

Hvad er Relational Algebra (RA)? Kom herunder ind på: relation, tuple, set og operatorer (select, projection, join...) samt RA' betydningen for opbygningen af SQL (Structured Query Language) erklæringer

En relationel database defineres ud fra relationel algebra. Et relationelt komplet sprog kan udføre alle basale og meningsfulde operationer på relationer. Da SQL er afledt af relationel algebra, er det relationelt komplet.

Relational Algebra er:

- Den formelle beskrivelse af, hvordan en relationel database fungerer.
- an interface to the data stored in the database itself
- Matematikken bag SQL-operationerne (Men operationerne er ikke ens)

Terminologi:

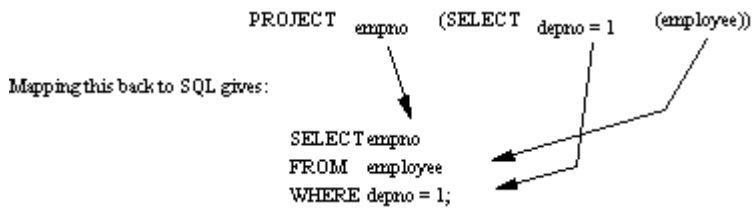
- Relation et sæt af tupler (Tabel)
- Tuple (en række) - er en samling af attributter som beskriver en hel entitet.
- Attribute - a real world role played by a named domain.
- Domæn – er et set af atomare værdier (f.eks. int, string)
- Set – en matematisk definition for en samling af objekter som ikke indeholder nogen dupletter.

Operator:

Operation	My HTML	Symbol
Projection	PROJECT	π
Selection	SELECT	σ
Renaming	RENAME	ρ
Union	UNION	\cup
Intersection	INTERSECTION	\cap
Assignment	\leftarrow	\leftarrow

Operation	My HTML	Symbol
Cartesian product	X	\times
Join	JOIN	\bowtie
Left outer join	LEFT OUTER JOIN	\bowtie_{L}
Right outer join	RIGHT OUTER JOIN	\bowtie_{R}
Full outer join	FULL OUTER JOIN	\bowtie_{F}
Semijoin	SEMIJOIN	\ltimes

Relationel algebra til SQL:



Join:

Der er mange forskellige slags joins. Join bruges til at kombinere relaterede tupler fra to relationer. Join laves for at vise meningsfuldt data. Join kan kræve kompatible attributter på baggrund af join-betingelse.

Cartesisk produkt (cross join): joiner alle tupler i den ene tabel med alle tupler i den anden tabel.

Natural join: Sætter tupler fra to relationerne (tabeller) sammen, hvor de har en fælles attribut (samme navn og type (kaldes også EQUI-JOIN)). Dette fjerner duplikerede attributter

Inner join: Kombinerer to relationer på baggrund af et kriterie med identiske attributter.

Left outer join: Tager alle tupler fra den venstre relation og kombinerer dem med de kompatible tupler (på baggrund af et kriterie) fra den højre relation.

Full outer join: Tager alle tupler fra begge relationer og kombinerer dem på baggrund af et kriterie. De relationer, der ikke er kompatible med kriteriet bliver også taget med.

Set Operationer:

En set operation er en forening/sammensætning af to identiske relationer på baggrund af forskellige kriterier.

Det kræver, at de to relationer den udføres på er "union compatible":

- De har samme antal af attributter
- Kolonnerækkefølgen i de to relationer er ens.

Union:

- Foreningen af to relationer i en ny relation, der omfatter alle tupler. Duplikerede tupler er elimineret.

Intersection:

- Sammensætning af to relationer, der omfatter af alle tupler som er i begge tupler.

Difference:

- Sammensætning af to relationer, der omfatter af alle de tupler, der er i den ene relation, men ikke i den anden.

Eksempel:

SQL:

```
SELECT DISTINCT dname
FROM department, course, empcourse, employee
WHERE cname = 'Further Accounting'
  AND course.courseno = empcourse.courseno
  AND empcourse.empno = employee.empno
  AND employee.depno = department.depno;
```

The equivalent relational algebra is

```
PROJECTdname (department JOINdepno = depno (
PROJECTdepno (employee JOINempno = empno (
  PROJECTempno (empcourse JOINcourseno = courseno (
    PROJECTcourseno (SELECTcname = 'Further Accounting' course)
  ))
))
))
```