CS 1555

Lecture 10

**Structured Query Language (SQL) – DML**

Relational query languages

- Allow manipulation and retrieval of data from a database

- User queries should be declarative, specifying what is to be retrieved; the how is the responsibility of the system

- Relational algebra and relational calculus form basis for “real” relational languages

SQL select statement

- SELECT [DISTINCT | ALL] attribute-list

FROM table-list

WHERE selection-condition

GROUP BY grouping-attribute(s)

HAVING grouping-condition

ORDER BY {attribute ASC | DESC} pairs

Execution abstraction

- Transaction is a logical unit of work in DBMSs

- Execution of a program segment that performs some function or task by accessing shared data

ACID properties

- Atomicity: either all operations associated with a transaction happen or none happen

- Consistency preservation: a transaction satisfies the integrity constraints on the database

- Isolation: transactions are independent, the result of the execution of concurrent transactions is the same as if transactions were executed one after the other

- Durability: effects of completed transactions become permanent surviving any subsequent failures

Relational operators in SQL

- PI (project) 🡨🡪 SELECT

- SIGMA (select) 🡨🡪 WHERE

Basic SQL: single table manipulation

- DISTINCT is an optional keyword indicating that the answer should not contain duplicates

- Default is that duplicates are not eliminated because it’s much faster

Aliasing in SQL: the AS operator

- Renaming attributes in the result of a query: SELECT SID AS Student\_ID FROM STUDENT;

- Table alias can be achieved with the AS operator in the FROM-clause: SELECT S.Major FROM STUDENT AS S…

- Renaming of attributes within a query: SELECT \* FROM STUDENT AS S(ID, FN, MJ)…

Aggregate functions

- Tuple grouping based on the value of some attributes

- F(B) = aggregate function on attribute B

- SUM, MIN, MAX, AVG, COUNT

Note on COUNT

- COUNT(attribute-name) doesn’t count NULLs

- COUNT(\*) returns cardinality (i.e. does count NULLs)

- COUNT(DISTINCT attribute-name) returns number of distinct values

Arithmetic Operator

- Arithmetic operators (+, -, \*, /) can be applied on numeric values in any expression

- Ex: SELECT 1.1 \* SUM(Salary) FROM LIBRARIAN;

- Increment (+) and decrement (-) operators can be applied on data types: date, time, timestamp

Grouping of tuples

- Tuple grouping based on the value of some attributes

- Grouping attributes must appear in SELECT statement to be meaningful

- WHERE is evaluated first and then the grouping is done

Sorting the result

- ORDER BY order-list

- order-list: list of <attribute, order> pairs

- order: ASC (default), DESC)

- attribute relative position is allowed: 2 ASC, 1 DESC

- Ex: SELECT \* FROM STUDENT WHERE QPA >= 3.5 ORDER BY LName, ASC, FName ASC, MI DESC;

Manipulating NULL values

- NULL values must be considered explicitly

- IS NULL and IS NOT NULL

- NULL in a condition yields UNKNOWN

- SQL provides operators to test for specific conditions

- IS FALSE and IS NOT FALSE

- IS TRUE and IS NOT TRUE

- IS UNKNOWN and IS NOT UNKNOWN