CS2102 Midterm Cheatsheet v1.0 (2021-04-28)
by Aadit Rahul Kamat, page 1 of 1

1 Relational Algebra

- Superkey: combo of attributes that uniquely defines its tuples
- Key (e.g candidate key, foreign key) : A type of superkey that is minimal: no proper subset of key is a superkey
- Domain: a set of atomic values
  - Each value of an attribute A_i is either domain(A_i) or null
- Foreign key constraint: each foreign key value in referencing relation must either:
  - appear as primary key value in referenced relation
    or
  - be a null value
- Set operators require input relations to be union compatible i.e. two relations that:
  - have the same number of attributes
  - the corresponding attributes have the same domains
- Union compatible relations do not necessarily use the same attribute names
- Cross product:

$$R \times S = (a,b,c,x,y)|(a,b,c)R,(x,y)S$$

- Join: Combines cross product and selection (and possibly projection)
  - Inner:

$$R_c S = {}_c(R \times S)$$

  - Natural: Inner join based on renamed attributes

$$R_c S = (R_{c_{a_1:b_1,\ldots,a_n:b_n}}(S))$$

  - A **dangling tuple** is a tuple in a join operand that does not participate in the join
- Outer:
  - Left:

$$R_c S = (R_c S)(dangle(RcS) \times null(S))$$

  - Right:

$$R_c S = (R_c S)(null(R) \times dangle(S_c R))$$

  - Full:

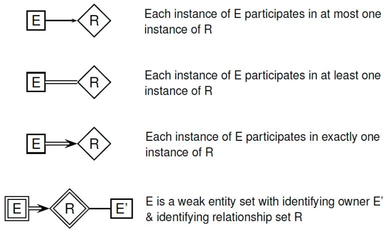
$$R_c S = (R_c S)(null(R) \times (S_c R))$$

2 ER diagrams

- Relationship roles are shown explicitly when one entity set appears two or more times in a relationship set

$$Key(R) = A \cup \bigcup_{E_i \in E} Key(E_i)$$

- Participation constraints:



Each instance of E participates in at most one instance of R

Each instance of E participates in at least one instance of R

Each instance of E participates in exactly one instance of R

E is a weak entity set with identifying owner E' & identifying relationship set R

3 SQL

- Comments: - - or /* */
- Commands:
  - **create table**
  - **drop table if exists** <tablename> **cascade**;
  - **insert into** <tablename> values (<values>);
  - **delete from** <tablename> where <condition>;
  - **FOREIGN KEY** ... **REFERENCES** ... **ON DELETE/UPDATE action**;
  - **alter table** Students **alter column** dept **drop default**;
  - **alter table** Students **drop column** dept;
  - **alter table** Students **add constraint** fk_grade **foreign key** (grade) **references** Grades;
- Constraint Specifications:
  - Column constraints
  - Table constraints
  - Assertions
- Constraint Types:
  - Not-null constraints
  - Unique constraints
  - Primary key constraints
  - Foreign key constraints
  - Check constraints
- Constraint violations:
  - NO ACTION: rejects delete/update if it violates constraint (default option)
  - RESTRICT: similar to NO ACTION except that constraint checking cant be deferred
  - CASCADE: propagates delete/update to referencing tuples
  - SET DEFAULT: updates foreign keys of referencing tuples to some default value
  - SET NULL: updates foreign keys of referencing tuples to null value
- Transaction: begin; <SQL Commands> commit;
- Deferrable constraints:
  - deferrable initially deferred
  - deferrable initially immediate
    Eg: constraint employees_fkey foreign key
    (managerId) references Employees
    deferrable initially immediate