CSCD 327: Relational Database Systems

Database updates

Instructor: Dr. Dan Li

Modification of the Database

- **Deletion** of tuples from a given relation
- **Insertion** of new tuples into a given relation
- **Updating** values in some tuples in a given relation

SQL – Deletion

- Delete all customers

  delete from *customer*

  cannot use *delete * from instructor*

  delete from ONE table only (no *join*)

- Delete all account records in the Perryridge branch.

  delete from *account*
  where *branch_name*= 'Perryridge';

- Delete all accounts at branches located in Needham.

  delete from *account*
  where *branch_name* **in** (**select** *branch_name*
  from *branch*
  where *branch_city='Needham'*);

SQL - Deletion (Cont.)

- Delete all instructors whose salary is less than the average salary of instructors

  delete from *instructor*
  where *salary*< (**select avg** (*salary*) **from** *instructor*);

Problem: as we delete tuples from *instructor*, the average salary changes

Solution:

First, compute **avg** salary and find all tuples to delete (create a temporary table)

Next, delete all tuples found above (without recomputing **avg** or retesting the tuples)

Error message: You can't specify target table 'instructor' for update in FROM clause.

You can't modify the same table which you use in the SELECT part.

SQL – Insertion

- Add a new tuple to *course*

  insert into *course*
  values (’CS-437’, ’Database Systems’, ’Comp. Sci.’, 4);

- or equivalently

  insert into *course* (*course_id*, *title*, *dept_name*, *credits*)
  values (’CS-437’, ’Database Systems’, ’Comp. Sci.’, 4);

- Add a new tuple to *student* with *tot_creds* set to null

  insert into *student*
  values (’3003’, ’Green’, ’Finance’, *null*);

SQL - Insertion (Cont.)

- Add all instructors to the *student* relation with tot_creds set to 0

```
insert into student
   select ID, name, dept_name, 0
  from   instructor
```

- The **select from where** statement is evaluated fully before any of its results are inserted into the relation (otherwise queries like
  insert into *table*1 **select** * **from** *table*1
  would cause problems)

SQL – Updates

- Increase salaries of instructors whose salary is over $100,000 by 3%, and all others receive a 5% raise
  - Write two **update** statements:

```
update instructor
set salary = salary * 1.03
where salary > 100000;
update instructor
set salary = salary * 1.05
where salary <= 100000;
```

  - The order is important (how about switch the above two updates?)
  - Can be done better using the **case** statement (next slide)

Case Statement for Conditional Updates

- Same query as before but with case statement

```
update instructor
  set salary = case
                 when salary <= 100000 then salary * 1.05
                 else salary * 1.03
               end
```