

## Database Writing Activity

### Learning objectives

- Describe how a database performance can be monitored and improved;
- Discuss and apply approaches for improving a database's performance, including denormalization, use of indexes, and data partitioning.

### Reading questions

1. Discuss three different kinds of knowledge that are needed for an expert in physical database design. (3 points)
2. What are the key inputs into physical database design? In other words, what needs to already have been accomplished in the design process prior to entering a stage of physical database design? (4 points)
3. Discuss the implementation trade-offs for how to handle derived attributes. Begin with a brief description of what a "derived attribute" is. (2 points)
4. Discuss two basic approaches for implementing general constraints. Begin with a brief description of "general constraint." (2 points)
5. Describe the difference between a "file organization" and an "index" (Hint: See Appendix F). (4 points)
6. What is a clustered table? How can it improve performance? (2 points)
7. Why does one need to analyze transactions in order to proceed with physical database design? What is the basic process for analyzing transactions? For example, what facts and assumptions must be uncovered and how is this information used? (4 points)
8. In your own words, describe why the use of a hash index for the attribute Y would be a poor choice (2 points):  
`SELECT ... WHERE Y BETWEEN 10 and 500 // Assume Y is an integer`
9. Show the SQL for adding an index to a table. (2 points)
10. Concerning physical storage, what is a "page"? How does a page relate to a relation, to a tuple, and to an attribute in a tuple? (3 points)
11. Precisely define **functional dependency**, **1NF**, **2NF**, and **3NF**. (4 points)
12. What is denormalization? When is it appropriate to consider denormalization? In general, what are the downsides and upsides? (2 points)
13. Under what circumstances can it make sense to duplicate data in one table (which would normally not be duplicated)? (2 points)
14. Briefly describe the concept of "partitioning," explaining how "horizontal" partitioning is different from "vertical" partitioning. (2 points)
15. In your own words, describe what kind of index is needed so that a **best-match** query can be efficiently computed against a large body of text. A best-match to returns a

Spring 2013

**W03** (44 points)

Due: April 24 @ noon

ranking of possible documents. How does PostgreSQL handle such queries What exactly is the general challenge with such queries? (Hint: See section in PostgreSQL documentation on search types). (2 points)

16. Assume that you have 10,000 records in a database, each with a field of text called message. Message consists of 500 to 1,000 words. Suppose that you would like to perform a best-match search on the field message and find the matching records, in a ranked ordering. Sketch out the SQL for solving this problem showing (1) The index that you would apply and (2) How you would query the table. ( 4 points)