

## Database Management System – 23 (SQL – INSERT, DELETE, UPDATE)

Ajay James  
Asst. Prof in CSE  
Government Engineering College Thrissur

### Outline

- INSERT
- DELETE
- UPDATE
- Referential integrity constraint naming (lecture 22)
- Exercise - 6

## INSERT Command

- INSERT is used to add a single tuple (row) to a relation (table)
- Must specify the relation name and a list of values for the tuple
- Values should be listed in the same order in which the corresponding attributes were specified in the CREATE TABLE command

```
INSERT INTO EMPLOYEE
VALUES ('Richard', 'K', 'Marini', '653298653', '1962-12-30', '98
Oak Forest, Katy, TX', 'M', 37000, '653298653', 4 );
```

## INSERT

```
INSERT INTO EMPLOYEE (Fname, Lname, Dno, Ssn)
VALUES ('Richard', 'Marini', 4, '653298653');
```

- Specify explicit attribute names that correspond to the values provided in the INSERT command
- Useful if a relation has many attributes but only a few of those attributes are assigned values in the new tuple

## INSERT Violation

**INSERT INTO**            **EMPLOYEE** (Fname, Lname, Ssn, Dno)  
**VALUES**                ('Robert', 'Hatcher', '980760540', 2);

**INSERT INTO**            **EMPLOYEE** (Fname, Lname, Dno)  
**VALUES**                ('Robert', 'Hatcher', 5);

## DELETE Command

- DELETE command removes tuples from a relation

**DELETE FROM**            **EMPLOYEE**  
**WHERE**                    Lname = 'Brown';

- WHERE clause - to select the tuples to be deleted
- Tuples are explicitly deleted from only one table at a time
- Deletion may propagate to tuples in other relations if referential triggered actions are specified
- Zero, one, or several tuples can be deleted by a single DELETE command

## Delete Examples

```
DELETE FROM EMPLOYEE  
WHERE Ssn = '123456789';
```

```
DELETE FROM EMPLOYEE  
WHERE Dno = 5;
```

```
DELETE FROM EMPLOYEE;
```

## UPDATE Command

- Used to modify attribute values of one or more selected tuples
- WHERE clause in the UPDATE command selects the tuples to be modified from a single relation
- Updating a primary key value may propagate to the foreign key values of tuples in other relations

## UPDATE Examples

```
UPDATE PROJECT
SET Plocation = 'Bellaire', Dnum = 5
WHERE Pnumber = 10;
```

```
UPDATE EMPLOYEE
SET Salary = Salary * 1.1
WHERE Dno = 5;
```

## Examples of Referential triggered action

```
CREATE TABLE EMPLOYEE
(
  ...,
  Dno INT NOT NULL DEFAULT 1,
  CONSTRAINT EMPPK
  PRIMARY KEY (Ssn),
  CONSTRAINT EMPSUPERFK
  FOREIGN KEY (Super_ssn) REFERENCES EMPLOYEE(Ssn)
  ON DELETE SET NULL ON UPDATE CASCADE,
  CONSTRAINT EMPDEPTFK
  FOREIGN KEY (Dno) REFERENCES DEPARTMENT(Dnumber)
  ON DELETE SET DEFAULT ON UPDATE CASCADE);

CREATE TABLE DEPARTMENT
(
  ...,
  Mgr_ssn CHAR(9) NOT NULL DEFAULT '888665555',
  ...,
  CONSTRAINT DEPTPK
  PRIMARY KEY (Dnumber),
  CONSTRAINT DEPTSK
  UNIQUE (Dname),
  CONSTRAINT DEPTMGRFK
  FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn)
  ON DELETE SET DEFAULT ON UPDATE CASCADE);

CREATE TABLE DEPT_LOCATIONS
(
  ...,
  PRIMARY KEY (Dnumber, Dlocation),
  FOREIGN KEY (Dnumber) REFERENCES DEPARTMENT(Dnumber)
  ON DELETE CASCADE ON UPDATE CASCADE);
```

## Exercise – 6

1. Insert a few employees in the database using 2 types of insert commands
2. Change the department of any employee
3. Insert a new department
4. Insert a dependent
5. Update the employee table
6. Insert 3 rows in Works\_on relation
7. Delete some records from the tables

Thank you