Operators : which act upon operands 2 \* 3 , here 2 and 3 are operands , \* is operator

Arithmatical Operators : + - / % \*

Relational / Comparison : > < <> = >= <=

Logical Operators : AND, OR, NOT

-10 > Unary Monadic Operator : it requires 1 operand

5 – 3 , Here – is Binary or Dyadic operator , It requires 2 operands

2 \* 3

Set operators : works on the relations OR tables

UNION

UNION ALL

INTERSECT

MINUS

drop table hockey

drop table cricket

create table hockey (playername varchar(20), score int, address varchar(20))

create table cricket (playername varchar(20), runs int, state varchar(20))

insert into hockey values

('Ajay', 89, 'Delhi'),

('Deepak', 19, 'Delhi'),

('Gagan', 34, 'Calcutta'),

('Sagar', 90, 'Bombay'),

('Harish', 89, 'Bombay'),

('Sanjeev', 89, 'Delhi')

insert into cricket values

('Ajay', 890, 'Delhi'),

('Kapil', 39, 'Delhi'),

('Gagan', 34, 'Calcutta'),

('Pritpal', 90, 'Delhi'),

('Jay', 89, 'Bombay'),

('Bobby', 89, 'Delhi'),

('Jatin', 89, 'Delhi')

select \* from hockey

select \* from cricket

-- Players who are playing either of the game

select playername from hockey

UNION

select playername from cricket

-- Players who are playing eiter of the game,

--there can be duplicacy of records

select playername from hockey

UNION ALL

select playername from cricket

-- common records , players who plays both the ganes

select playername from hockey

INTERSECT

select playername from cricket

-- displays all records who are only playing hockey

select playername from hockey

EXCEPT

select playername from cricket

-- displays all records who are only playing crieckt

select playername from cricket

EXCEPT

select playername from hockey

select playername , score , address from hockey

UNION

select playername , runs , state from cricket