

Project Development - Database design

Intersect and Minus:

```
/* Intersect
    returns only distinct rows of the two queries
    Current version of MySQL is not supports for intersect operator
*/
select EmpName from t1_emp
intersect
select EmpName from t2_emp;

/* Minus
    returns the records which is not common in two queries
*/
select EmpName from t1_emp
minus
select EmpName from t2_emp;
```

Joins in MySQL:

1. When we require the data from more than single table
2. A relational database consists of multiple related tables together using of common column that is in generally we call it as "Foregin Key".
3. If you're trying to apply a join based on the same column then it refers to "self-join", It might be two table or more than two tables.

```

create table tbl_Student(
    Id int auto_increment,
    StudentName varchar(50) not null,
    EmailID varchar(50) not null,
    StudentPwd varchar(30) not null,
    PhoneNumber bigint,
    Age int not null,
    DeptId int,
    primary key(Id)
);

```

```

create table tbl_Student_Dept(
    DeptId int auto_increment,
    DeptName varchar(100) not null,
    DeptLocation varchar(100) not null,
    primary key(DeptId)
);

```

```

select * from tbl_Student;
select * from tbl_Student_Dept;

```

```

insert into tbl_Student(StudentName,EmailID,StudentPwd,PhoneNumber,Age,DeptID) values ('Anil','anil@dhyanaahita.org','12345',9988776655,29,1),
('Harsha','harsha@dhyanaahita.org','12345',8877665544,30,2),('Ajay','ajay@dhyanaahita.org','12345',8877661122,20,1),
('Dinesh','dinesh@dhyanaahita.org','12344',8877662233,23,3);

```

```

insert into tbl_Student_Dept(DeptName,DeptLocation) values('CSE','Hyd'),('ECE','Hyd'),('MECH','HYD'),('AERO','HYD');

```

```

/*
    Select List : StudentName,PhoneNumber,Age,DeptName,DeptLocation
*/

```

```

select StudentName,PhoneNumber,Age,DeptName,DeptLocation from tbl_Student,tbl_Student_Dept
where tbl_Student.DeptID = tbl_Student_Dept.DeptID;

```

```

/* Replace by using of Alias Names
    Alias Names : Alternate names for the columns as well as tables
*/

```

```

select a.StudentName,a.PhoneNumber,a.Age,b.DeptName,b.DeptLocation from tbl_Student a,tbl_Student_Dept b
where a.DeptId = b.deptId;

```

```

/*
    Select List : StudetName,PhoneNumber,Age,DeptName,DeptLocation
    Condition : Age should be greater than 25
*/

```

```

select a.StudentName,a.PhoneNumber,a.Age,b.DeptName,b.DeptLocation from tbl_Student a,tbl_Student_Dept b
where a.DeptId = b.DeptId and a.Age > 25;

```

```

/*
    Select List : StudetName,PhoneNumber,Age,DeptName,DeptLocation
    Condition : Age should be greater than 20 and Belongs to DeptId as 1
*/

```

```

select a.StudentName,a.PhoneNumber,a.Age,b.DeptName,b.DeptLocation from tbl_Student a,tbl_Student_Dept b
where a.DeptId = b.DeptId and a.Age > 25 and b.DeptId = 1;

```

```

/*
    Select List : StudentName,PhoneNumber,Age,DeptName,DeptLocation
    Condition : Name should be starts with a and DeptId = 1
*/

```

```

select a.StudentName,a.PhoneNumber,a.Age,b.DeptName,b.DeptLocation from tbl_Student a,tbl_Student_Dept b
where a.DeptId = b.DeptId and a.StudentName like 'a%' AND a.DeptId in (1);

```

```

/*
    Select List : StudentName,PhoneNumber,Age,DeptName,DeptLocation
    Condition : Name should be starts with a and DeptId should be anything from 1 to 4
*/

```

```

select a.StudentName,a.PhoneNumber,a.Age,b.DeptName,b.DeptLocation from tbl_Student a,tbl_Student_Dept b
where a.DeptId = b.DeptId and a.StudentName like 'a%' AND a.DeptId in (1,2,3,4);

```

```

/*
    Customer Table - CustomerNumber
    Orders Table - CustomerNumber
    Orderdetails Table - orderNumber
    orders Table - orderNumber
    Payments Table - customerNumber
*/

```

```

select a.customerName,a.phone,a.creditLimit,b.shippedDate,b.status from customers a,orders b
where a.CustomerNumber = b.CustomerNumber;

```

```

select a.customerName,a.phone,a.creditLimit,b.shippedDate,b.status from customers a,orders b
where a.CustomerNumber = b.CustomerNumber and b.status = 'Cancelled';

```

1. Left Join
2. Right Join
3. Cross Join