Explain closure of attributes in DBMS

Closure of an attribute x is the set of all attributes that are functional dependencies on X with respect to F. It is denoted by X^+ which means what X can determine.

Algorithm

Let's see the algorithm to compute X^+

- Step 1 − $X^+ = X$
- Step 2 − repeat until X^+ does not change
    - For each FD Y->Z in F
        - If $Y \subseteq X^+$ then $X^+ = X^+ \cup Z$

Example 1

Consider a relation R(A,B,C,D,E,F)

F: E->A, E->D, A->C, A->D, AE->F, AG->K.

Find the closure of E or E^+

Solution

The closure of E or E+ is as follows −

```
E+ = E
  =EA       {for E->A add A}
  =EAD      {for E->D add D}
  =EADC     {for A->C add C}
  =EADC     {for A->D D already added}
  =EADCF    {for AE->F add F}
  =EADCF    {for AG->K don't add k AG ⊄ D+)
```

Example 2

Let the relation R(A,B,C,D,E,F)

F: B->C, BC->AD, D->E, CF->B. Find the closure of B.

Solution

The closure for B is as follows −

B^+ = {B,C,A,D,E}

Closure is used to find the candidate keys of R and compute F^+

Candidate key of R: X is a candidate keys of R if X->{R}

For example,

R(A,B,C,D,E,F) WHERE F:A->BC, B->D, C->DE, BC->F. Then, find the candidate keys of R.

Solution

A^+= {A,B,C,D,E,F}={R}=>A is a candidate key

B^+= {B,D} => B is not a candidate key

C^+= {C,D,E} => C is not a candidate key

BC^+= {B,C,D,E,F} => BC is not a candidate key

Closure of F (F+): F+ is the set of all FDs that can be inferred/ derived from F. Using Armstrong Axioms repeatedly on F, we can compute all the FDs.

Example

R(A,B,C,D,E) AND F: A->B,B->C, C->D, A->E. Find the closure of F

Solution

A^+= {A,B,C,D,E}

B^+= {B,C,D}

C^+= {C,D}

F^+= {A->A, A->B, A->C, A->D, A->E, B->B, B->C, B->D, C->C, C->D}