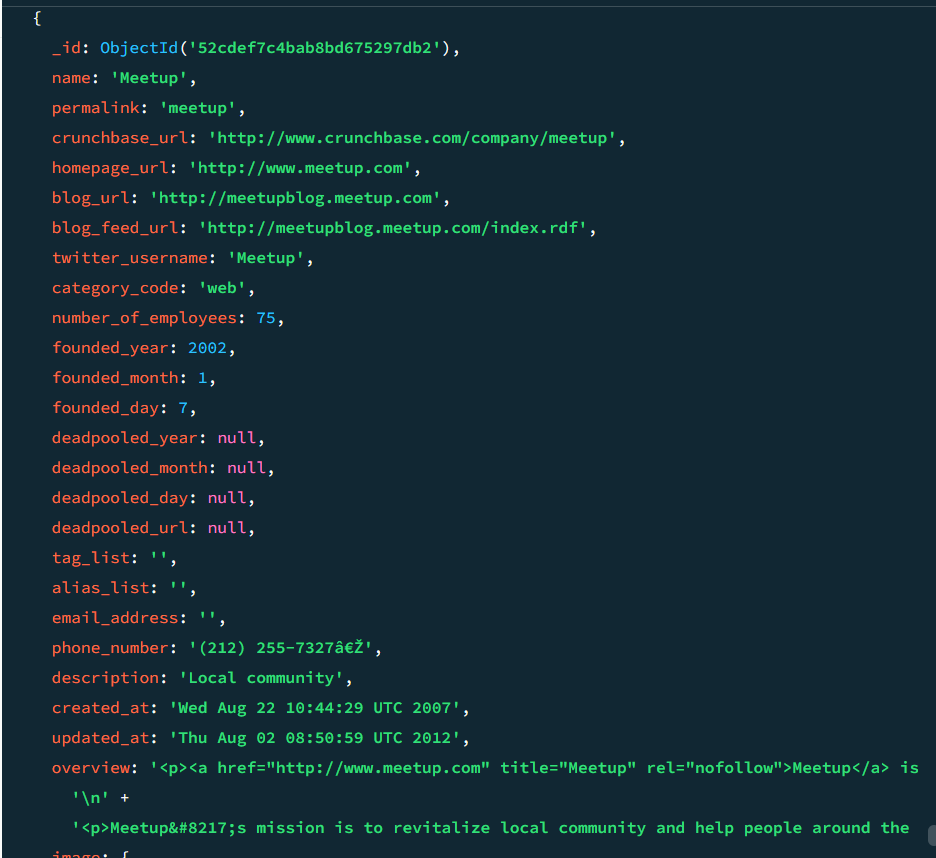
This query retrieves documents where the category\_code field is 'web'. The second argument { name: 1, \_id: 0 } is a projection, which ensures that only the name field is displayed while excluding the \_id field.

Task 3



This query searches for documents where the name field is 'Twitter'. It returns only the fields name, category\_code, and founded\_year, while excluding \_id.

Task 4

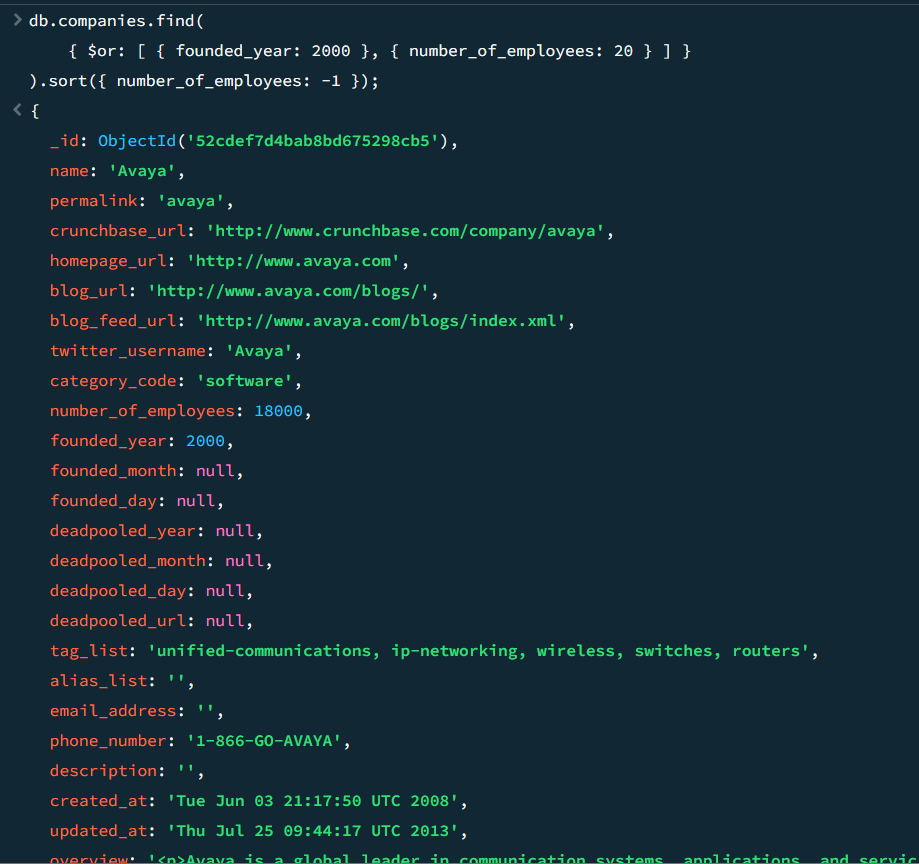


This query retrieves documents where the category\_code is 'web', but limits the output to the first 50 results using the .limit(50) function.

Task 5



This query retrieves documents where category\_code is 'enterprise' and founded\_year is 2005. The projection ensures only the fields name, category\_code, and founded\_year are displayed.

Task 6  


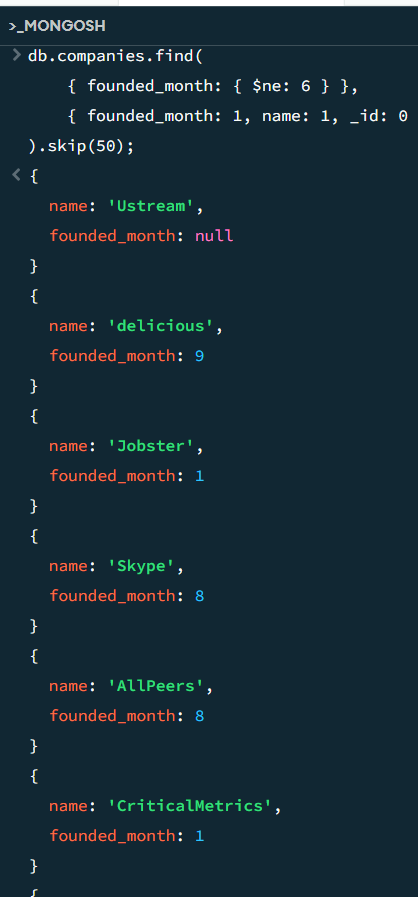
This query uses the $or operator to find documents where either founded\_year is 2000 or number\_of\_employees is 20. The .sort({ number\_of\_employees: -1 }) function orders the results in descending order of number\_of\_employees.

Task 7

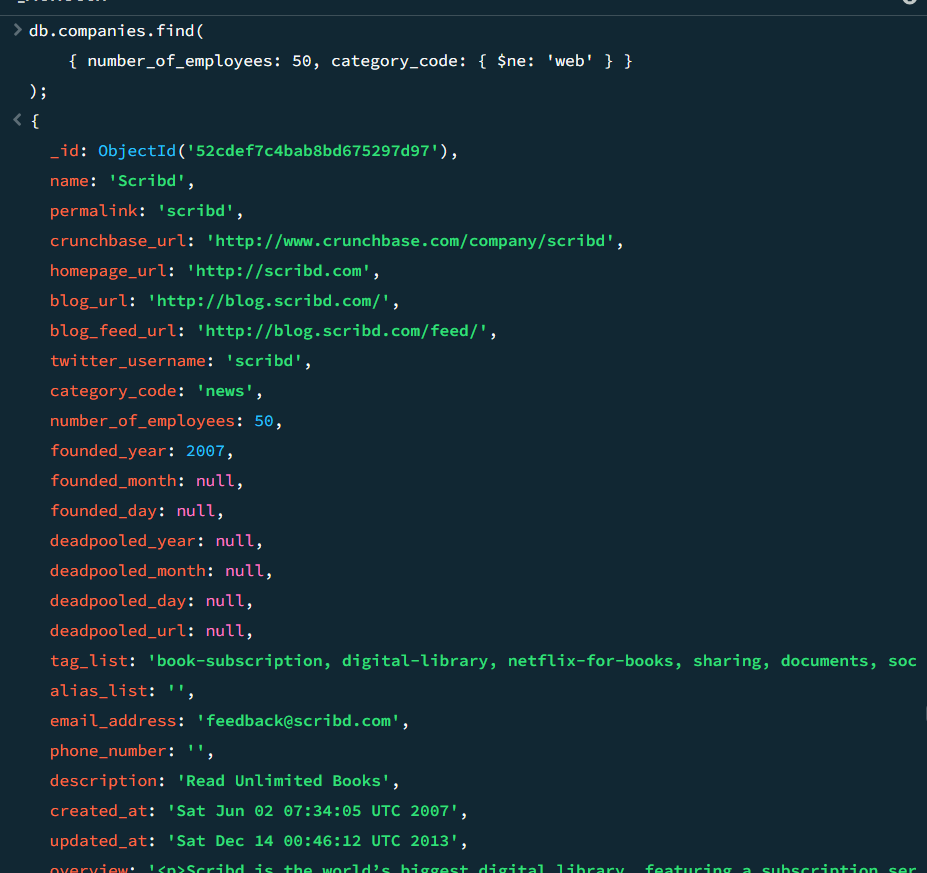


This query uses the $nin operator to exclude documents where category\_code is either 'web' or 'social'. It limits the results to 20 documents and displays only the name and category\_code fields.

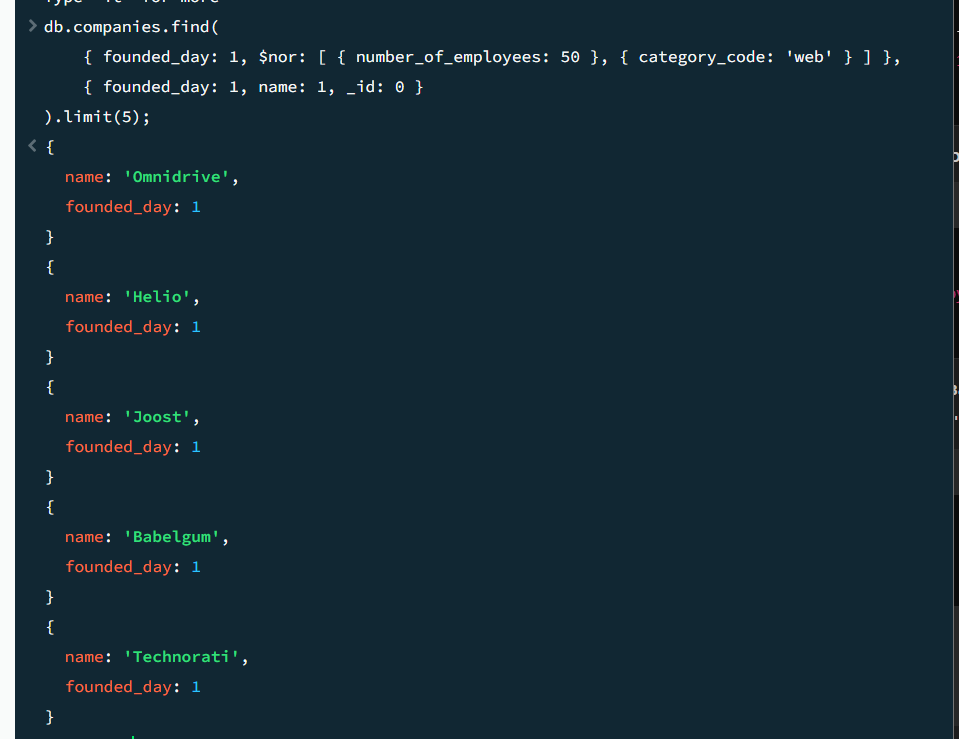
Task 8



This query uses the $ne operator to exclude documents where founded\_month is 6 (June). It skips the first 50 documents using .skip(50) and retrieves only the founded\_month and name fields.

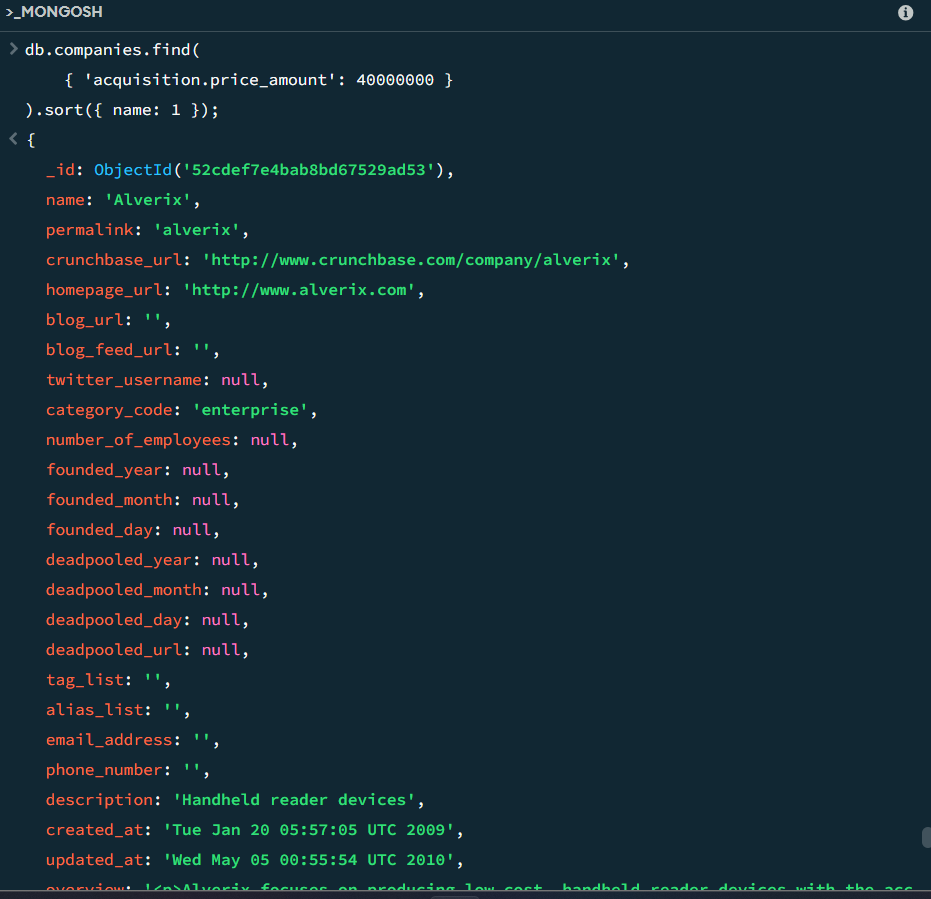
Task 9  


This query retrieves documents where number\_of\_employees is 50 but excludes those where category\_code is 'web' using the $ne operator.

Task 10  


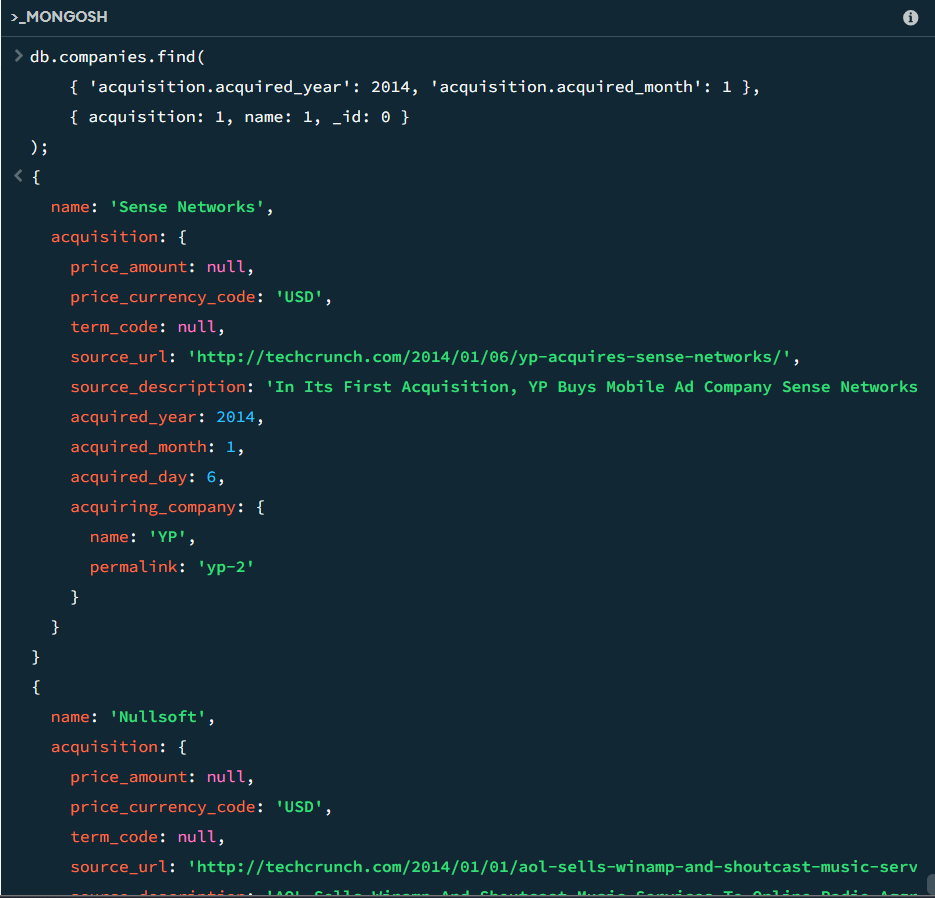
This query uses $nor to exclude documents where number\_of\_employees is 50 or category\_code is 'web'. It limits the results to 5 and retrieves only the founded\_day and name fields.

Task 11



This query searches for documents where the nested field price\_amount in the acquisition object is 40,000,000. It sorts the results alphabetically by the name field using .sort({ name: 1 }).

Task 12



This query searches for documents where the nested field price\_amount in the acquisition object is 40,000,000. It sorts the results alphabetically by the name field using .sort({ name: 1 }).