

1. Tuliskan perintah dengan **Inner Join**, untuk menampilkan ID student, Nama belakang student, Nama belakang Prof, dimana grade diatas 84.

Jawab:

select st.STUD\_ID, st.STUD\_LNAME, pr.PROF\_LNAME

from STUDENT st

join PROF pr on st.STUD\_ID = pr.PROF\_ID

where grade >84 group by STUD\_ID

1. Tuliskan perintah dengan **Outer Join**, untuk menampilkan:
2. Nama Student yang belum dan sudah pernah enroll class

select STUD\_FNAME, STUD\_LNAME, CLASS\_ID

from STUDENT st full outer join CLASS cl

on st.STUD\_ID = cl.CLASS\_ID

1. Nama Prof yang belum dan sudah pernah mengajar class

select PROF\_FNAME, PROF\_LNAME, CLASS\_ID

from PROF pr full outer join CLASS cl

on pr.PROF\_ID = cl.CLASS\_ID

**DML Sub Query dan Set Operator**. Tuliskan perintah untuk:

1. Menampilkan ID Prof yang pernah mengajar minimal 1 class, dengan menggunakan:
   * 1. Join

select pr.PROF\_ID, cl.CLASS\_ID

from PROF pr join CLASS cl

on pr.PROF\_ID = cl.CLASS\_ID

where CLASS\_ID = not null

* + 1. Sub Query IN

select PROF\_ID

from PROF

where CLASS\_ID in(

select distinct CLASS\_ID

from class)

* + 1. Sub Query EXISTS

select PROF\_ID

from PROF

where exists (select CLASS\_ID from CLASS

where pr.PROF\_ID = cl.CLASS\_ID and CLASS\_ID = not null)

* + 1. INTERSECT Set Operator

select PROF\_ID

from PROF

intersect

select PROF\_ID

from PROF pr, CLASS cl

where pr.PROF\_ID = cl.CLASS\_ID, and CLASS\_ID = not null

1. Menampilkan ID student yang belum pernah enroll class, dengan menggunakan:
2. Sub Query NOT IN

select STUD\_ID

from STUDENT

where CLASS\_ID not in(

select distinct CLASS\_ID

from CLASS)

1. Sub Query NOT EXISTS

select STUD\_ID

from STUDENT

where not exists (select CLASS\_ID from CLASS

where st.STUD\_ID = cl.CLASS\_ID and CLASS\_ID = not null)

1. EXCEPT Set Operator

Select STUD\_ID

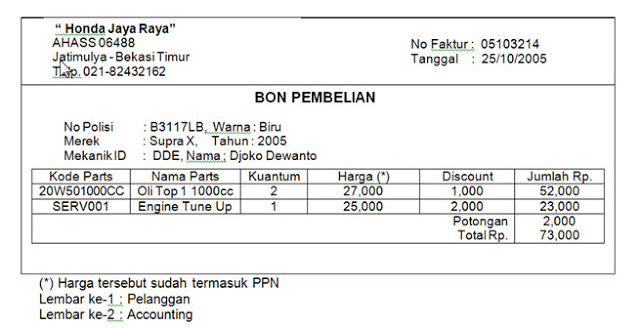
from STUDENT

except

select STUD\_ID

from STUDENT st, CLASS cl

where st.STUD \_ID = cl.CLASS\_ID and CLASS\_ID = not null



1. Terapkan **Normalisasi** **1-NF, 2-NF, dan 3-NF** dengan menentukan repeating groups, partial dependency dan transitive dependency untuk form berikut, dimana Candidate Key nya adalah No.Faktur dan kode parts.

UNF: **No\_Faktur**, Tanggal, No\_Polisi, Warna, Merek, Tahun, Mekanik\_ID, Nama, Potongan, {**kode\_parts**, nama\_parts, kuantum, harga, discount}

1NF:

HeaderPembelian: **No\_Faktur**, Tanggal, No\_Polisi, Warna, Merek, Tahun, Mekanik\_ID, Nama, Potongan

DetailPembelian: **No\_Faktur**, **kode\_parts**, nama\_parts, kuantum, harga, discount

2NF:

HeaderPembelian: **No\_Faktur**, Tanggal, No\_Polisi, Warna, Merek, Tahun, Mekanik\_ID, Nama, Potongan

DetailPembelian: **No\_Faktur**, **kode\_parts**, kuantum, discount

Part: **kode\_parts**, nama\_parts, harga

3NF:

HeaderPembelian: **No\_Faktur**, Tanggal, No\_Polisi, Mekanik\_ID, Potongan

Kendaraan: **No\_Polisi**, Warna, Merek, Tahun

Mekanik: **Mekanik\_ID**, Nama

DetailPembelian: **No\_Faktur**, **kode\_parts**, kuantum, discount

Part: **kode\_parts**, nama\_parts, harga