

Relationel Algebra

Indholdsfortegnelse

Relationel Algebra	1
Indholdsfortegnelse	1
Operationer på den relationelle model	2
Mængdeoperationerne	2
Union	2
Difference.....	2
Intersection	3
Hvilke relationer er union-kompatible?.....	3
Andre egenskaber ved operationerne	3
Kartesisk Produkt	4
De specifikke relationelle operationer - Selektion og Projektion	4
Selektion operationen	4
Sammenligningsoperatorer	5
Logiske operatorer.....	5
Eksempler på Selektion	5
Projektion operationen.....	7
Eksempler på Projektion.....	8
Kombineret anvendelse af Selektion og Projektion.....	8
Aggregatfunktioner	9
Eksempler på aggregatfunktioner.....	9
Join-operationer	10
Generisk join.....	10
Eksempler på generiske join-operationer	11
Natural Join-operationer.....	12
Ydre Join (outer join).....	12
Venstre ydre join.....	13
Højre ydre join.....	13
Fuld ydre join.....	13
Ydre Union (Outer Union).....	14

Operationer på den relationelle model

- Relationel Algebra er en samling af **Operationer** på **Relationer**. (husk den særlige og korrekte betydning af begrebet relation!)
- Relationerne er operander og resultatet af en operation er en anden relation.
- Man skelner mellem to hovedtyper af relationelle operationer:
 1. Mængdeoperationerne (Set operations):
Union, Intersection, Difference og Kartesisk produkt.
 2. De specifikke relationelle operationer:
Selektion, Projektion, Join, Division

Mængdeoperationerne

Vi bruger følgende relationer: **R** og **S**

R

Fornavn	Efternavn	Alder
Villy	Smith	22
Hanne	Green	28
Marie	Keldsen	23
Tom	Johansen	32

S

Fornavn	Efternavn	Alder
Frederik	Gundersen	36
Hanne	Green	28
Dennis	Markussen	27

Union

$R \cup S$ (Foreningsmængden)

Resultat: Relation med tupler fra R og S med evt. duplikater fjernet.

Difference

$R - S$ (Mængdedifferensen)

Resultat: Relation med tupler fra R men ikke fra S

Intersection

$R \cap S$ (Fællesmængden)

Resultat: Relation med tupler som findes både i R og S.

Resultaterende relationer:

$R \cup S$

Fornavn	Efternavn	Alder
Villy	Smith	22
Hanne	Green	28
Marie	Keldsen	23
Tom	Johansen	32
Frederik	Gundersen	36
Dennis	Markussen	27

$R - S$

Fornavn	Efternavn	Alder
Villy	Smith	22
Marie	Keldsen	23
Tom	Johansen	32

$R \cap S$

Fornavn	Efternavn	Alder
Hanne	Green	28

Hvilke relationer er union-kompatible?

- Relationernes attributter behøver ikke være identiske (ie: have samme attributnavn) for at man kan udføre operationerne: union, intersection og difference.
- De skal imidlertid have samme **relationsgrad** (ie: antal attributter), og korresponderende attributter skal have samme **domæne** (ie: attributtens datatype og størrelse).

Andre egenskaber ved operationerne

- Union, Intersection og difference operatorer må kun anvendes ved union-kompatible relationer.
- Union og Intersection er kommutative operationer, idet

$$R \cup S = S \cup R \text{ og}$$

$$R \cap S = S \cap R$$
- Difference operationen er IKKE kommutativ, idet

$$R - S \neq S - R$$

Kartesisk Produkt

- Resultatet af operationen er en relation, der indeholder alle kombinationer af tupler fra to relationer.

R

Fornavn	Efternavn	Alder
Villy	Smith	22
Marie	Keldsen	23
Tom	Johansen	32

S

Hovedret	Dessert
DanskBøf	Rødgrød
Kylling	BananaSplit

R × S

Fornavn	Efternavn	Alder	Hovedret	Dessert
Villy	Smith	22	DanskBøf	Rødgrød
Villy	Smith	22	Kylling	BananaSplit
Marie	Keldsen	23	DanskBøf	Rødgrød
Marie	Keldsen	23	Kylling	BananaSplit
Tom	Johansen	32	DanskBøf	Rødgrød
Tom	Johansen	32	Kylling	BananaSplit

De specifikke relationelle operationer - Selektion og Projektion

Både Selektion og Projektion er **unære** operationer (ie: de anvender kun én operand).

Selektion operationen

- Anvendt symbol: σ (sigma).
- Selektion operationen fungerer som et filter på en relation ved at den kun returnerer et bestemt antal tupler.
- Den resulterende relation får samme relationsgrad som den originale relation.
- Den resulterende relation kan få færre tupler end den originale relation.
- De tupler, der returneres er afhængige af en **kondition**, (betingelse) som er en del af selektion operationen.

Eksempel : $\sigma_C(\mathbf{R})$ returnerer kun de tupler i \mathbf{R} , som tilfredsstiller konditionen \mathbf{C}

En kondition \mathbf{C} kan bestå af enhver kombination af sammenlignende eller logiske operatorer som opererer på \mathbf{R} 's attributter.

Sammenligningsoperatører

Sammenligningsoperatører, (relational operators), er følgende: (øverste tabelrække viser de sædvanlige symboler i sproget C (C++) eller Java, og nederste tabelrække de traditionelle symboler i relationel algebra)

Lig med	Mindre end	Større end	Større end eller lig med	Mindre end eller lig med	Forskellig fra
==	<	>	>=	<=	!=
=	<	>	≥	≤	≠

Logiske operatører

Logiske operatører er følgende: (igen er sidste række symbolerne fra relationel algebra)

Og	Eller	Ikke
&&		!
^	∨	¬

Sandhedstabellerne for de logiske operatører er som følger:

^	T	F
T	T	F
F	F	F

∨	T	F
T	T	T
F	T	F

¬	
T	F
F	T

Eksempler på Selektion

Eksemplerne anvender relationen ANSAT med følgende tupler og attributter:

Navn	Lokale	Afdeling	Stilling
Smith	400	EDB	Assistent
Johansen	220	Regnskab	Fuldmægtig
Green	160	Regnskab	Assistent
Brun	420	EDB	AfdLeder
Smith	500	Finans	AfdLeder

Opgave 1:

Udvælg kun de ansatte i EDB afdelingen:

Operation	Kondition	Relation
σ	Afdeling = 'EDB'	ANSAT
Formel:	$\sigma_{\text{Afdeling} = \text{'EDB'}}(\text{ANSAT})$	

Resulterende relation:

Navn	Lokale	Afdeling	Stilling
Smith	400	EDB	Assistent
Brun	420	EDB	AfdLeder

Opgave 2:

Udvælg kun Ansatte ved **Navn** Smith, som har **Stillingen** Assistent.

Operation	Kondition	Relation
σ	Navn = 'Smith' \wedge Stilling = 'Assistent'	ANSAT
Formel:	$\sigma_{\text{Navn} = \text{'Smith'} \wedge \text{Stilling} = \text{'Assistent'}}(\text{ANSAT})$	

Resulterende relation:

Navn	Lokale	Afdeling	Stilling
Smith	400	EDB	Assistent

Opgave 3:

Udvælg kun de Ansatte, som er enten har **stillingen** AfdLeder eller er i Regnskabs-**Afdelingen**

Operation	Kondition	Relation
σ	Stilling = 'AfdLeder' \vee Afdeling= 'Regnskab'	ANSAT
Formel:	$\sigma_{\text{Stilling} = \text{'AfdLeder'} \vee \text{Afdeling} = \text{'Regnskab'}} (\text{ANSAT})$	

Resulterende relation:

Navn	Lokale	Afdeling	Stilling
Brun	420	EDB	AfdLeder
Johansen	220	Regnskab	Fuldmægtig
Green	160	Regnskab	Assistent
Smith	500	Finans	AfdLeder

Opgave 4:

Udvælg kun de Ansatte, som IKKE er i EDB-**afdelingen** ELLER har **stillingen**_Fuldmægtig:

Operation	Kondition	Relation
σ	$\neg (\text{Afdeling} = \text{'EDB'} \vee \text{Stilling} = \text{'Fuldmægtig'})$	ANSAT
Formel:	$\sigma_{\neg (\text{Afdeling} = \text{'EDB'} \vee \text{Stilling} = \text{'Fuldmægtig'})} (\text{ANSAT})$	

Resulterende relation:

Navn	Lokale	Afdeling	Stilling
Green	160	Regnskab	Assistent
Smith	500	Finans	AfdLeder

Projektion operationen

Projektion er som tidligere nævnt også en unær operation (anvender kun én operand).

- Anvendt symbol: Π (π).
- Projektion begrænser antallet af attributter, som vil blive returneret fra den originale relation
- Den resulterende relation vil have det samme antal tupler som den originale relation.
- Relationsgraden på den resulterende relation kan være lig med eller mindre end i den originale relation.

Syntakseksempel: $\Pi_{\text{attributliste}} (\mathbf{R})$

Attributliste omfatter de attributter, der skal vises i den resulterende relation, og (R) er navnet på den originale relation

Eksempler på Projektion

Vi genanvender relationen ANSAT fra det foregående.

Opgave 5:

Vis kun **Navn** og **Afdeling** fra tuplerne i ANSAT

Syntaks: $\Pi_{\text{Navn,Afdeling}}(\text{ANSAT})$

Resulterende relation:

Navn	Afdeling
Smith	EDB
Johansen	Regnskab
Green	Regnskab
Brun	EDB
Smith	Finans

Kombineret anvendelse af Selektion og Projektion

Selektion og Projektion operationerne kan kombineres.

Eksempel på kombineret brug:

Opgave 6:

Vis **Navn** på alle Ansatte i EDB-afdelingen

Syntaks: $\Pi_{\text{Navn}}(\sigma_{\text{Afdeling}='EDB'}(\text{ANSAT}))$

Resulterende relation:

Navn
Smith
Brun

Opgave 7:

Vis **Navn** og **Stilling** på de Ansatte, som IKKE er i EDB-Afdelingen ELLER har **stillingen** Fuldmægtig.

Syntaks:

$\Pi_{\text{Navn, Stilling}}(\sigma_{\neg(\text{Afdeling}='EDB' \vee \text{Stilling}='Fuldmægtig')}(\text{ANSAT}))$

Resulterende relation:

Navn	Stilling
Green	Assistent
Smith	AfdLeder

Aggregatfunktioner

Aggregatfunktioner er operationer, der opererer på én af relationens attributter vertikalt gennem dens tupler.

- SUM
- MINIMUM (MIN)
- MAXIMUM (MAX)
- AVERAGE, MEAN, MEDIAN (AVG, MEAN, MED)
- COUNT

Aggregatfunktioner returnerer en simpel (ie: ikke sammensat) værdi.

Syntaks: \mathcal{F} Funktionsnavn (attributnavn) (Relationsnavn)

Eksempler på aggregatfunktioner

Vi genbruger delvis den originale relation ANSAT fra før:

Navn	Lokale	Afdeling	Gage
Smith	400	EDB	24000
Johansen	220	Regnskab	32000
Green	160	Regnskab	21000
Brun	420	EDB	41000
Smith	500	Finans	37000

Opgave 8:

Find minimums gagen!

Syntaks: \mathcal{F} MIN (Gage) (ANSAT)

Resultat:

MIN(Gage)
21000

Opgave 9:

Find gennemsnits-gagen

Syntaks: \mathcal{F} AVG (Gage) (ANSAT)

Resultat:

AVG(Gage)
31000

Opgave 10:

Tæl antallet af Ansatte i EDB-afdelingen

Syntaks: \mathcal{J} COUNT (Navn) ($\sigma_{\text{Afdeling} = \text{'EDB'}}(\text{ANSAT})$)

Resultat:

COUNT(Navn)
2

Kommentar:

Funktionen skal have et attributnavn at arbejde på, vi vælger attributten Navn, efter først at have anvendt en Selektion operation til at returnere en ny relation, der kun indeholder tupler med afdeling = 'EDB'

Opgave 11:

Find den totale gage-sum for Regnskabs-afdelingen

Syntaks: \mathcal{J} SUM (Gage) ($\sigma_{\text{Afdeling} = \text{'Regnskab'}}(\text{ANSAT})$)

Resultat:

SUM(Gage)
155000

Join-operationer

Generisk join

Join operationer sammenføjer to relationer til én, idet deres attributter og tupler kombineres på en specificeret måde. Join-operatorens argumenter er attributnavne fra begge de to relationer, der ønskes sammenføjet.

Generisk Join symbolet er: \bowtie

Eksempler på generiske join-operationer

Vi genbruger den tidligere anvendte relation ANSAT:

Navn	Lokale	Afdeling	Gage
Smith	400	EDB	24000
Johansen	220	Regnskab	32000
Green	160	Regnskab	21000
Brun	420	EDB	41000
Smith	500	Finans	37000

Desuden findes relationen PRIVAT med følgende indhold:

Navn	Telefon	Firmabil	Gratiale
Smith	6612 1414	Null	18000
Johansen	6611 1337	XA24.547	35000
Brun	6579 4453	TA54.431	35000
Smith	6224 2423	OE65.561	25000
Boss	Hemmeligt	KK18.111	150000

Opgave 12:

Find al information om hver Ansat fra begge relationer:

Syntaks:

Relation 1	Operator	Kondition	Relation 2
ANSAT	⋈	Navn = Navn	PRIVAT

Resulterende relation:

Navn	Lokale	Afdeling	Gage	Navn	Telefon	Firmabil	Gratiale
Smith	400	EDB	24000	Smith	6612 1414	Null	18000
Johansen	220	Regnskab	32000	Johansen	6611 1337	XA24.547	35000
Brun	420	EDB	41000	Brun	6579 4453	TA54.431	35000
Smith	500	Finans	37000	Smith	6224 2423	OE65.561	25000

Join operationer, hvor konditionens operator er = (lig med) kaldes for **Equi-joins**.

Konditionen kan anvende alle sammelnende og logiske operatorer som vist på side 4.

Bemærk, at attributter med samme navn gentages i den resulterende relation.

Opgave 13:

Find al information om Ansatte, hvis **gratiale** overstiger deres **gage**.

Syntaks:

Relation 1	Operator	Kondition	Relation 2
ANSAT	\bowtie	$(\text{Navn} = \text{Navn}) \wedge (\text{Gratiale} > \text{Gage})$	PRIVAT

Resulterende relation:

Navn	Lokale	Afdeling	Gage	Navn	Telefon	Firmabil	Gratiale
Johansen	220	Regnskab	32000	Johansen	6611 1337	XA24.547	35000

Natural Join-operationer.

Bemærk som tidligere nævnt, at den generiske join operation returnerer en relation, hvor fælles attributter gentages.

Natural Join operationen fjerner disse duplikate attributter.

Natural Join symbolet er *

Natural join anvendes som en Equi-Join på de fælles attributter, hvoraf den ene (duplikatet) fjernes i den resulterende relation.

Eksempel:

Syntaks:

Relation 1	Operator	Relation 2
ANSAT	*	PRIVAT

Resulterende relation: (attributten Navn forekommer kun én gang)

Navn	Lokale	Afdeling	Gage	Telefon	Firmabil	Gratiale
Smith	400	EDB	24000	6612 1414	Null	18000
Johansen	220	Regnskab	32000	6611 1337	XA24.547	35000
Brun	420	EDB	41000	6579 4453	TA54.431	35000
Smith	500	Finans	37000	6224 2423	OE65.561	25000

Ydre Join (outer join)

Vi har hidtil udelukkende arbejdet med join operationer, der returnerer en relation med tupler, hvor de attributter vi opererer på, har matchende værdier

Ydre join operationer kan imidlertid sammenføje de to originale relationer til en resulterende relation, hvor

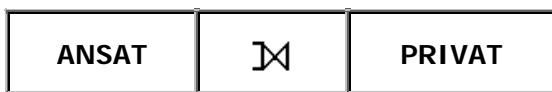
ikke matchende tupler medtages.

Der findes 3 typer af ydre join operationer:

1. Venstre ydre join (Left Outer Join) medtager alle tupler fra relationen til venstre for konditionsudtrykket, men kun de tupler, der matcher iflg. konditionen fra relationen til højre for konditionsudtrykket.
2. Højre ydre join (Right outer Join) medtager alle tupler fra relationen til højre for konditionsudtrykket, men kun de tupler, der matcher iflg. konditionen fra relationen til venstre for konditionsudtrykket
3. Fuld ydre join (Full Outer Join) medtager alle tupler fra begge relationer

Eksempler:

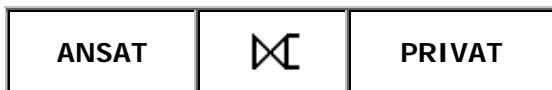
Venstre ydre join



Resultat:

Navn	Lokale	Afdeling	Gage	Telefon	Firmabil	Gratiale
Smith	400	EDB	24000	6612 1414	Null	18000
Johansen	220	Regnskab	32000	6611 1337	XA24.547	35000
Green	160	Regnskab	21000	Null	Null	Null
Brun	420	EDB	41000	6579 4453	TA54.431	35000
Smith	500	Finans	37000	6224 2423	OE65.561	25000

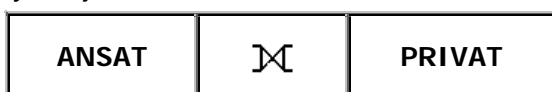
Højre ydre join



Resultat:

Navn	Lokale	Afdeling	Gage	Telefon	Firmabil	Gratiale
Smith	400	EDB	24000	6612 1414	Null	18000
Johansen	220	Regnskab	32000	6611 1337	XA24.547	35000
Brun	420	EDB	41000	6579 4453	TA54.431	35000
Smith	500	Finans	37000	6224 2423	OE65.561	25000
Boss	Null	Null	Null	Hemmeligt	KK18.111	150000

Fuld ydre join



Resultat:

Navn	Lokale	Afdeling	Gage	Telefon	Firmabil	Gratiale
Smith	400	EDB	24000	6612 1414	Null	18000
Johansen	220	Regnskab	32000	6611 1337	XA24.547	35000
Green	160	Regnskab	21000	Null	Null	Null
Brun	420	EDB	41000	6579 4453	TA54.431	35000
Smith	500	Finans	37000	6224 2423	OE65.561	25000
Boss	Null	Null	Null	Hemmeligt	KK18.111	150000

Ydre Union (Outer Union)

Ydre Union operationen anvendes til sammenlægning af partielt unionkompatible relationer.

Anvendt symbol: \cup_*

Eksempel: Vi har følgende 2 relationer:

1. KOSTPLAN

Ugedag	Middagsret
Mandag	Stegt Flæsk
Tirsdag	Dansk Bøf
Onsdag	Kogt Torsk
Torsdag	Frikadeller
Fredag	Spaghetti
Lørdag	Dyreryg
Søndag	Kylling

2. LIVRET

Navn	Middagsret
Hr. Hansen	Dansk Bøf
Hr. Nielsen	Dansk Bøf
Frk. Brun	Kylling
Frk. Sort	Dyreryg
Hr. Carlsen	Frikadeller
Frk. Bramsen	Spaghetti
Lille Ida	Røde Pølser

Opgave 14:Udfør en outer union af de to relationer: KOSTPLAN \cup^* LIVRET

Resultat:

Middagsret	Navn	Ugedag
Stegt Flæsk	Null	Mandag
Dansk Bøf	Hr. Hansen	Tirsdag
Dansk Bøf	Hr. Nielsen	Tirsdag
Kogt Torsk	Null	Onsdag
Frikadeller	Hr. Carlsen	Torsdag
Røde Pølser	Lille Ida	Null
Spaghetti	Frk. Bramsen	Fredag
Dyreryg	Frk. Sort	Lørdag
Kylling	Frk. Brun	Søndag