P1.

1. p1(X,Y,Z) :- r(X,Y,Z); s(X,Y,Z).
2. p2(X,Y,Z) :- r(X,Y,Z), s(X,Y,Z).
3. p3(X,Y,Z) :- r(X,Y,Z), not( s(X,Y,Z) ).
4. temp(X,Y,Z) :- r(X,Y,Z); s(X,Y,Z).

p4(X,Y,Z) :- temp(X,Y,Z), not( t(X,Y,Z) ).

1. temp1(X,Y,Z) :- r(X,Y,Z), not( s(X,Y,Z) ).

temp2(X,Y,Z) :- r(X,Y,Z), not( t(X,Y,Z) ).

p5(X,Y,Z) :- temp1(X,Y,Z), temp2(X,Y,Z).

1. p6(X,Y) :- r(X,Y,Z).
2. temp1(X,Y) :- r(X,Y,Z).

temp2(Y,Z) :- s(X,Y,Z).

temp3(X,Y) :- temp2(X,Y).

p7(X,Y) :- temp1(X,Y), temp3(X,Y).

P2.

1. p1(X,Y,Z,A,B,C) :- r(X,Y,Z), s(A,B,C), X = B.
2. p2(X,Y,Z,A,B,C) :- r(X,Y,Z), s(A,B,C), X < B, Y < C.
3. p3(X,Y,Z,A,B,C) :- r(X,Y,Z), s(A,B,C), (X < B; Y < C).
4. temp(X,Y,Z,A,B,C) :- r(X,Y,Z), s(A,B,C), (X < B; X > B).

p4(X,Y,Z,A,B,C) :- r(X,Y,Z), s(A,B,C), not( temp(X,Y,Z,A,B,C) ).

P3.

1. p :=
2. p :=
3. p :=

P4.

CREATE TABLE parent (

NameA varchar(40) not null,

NameB varchar(40) not null

);

CREATE VIEW Ancestor AS (

WITH RECURSIVE Ancestor AS (

( SELECT NameA, NameB

FROM Parent )

UNION

( SELECT P.NameA, A.NameB

FROM Parent P, Ancestor A

WHERE P.NameB = A.NameA ))

SELECT \*

FROM Ancestor

);