**Assignment 3 – Fall 2023**

To be posted on November. 27th

Description:

It takes 7% of your course assessment.

Please submit on or before 6pm on December. 8th

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Question** | **True / False** | **Multiple Select** | **Case Questions** | **SQL Commands** | **Total** |
| **Number** | **20 (1’)** | **15 (2’)** | **7** | **12** | **54** |
| **Mark** | **20** | **30** | **24** | **26** | **100** |

For Fill in the Blank: zero mark would be given to answers not following the correct format, as well as the ones contain typos/errors.

For Multiple Select: no partial gradings, i.e., partially correct answers are given zero mark.

Case Questions: can be both multiple-choice single-select questions and fill-in-the blank questions. You are invited to read the case descriptions presented carefully, try to understand it and then answer the questions.

**True or False**

1. Data Manipulation Language manipulate data by TRUNCATE command which remove data from the table. [×]
2. When an on-delete cascade exist, deleting a record in the parent table requires manually removing all records in the child table where a foreign key reference exists. [×]
3. COMMIT is a transaction control language (TCL) command that is used to permanently save changes to the database. [√]
4. The GRANT statement is used for revoking privileges from users in SQL. [×]
5. The INTERSECT operator is used to combine the results of two SELECT statements and remove duplicates. [×]
6. The CASE / SWITCH statement in SQL is used for data type conversion. [×]
7. The VIEW statement is used to create a virtual table based on the result of a SELECT statement. [√]
8. Data Warehouses are subject-oriented, meaning they are designed to support the needs of a specific business area. [√]
9. A Snowflake Schema is a type of schema where dimension tables are normalized, leading to more complex joins. [√]
10. OLAP databases are designed for transactional processing, focusing on day-to-day operations of an organization. [×]
11. Bitmap indexes are particularly effective for columns with a lot of unique values. [×]
12. Materialized Views store data physically and are updated in real-time with changes in the source data. [×]
13. Primary storage is non-volatile and retains data even when the power is turned off. [×]
14. Volatile storage retains data even when the power is turned off. [×]
15. On physical storage medium, a data block, also known as a sector, is the smallest addressable unit of data storage that can be read or written to a magnetic disk. [√]
16. Records in a heap file are stored in a specific order based on a key field. [×]
17. A free list pointer chain is typically used in sorted/sequential files for efficient insertion of new records. [×]
18. The page directory in a heap file is used to store metadata about the file, such as file size and permissions. [×]
19. A sparse index allows for missing index entries between the values in a clustering index. [√]
20. A multi-level index is an index structure with multiple levels, allowing for efficient navigation. [√]

**Multiple Select**

1. What is the primary purpose of the UNION operator in SQL?

A. Combine the results of two SELECT statements and remove duplicates

B. Select data from a single table

C. Remove a table from the database

D. Update existing records in a table

1. Which of the following is a DDL (Data Definition Language) command?

A. SELECT

B. UPDATE

C. CREATE TABLE

D. DELETE

1. What critical functions does the ETL (Extract, Transform, Loa process perform?
2. Transmits data between systems in real-time for seamless integration
3. Executes queries for data retrieval, focusing on dynamic updates.
4. Involves extracting, transforming, and loading data to ensure quality.
5. Implements real-time data logging for continuous monitoring.
6. How does Bitmap Indexing contribute to optimizing data retrieval?

A. Uses graphics processing for storage, enabling visual representation.

B. Employs compression to reduce image storage space.

C. Applies an indexing strategy for categorical data, specifically enhancing query performance.

D. Supports real-time data visualization with dynamic bitmap rendering.

1. When exploring data in an OLAP system, what do "Rollup" and "Drilldown" refer to?

A. Terms related to construction techniques in building design.

B. Operations in OLAP for aggregating and disaggregating data.

C. Methods for data visualization, allowing users to explore and analyze data.

D. Project management terminologies for tracking progress and milestones.

1. In a data warehouse, what is the purpose of using Materialized Views?

A. Involves real-time data visualization for instant insights.

B. Represents precomputed query results, optimizing query performance.

C. Relies on dynamic data aggregation to adapt to changing data patterns.

D. Describes a database encryption mechanism for securing stored information.

1. Which of the following statements about the INNER JOIN and LEFT JOIN is true?

A. INNER JOIN returns all rows from both tables.

B. LEFT JOIN returns unmatched rows from the right table.

C. INNER JOIN returns only matched rows from both tables.

D. LEFT JOIN returns only unmatched rows from both tables.

1. Which characteristic is NOT typically associated with Data Warehousing?

A. Subject-oriented

B. Time-variant

C. Nonvolatile

D. Real-time processing

1. Which of the following is a characteristic of Operational Databases?

A. Subject-oriented

B. Nonvolatile

C. Used for day-to-day operations

D. Optimized for complex analytical queries

1. What does the term "Attribute Hierarchy" refer to in the context of Data Warehousing?

A. A set of attributes in a dimension table

B. The structure of a Data Cube

C. A tree-like structure representing relationships between dimension tables

D. The arrangement of attributes in a table

1. What is the smallest addressable unit of data on a magnetic disk?

A. Platter

B. Sector

C. Track

D. Spindle

1. Which file organization uses linked structures without any specific order?

A. Sorted/sequential file

B. Heap file

C. Clustered file

D. Hashed file

1. What is typically used for efficient insertion of new records in a sorted/sequential file?

A. Free list pointer chain

B. Ordering pointer chain

C. Linked structure

D. Hash function

1. Sparse Index:

A. Contains every possible key value

B. Contains only a subset of key values with pointers to the actual rows

C. Is a synonym for a dense index

D. Is only used for primary keys

1. What happens during Records re-organization with insertion in a Sorted/Sequential File?

A. New records are inserted randomly

B. Records are reorganized based on deletion

C. New records are inserted while maintaining the order with ordering pointer chain

D. The file is converted into a Heap File

**Case Questions**

Based on the table below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Name** | **Age** | **Gender** | **Department** | **Rank** |
| 1 | Alice | 28 | Female | HR | Manager |
| 4 | Bob | 35 | Male | IT | Analyst |
| 5 | Charlie | 42 | Male | Finance | Director |
| 7 | David | 30 | Male | Marketing | Executive |
| 8 | Emily | 45 | Female | HR | Director |
| 10 | Frank | 38 | Male | IT | Manager |
| 11 | Grace | 32 | Female | Finance | Analyst |
| 12 | Henry | 40 | Male | Marketing | Manager |

1. Index file A:

|  |  |
| --- | --- |
| **ID** |  |
| 1 |  |
| 4 |  |
| 5 |  |
| 7 |  |
| 8 |  |
| 10 |  |
| 11 |  |
| 12 |  |

Is a clustering index (1’) [√]

Is a dense index (1’) [√]

1. Index file B:

|  |  |
| --- | --- |
| **Department** |  |
| HR |  |
| IT |  |
| Finance |  |
| Marketing |  |

Is a clustering index (1’) [×]

Is a dense index (1’) [√]

1. For the above table of data, create bitmap index table on Gender and Rank. (4’)

|  |  |  |
| --- | --- | --- |
| **Gender** | **Female** | **Male** |
| Female | 1 | 0 |
| Male | 0 | 1 |
| Male | 0 | 1 |
| Male | 0 | 1 |
| Female | 1 | 0 |
| Male | 0 | 1 |
| Female | 1 | 0 |
| Male | 0 | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rank** | Manager | Analyst | Director | Executive |
| Manager | 1 | 0 | 0 | 0 |
| Analyst | 0 | 1 | 0 | 0 |
| Director | 0 | 0 | 1 | 0 |
| Executive | 0 | 0 | 0 | 1 |
| Director | 0 | 0 | 1 | 0 |
| Manager | 1 | 0 | 0 | 0 |
| Analyst | 0 | 1 | 0 | 0 |
| Manager | 1 | 0 | 0 | 0 |

1. Leverage logical operators to use the bitmap vectors in the above created index table to quickly access the records with Gender=Male, Rank=Manager. Write the logical operation (e.g., 10010\*00111=?):(4’)

Gender=Male bit vector: 10000101

Rank=Manager bit vector: 01110101

Logical operation: Manager =1 AND Male=1 🡪 10000101 AND 01110101 =00000101

1. Create a 4-Order B+-Tree index, by directly modifying the below figure. (4’)

1

4

7

1. After inserting a new record (ID=13), update the index file and re-draw the B+ Tree. (4’)

1

4

7

1. Based on the B+ tree index updated in last step, insert another record (ID=9), update the index file and re-draw the B+ Tree. (4’)

1

4

7

**SQL commands**

With the course\_demo schema we used in the class, we have tables created by “MySQL Create tables.sql” and “MySQL Insert Data.sql”. Based on the tables, write SQL commands to achieve the below purposes.

1. Write an SQL query to retrieve the names of instructors who joined before the year 2015. (2’)

SELECT name

FROM instructor

WHERE joindate < '2015-01-01';

1. Write an SQL query to update the GPA of student '110092' to 3.75. (2’)

UPDATE student

SET gpa = 3.75

WHERE sid = '110092';

1. Write an SQL query to retrieve the names and GPAs of students who have a GPA greater than 3.5, ordered by GPA in descending order. (2’)

SELECT sname, gpa

FROM student

WHERE gpa > 3.5

ORDER BY gpa DESC;

1. Write an SQL query to create a view named 'student\_view' that shows the student ID, name, and enrollment year for students belonging to the 'Marketing' department. (2’)

CREATE VIEW student\_view AS

SELECT sid, sname, enrollyear

FROM student

WHERE dptno = '200';

1. Write an SQL query to find the total number of students in each department. (2’)

SELECT dptno, COUNT(\*) AS total\_students

FROM student

GROUP BY dptno;

1. Write an SQL query to calculate the average number of courses taught by instructors in each department. (2’)

SELECT d.dptname, AVG(i.taught) AS avg\_courses\_taught

FROM department d

LEFT JOIN instructor i ON d.dptno = i.dptno

GROUP BY d.dptname;

1. Write an SQL query to find the department names along with the total number of instructors in each department, only including departments with more than two instructors. (2’)

SELECT d.dptname, COUNT(i.iid) AS total\_instructors

FROM department d

LEFT JOIN instructor i ON d.dptno = i.dptno

GROUP BY d.dptname

HAVING COUNT(i.iid) > 2;

1. Write an SQL query to find the names of instructors who have taught the highest number of courses. If there is a tie, list all instructors. (2’)

SELECT i.name

FROM instructor i

WHERE i.taught = (SELECT MAX(taught) FROM instructor);

1. Write an SQL query to find the students who have the highest GPA in their department, along with the name of their advisor. (2’)

SELECT s.sid, s.sname, s.gpa, d.dptname AS department, i.name AS advisor

FROM student s

JOIN department d ON s.dptno = d.dptno

JOIN instructor i ON s.advisorid = i.iid

WHERE s.gpa = (SELECT MAX(gpa) FROM student WHERE dptno = s.dptno);

1. Write an SQL query to find the students who have joined at least two clubs and have a GPA greater than the average GPA of all students. (2’)

SELECT s.sname, s.gpa, COUNT(jc.cid) AS total\_clubs

FROM student s

JOIN joinclub jc ON s.sid = jc.sid

WHERE s.gpa > (SELECT AVG(gpa) FROM student)

GROUP BY s.sname, s.gpa

HAVING total\_clubs >= 2;

1. Combine the names of students and instructors who have 'o' in the third position of their names. (3’)

SELECT sname AS name

FROM student

WHERE SUBSTR(sname, 3, 1) = 'o'

UNION

SELECT name

FROM instructor

WHERE SUBSTR(name, 3, 1) = 'o';

1. Write an SQL query to display the names of students along with a column indicating their performance based on GPA: 'Excellent' if GPA > 3.8, 'Good' if GPA is between 3.5 and 3.8 (inclusive), and 'Average' otherwise. (3’)

SELECT sname,

CASE

WHEN gpa > 3.8 THEN 'Excellent'

WHEN gpa BETWEEN 3.5 AND 3.8 THEN 'Good'

ELSE 'Average'

END AS performance

FROM student;