

Assignment 2

Chapter11:

1. A system table space, a user data table space, an index table space, and a temporary table space are examples of _____.
 - a.procedure caches
 - b.file groups
 - c.data caches
 - d.operation modes
2. A(n) _____ is a logical grouping of several data files that store data with similar characteristics.
 - a.procedure cache
 - b.table space
 - c.data cache
 - d.listener
3. Automatic query optimization means that the:
 - a.optimization takes place at compilation time by the programmer.
 - b.DBMS finds the most cost-effective access path without user intervention.
 - c.optimization process is scheduled and selected by the end user or programmer.
 - d.database access strategy is defined when the program is executed.
4. From the performance point of view, _____ databases eliminate disk access bottlenecks.
 - a.RAID
 - b.distributed
 - c.index-organized
 - d.in-memory
5. If there is no index, the DBMS will perform a _____ scan.
 - a.loop
 - b.range
 - c.row ID table access
 - d.full table
6. In standard SQL, the optimizer hint ALL_ROWS is generally used for _____ mode processes.
 - a.interactive
 - b.real-time
 - c.batch
 - d.transaction
7. In the context of RAID levels, _____ refers to writing the same data blocks to separate drives.
 - a.striping
 - b.mirroring
 - c.partitioning
 - d.aggregating

8. Knowing the sparsity of a column helps you decide whether the use of _____ is appropriate.
- a.query processing
 - b.query optimization
 - c.an index
 - d.a full table scan
9. On the client side, the objective is to generate an SQL query that returns a correct answer in the least amount of time, using a minimum amount of resources at the server end. The activities required to achieve this goal are commonly referred to as _____ tuning.
- a.client SQL
 - b.database SQL
 - c.SQL performance
 - d.DBMS performance
10. On the server side, the database environment must be properly configured to respond to clients' requests in the fastest way possible, while making optimum use of existing resources. The activities required to achieve this goal are commonly referred to as _____ tuning.
- a.client and server
 - b.database SQL
 - c.SQL performance
 - d.DBMS performance
11. The DBMS _____ the SQL query and chooses the most efficient access/execution plan.
- a.parses
 - b.executes
 - c.fetches
 - d.processes
12. The DBMS _____ the data and sends the result set back to the client.
- a.parses
 - b.executes
 - c.fetches
 - d.processes
13. The _____ cache is used as a temporary storage area for ORDER BY or GROUP BY operations, as well as for index-creation functions.
- a.data
 - b.SQL
 - c.sort
 - d.optimizer
14. The _____ is a shared, reserved memory area that stores the most recently executed SQL statements or PL/SQL procedures, including triggers and functions.
- a.buffer cache
 - b.procedure cache
 - c.data cache
 - d.permanent storage

15. The _____ must be set large enough to permit as many data requests to be serviced from cache as possible.
- a.data cache
 - b.SQL cache
 - c.sort cache
 - d.optimizer mode
16. The _____ process analyzes SQL queries and finds the most efficient way to access data.
- a.optimizer
 - b.scheduler
 - c.listener
 - d.user
17. The _____ table space is used to store the data dictionary tables.
- a.system
 - b.user data
 - ctemporary
 - d.rollback segment
18. The data cache is where the data read from the database data files are stored _____ the data have been read or _____ the data are written to the database data files.
- a.after; before
 - b.after; after
 - c.before; before
 - d.before; after
19. The data cache or _____ is a shared, reserved memory area that stores the most recently accessed data blocks in RAM.
- a.buffer cache
 - b.procedure cache
 - c.SQL cache
 - d.permanent storage
20. The majority of primary memory resources will be allocated to the _____ cache.
- a.data
 - b.SQL
 - c.sort
 - d.optimizer
21. To work with data, a DBMS must retrieve the data from _____ and place them in _____.
- a.data files; procedure cache
 - b.RAM; data cache
 - c.permanent storage; RAM
 - dtemporary files; procedure cache
22. When moving data from permanent storage to RAM, an I/O disk operation retrieves:
- a.an entire table.
 - b.an entire physical disk block.
 - c.only the row containing the attribute requested.
 - d.only the attribute which was requested.

23. When setting optimizer hints, _____ instructs the optimizer to minimize the overall execution time, that is, to minimize the time it takes to return the total number of rows in the query result set. This hint is generally used for batch mode processes.

- a.ALL_ROWS
- b.FIRST_ROWS
- c.INDEX(P_QOH_NDX)
- d.OPTIMIZATION_ROWS

24. Which of the following is the first step of query processing at the DBMS server end?

- a.Executing
- b.Parsing
- c.Fetching
- d.Delivering

25. _____ is the central activity during the parsing phase in query processing.

- a.Clustering
- b.Partitioning
- c.Query validation
- d.Query optimization

26. _____ refers to the number of different values a column could possibly have.

- a.Database statistics
- b.Data sparsity
- c.A bitmap index
- d.Clustering

Deliverables

MySQL Server

Using **only SQL**: Save as **lis3781_a2_solutions.sql**

- A. Tables and insert statements.
 - B. Include indexes and foreign key SQL statements (see below).
 - C. Include ***your*** query result sets, including grant statements.
 - D. The following tables should be created and populated with at least 5 records **both** locally and to the CCI server.
 - E. **No Credit will be given if tables and data do not forward-engineer to the CCI server.**
-
1. **Using SQL ONLY, NOT** MySQL Workbench:
 2. Locally: create **yourfsuid** database, and two tables: **company** and **customer**
NOTE: Also, these two tables must be populated in **yourfsuid** database on the CCI server.
 - a. Use 1:M relationship: **company** is parent table
 - b. **company** attributes:
 - i. cmp_id (pk)
 - ii. cmp_type enum('C-Corp','S-Corp','Non-Profit-Corp','LLC','Partnership')
 - iii. cmp_street
 - iv. cmp_city
 - v. cmp_state
 - vi. cmp_zip (zf)
 - vii. cmp_phone
 - viii. cmp_ytd_sales
 - ix. cmp_url
 - x. cmp_notes
 - c. **customer** attributes:
 - i. cus_id (pk)
 - ii. cmp_id (fk)
 - iii. **cus_ssn (binary 64)**
 - iv. **cus_salt (binary 64)**
 - v. cus_type enum('Loyal','Discount','Impulse','Need-Based','Wandering')
 - vi. cus_first
 - vii. cus_last
 - viii. cus_street
 - ix. cus_city
 - x. cus_state
 - xi. cus_zip (zf)
 - xii. cus_phone
 - xiii. cus_email
 - xiv. cus_balance
 - xv. cus_tot_sales
 - xvi. cus_notes
 - d. Create suitable indexes **and** foreign keys:
(Review Notes > **Enforcing_PK_FK_Relationship.pdf**)
 - e. **Enforce pk/fk relationship: on update cascade, on delete restrict**

Local Admin: (you do **not** have permissions to create users on the CCI Server)

Create two different users (user3 and user4), with two different passwords: both users can access from localhost only.

See example: Notes>DBA>**Granting_Privileges.pdf**

1. Limit **user3** to select, update, and delete privileges on company and customer tables
2. Limit **user4** to select, and insert privileges on customer table

Log into local server as each user:

3. Verify database/table permissions, show grants:
 - a. you/admin
 - b. user3
 - c. user4
4. Display current user (**user4**) and MySQL version
5. List tables (as admin)
6. Display structures for both tables
 - a. company
 - b. customer
7. Display data for both tables:
 - a. company
 - b. customer
8. **Verify fk options:** Display query result set of customer table, including modified fk, by updating pk in parent table (company), change pk value from 1 to 6. Copy and paste SQL commands and query result sets displaying change:
9. **Verify fk options:** Display the SQL statement(s), and query result set that prevented the parent table (company) from deleting a record w/o deleting the associated child table (customer) records first. Include delete statement, and resulting error.
10. Log in as **user3**:
 - a. show the SQL INSERT statement, and corresponding query result set that prevented user3 from inserting data in the company table
 - b. show the SQL INSERT statement, and corresponding query result set that prevented user3 from inserting data in the customer table
11. Log in as **user4**:
 - a. show the SQL statement, and corresponding query result set that prevented user4 from "seeing" company table:
 - b. same as above, though, prevented from being able to delete from the customer table:
12. Log in as **admin**: remove both tables (structure and data), and show commands: