

Constraints and security in SQL (Ch. 8.6, Ch22.2)

Database Management
COP4540, SCS, FIU

SQL Constraints

- Primary Key or Foreign key Constraints

```
CREATE TABLE Students (  
  SSN CHAR(9),  
  Dept CHAR(4),  
  name VARCHAR(20),  
  address VARCHAR(40)  
  CONSTRAINT SnnIsKey PRIMARY KEY (SSN),  
  CONSTRAINT DeptIsFK FOREIGN KEY (Dept) REFERENCE DEPT(Name),  
);
```

- Altering Constraint on Tables

- Using ALTER TABLE command
 - ALTER TABLE DROP CONSTRAINT constraintName
 - ALTER TABLE ADD CONSTRAINT constraintName constraintDef

Database Management
COP4540, SCS, FIU

Attribute-Based CHECK Constraints

- More complex constraints can be attached to an attribute declaration by keyword CHECK.
- The condition can be anything that could follow WHERE in an SQL query.
- Example
gender CHAR(1) CHECK (gender IN ('F', 'M'))
- An attribute-based CHECK constraint is checked whenever any tuple gets a new value for this attribute.

Database Management
COP4540, SCS, FIU

Global Constraints

- Complex constraints that involve relationships between several attributes or even several different relations.
- Tuple-Based CHECK constraints, which restrict any aspect of the tuples of a single relation.
- Assertions, which are constraints that may involve entire relations or several tuple-variables ranging over the same relation.

Database Management
COP4540, SCS, FIU

Tuple-based Constraints

- When define the table with a CREATE TABLE statement, we can add a CHECK followed by a parenthesized condition.
- Such condition can be anything that could appear in a WHERE clause.
 - Condition may involve all the attributes in the table
 - Other table may be involved, but only in the sub-queries.
- It is checked every time a tuple is inserted into and every time a tuple of the relation is updated.
- A tuple-based CHECK is invisible to other relations.

Database Management
COP4540, SCS, FIU

Assertions

- The Keywords CREATE ASSERTION
- The name of the assertion.
- The key word CHECK.
- A parenthesized condition.
 - CREATE ASSERTION <name> CHECK <condition>
- Condition is Checked for each modification that could potentially violate it.
- An assertion can be dropped by a statement consisting of the keywords DROP ASSERTION followed by the name of assertion.

DROP ASSERTION assertionName

Database Management
COP4540, SCS, FIU

Examples

- **Attribute-Based constraint**

```
ALTER TABLE Student ADD CONSTRAINT checkGender
CHECK (gender IN ('F', 'M'));
```

- **Tuple-based constraint**

```
ALTER TABLE Student ADD CONSTRAINT rightTitle
CHECK ( gender = 'F' AND title LIKE 'Ms. %' OR
gender = 'M' AND title LIKE 'Mr. %');
```

- **Assertion**

```
CREATE ASSERTION Salary_Constraint
CHECK ( NOT EXISTS
(SELECT * FROM Emp E1, Emp E2
WHERE E1.Salary > E2.Salary AND
E1.MGRSSN = E2.SSN));
```

Database Management
COP4540, SCS, FIU

Comparison of Constraints

Type of Constraint	When Activated	Guaranteed to Hold
Attribute Based	On insertion to relation or attribute update	Not if sub-queries
Tuple Based	On insertion to relation or tuple update	Not if sub-queries
Assertion	On change to any mentioned relation	Yes

Database Management
COP4540, SCS, FIU

Security and user authorization in SQL2

- Six Privileges
 - SELECT, INSERT, DELETE, and UPDATE
 - Applied to a relation, either a base table or a view.
 - REFERENCES
 - The right to refer to the relation in an integrity constraint.
 - USAGE
 - Defined on a domain or on several other kind of schema elements other than relations and assertions.

Database Management
COP4540, SCS, FIU

Creating privilege

- When a schema is created, it and all the tables and other schema elements in it are assumed owned by the user who create it. Thus, the user has all the privileges.
- Then, the privileges can be passed from user to user.

Database Management
COP4540, SCS, FIU

Granting privilege

- Using GRANT statement.

GRANT *privilege-list* ON *db-element* TO *user-list*

- the *privilege-list* include: SELECT, UPDATE, ..., or optionally the key words ALL PRIVILEGES.
- The *db-element* may be a base table, a view, a domain, or other elements.
- At the end of statement, there is an option:
 - WITH GRANT OPTION
 - With this option, the user who receive the privilege has the authorization to pass the privilege to another user.

Database Management
COP4540, SCS, FIU

Revoking privilege

- Using REVOKE statement

REVOKE *privilege-list* ON *db-element* FROM *user-list*

- Two options to see if any other granted privileges need to be revoked.
 - CASCADE
 - When the specified privileges are revoked, we also revoke any privileges that were granted only because of the revoked privilege.
 - RESTRICT
 - the revoke statement cannot be executed if the cascading rule would result in the revoking of any privileges due to the revoked privileges having been passed on to others.
- Revoke only the grant option
 - Replace REVOKE by REVOKE GRANT OPTION FOR

Database Management
COP4540, SCS, FIU

Grant diagram

- The diagram is a graph whose nodes correspond to a user and a privilege, saying $\langle \text{User/Privilege} \rangle$.
- If a User_1 grants privilege P to user User_2 , and this grant was based on the fact that U holds privilege Q , then we draw an arc from the node for $\langle \text{User}_1, Q \rangle$ to the node for $\langle \text{User}_2, P \rangle$.
- A $*$ after a user-privilege combination indicates that the privilege includes the grant option.
- $**$ after a user-privilege combination indicates that the privilege derives from ownership of the database element and was not due to a grant of the privilege from elsewhere.

Database Management
COP4540, SCS, FIU

Example

Tables: T_1 and T_2

Users: U_1, U_2, U_3 , and U_4

Suppose that U_1 is the owner of the tables T_1 and T_2 .

Statements:

U_1 :

GRANT SELECT, INSERT ON T_1 TO U_2, U_3 WITH GRANT OPTION;

GRANT SELECT ON T_2 TO U_2, U_3 WITH GRANT OPTION;

U_2 :

GRANT SELECT, INSERT ON T_1 TO U_4 ;

GRANT SELECT ON T_2 TO U_4 ;

U_3 :

GRANT SELECT, INSERT(name) ON T_1 TO U_4 ;

GRANT SELECT ON T_2 TO U_4

Database Management
COP4540, SCS, FIU

