

1 List all the ToDoLists belonging to a given user. The identifier of the user should be specified as a condition in the WHERE clause of the query.

SELECT Id,Name FROM ToDoList  
WHERE Owner = 1;

2 List all the ToDoItems belonging to a given ToDoList. The identifier of the ToDoList should be specified as a condition in the WHERE clause of the query.

SELECT Id,Title FROM ToDoItem  
WHERE ToDoList\_Id=1;

3 As in (2), but now allow for pagination of ToDoItems. This means that the query should show a pre-defined amount of results, starting from a given tuple. Use the LIMIT clause implemented in MySQL (<http://dev.mysql.com/doc/refman/5.7/en/select.html>)

SELECT Id,Title FROM ToDoItem  
WHERE ToDoList\_Id=1  
LIMIT 10,10;

4 Add to the query in (3) the ability to filter ToDoItems according to 1) A range of date creation, 2) a priority level, 3) the completion status.

SELECT Id,Title FROM ToDoItem  
WHERE ToDoList\_Id=1

5 For a given ToDoItem, show all its sub-items.

6 For a given ToDoItem, show the value of all its tags. 3

7 For a given tag, show all the ToDoLists that contain ToDoItems which are tagged with that tag.

8 For each tag, calculate the number of currently pending and completed to-dos.

9 For each week in the current year, calculate the number of completed to-dos.

10 For each tag, retrieve the 10 to-dos with quickest completion time (i.e. the time between the creation and the completion of the to-do).

11 Calculate the frequency of co-occurrence of tags (i.e. the number of times each possible combination of tag pairs is used in the database)

12 For a given ToDoList, calculated the average time of completion of to-dos

13 List the to-dos having a completion time higher than the average time of completion for the todos belonging to the same ToDoList