Cheatography

DatabaseSystems Cheat Sheet

by kmuhle via cheatography.com/210901/cs/45584/

Relational Algebra

Relational Algebra	English	Explanation	SQL	Example
$\pi_L(R)$	**Projection** of R on L	Retrieves only column L from table R, removing duplicates.	`SELECT DISTINCT L FROM R;`	If R has columns (ID, Name, Age) and data {(1, Alice, 25), (2, Bob, 30)}, then $\pi_{Name}(R)$ returns {Alice, Bob}.
$\sigma_C(R)$	**Selection** (Filter) of R by condition C	Retrieves rows that satisfy condition C.	`SELECT DISTINCT * FROM R WHERE C;`	If R has {(1, Alice, 25), (2, Bob, 30)} and C is Age > 25, then $\sigma_{age>25}(R)$ returns {(2, Bob, 30)}.
$\rho_S(R)$	**Renaming** R as S	Renames relation R to S for easier reference.	`R AS S`	If R contains {(1, Alice), (2, Bob)}, then $\rho_T(R)$ treats it as T with the same data.

$R \bowtie S$	**Natural Join** of R and S	Joins R and S on common attributes.	`R NATURAL JOIN S`
			(or, as SELECT)
			`SELECT DISTINCT * from R NATURAL JOIN S;`
				If R = {(1, Alice), (2, Bob)} and S = {(1, NYC), (2, LA)}, $R \bowtie S$ = {(1, Alice, NYC), (2, Bob, LA)}.

$R \bowtie_\theta S$	**Theta Join** of R and S on condition θ	Joins R and S where θ holds.	`R INNER JOIN S`
			`SELECT DISTINCT * from R INNER JOIN S ON θ;`
			`SELECT DISTINCT * FROM R, S WHERE θ`
				If R has {(1, Alice), (2, Bob)} and S has {(1, NYC), (3, LA)}, then with θ: R.ID = S.ID, the result is {(1, Alice, NYC)}.

$R \otimes S$	**Cartesian Product** of R and S	Pairs every row of R with every row of S.	`SELECT DISTINCT * FROM R, S;`	If R = {(1, Alice)} and S = {(NYC), (LA)}, then $R \otimes S$ = {(1, Alice, NYC), (1, Alice, LA)}.

Operations

Operation	Symbol	Meaning	Example
Union	R ∪ S	Rows in either R or S	If R = {A, B, C} and S = {B, C, D}, then R ∪ S = {A, B, C, D}.
Inters-ection	R ∩ S	Rows in both R and S	If R = {A, B, C} and S = {B, C, D}, then R ∩ S = {B, C}.
Difference	R - S	Rows in R that are not in S	If R = {A, B, C} and S = {B, C, D}, then R - S = {A}.

Common SQL Types

Data Type	Description	Example
`VARCHAR(n)`	A text string.	'hello'
`INTEGER`	A whole number.	42
`FLOAT`	A floating point number.	9.7
`NUMERIC(m,n)`	A number with m digits, with n after the decimal point.	190.14 (E.g. mondey w/ NUMERIC(3,2))
`DATE`	A date in the calendar.	'2020-09-01'
`TIME`	A time of day.	'11:25:00'
`TIMESTAMP`	Both a date and a time.	'2020-09-01T 11:25:00'
`CHAR(n)`	Fixed-length string of length n.	'02155' (E.g. zip codes w/ CHAR(5))

Closures

The *closure* of a set of attributes X - denoted as X^+ - is the set of attributes determined by X according to a given set of FDs F

The Closure Algorithm:

- Start by setting X^+ to X .
- Repeat:
  - Choose an FD L -> R where $L \subseteq X^+$.
  - Add R to X^+.
  - Record L -> R as having been used.
- Until there are no usable FDs left unused.