CS 1555

Lecture 5

**Relational Database Model (continued)**

Foreign key constraints

- If constraints are enforced, referential integrity is achieved

- Like a “logical pointer” – no dangling references

- Either valid value or NULL if not part of a primary key

Referential integrity enforcement

- What are alternatives when a tuple in R1 is deleted?

1. Delete all R2 tuples that refer to it

- CASCADE in SQL

2. Disallow deletion of a R1 tuple that is referred to

- NO ACTION in SQL (default action taken)

3. Set the foreign key in R2 to some default

- SET DEFAULT in SQL

4. If the foreign key is not part of the primary key in R2, set it to NULL

- SET NULL in SQL

**SQL – Data Definition Language**

Create database 🡪 Create schema 🡪 Create table

- Schema is essentially a namespace: it contains named objects

- RESTRICT lets schema be removed only if it has no objects

- CASCADE will let everything be removed, including data and definitions

SQL data types: numeric, character strings, bit strings, temporal data

- NULL value valid for all types

Constraints on attributes: NOT NULL, DEFAULT value, PRIMARY KEY, UNIQUE, FOREIGN KEY

Observations on numeric types

- Like the datatype in C

- BIGINT for long integer or integer

- Truncation is towards 0

- Rounding is business instead of scientific

- Scientific: [0..4] rounds to 0, [5..9] rounds to 1

- Business: [0..4] rounds to 0, [6..9] rounds to 1, 5 rounds up 50% of the time and down 50% of the time

- Money or Currency data are numeric data with currency sign

Comment is double hyphen