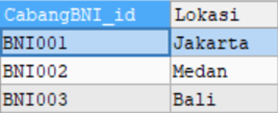
|  |
| --- |
| **LAPORAN PRAKTIKUM PENGENALAN BASIS DATA**  **REVIEW UAS** |
|  |
| **Agus Pranata Marpaung**  **13323033**  **DIII TEKNOLOGI KOMPUTER** |
| **INSTITUT TEKNOLOGI DEL**  **FAKULTAS VOKASI** |

1. Cross join merupakan join yang merupakan himpunan bagian dari kombinasi dua atau lebih tabel yang dikenal sebagai
2. **Cartesian Product**
3. UNION product
4. Product
5. STRUCTURE Product
6. MERGE Product

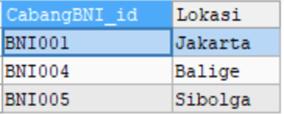
**Pembahasan:**

1. Diketahui 2 buah tabel dengan nama t\_ CabangBNI dan t\_ CabangBNIBaru

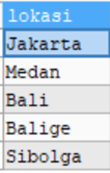
t\_ CabangBNI



t\_CabangBNIBaru



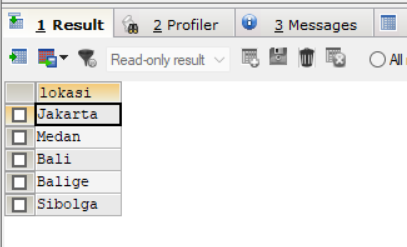
Sintaks yang diperlukan untuk menggabungkan semua lokasi dari kedua tabel sehingga seperti di bawah ini, adalah:



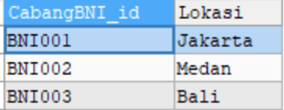
1. SELECT lokasi FROM CabangBNI UNI SELECT lokasi FROM CabangBNIBaru
2. SELECT lokasi FROM CabangBNI GABUNG SELECT lokasi FROM CabangBNIBaru
3. SELECT lokasi FROM CabangBNI PLUS SELECT lokasi FROM CabangBNIBaru
4. **SELECT lokasi FROM CabangBNI UNION SELECT lokasi FROM CabangBNIBaru**
5. SELECT lokasi FROM CabangBNI AND SELECT lokasi FROM CabangBNIBaru

**Penjelasan:**

1. **SELECT lokasi FROM CabangBNI UNION SELECT lokasi FROM CabangBNIBaru**:



1. Perhatikan tabel berikut ini: (t\_cabangbni)



Jika kita ingin memindahkan CabangBNI\_id dengan no BNI001 dan lokasi jakarta (data pada row 1) ke sebuah tabel dengan nama CabangBNIBaru,

t\_ CabangBNIBaru

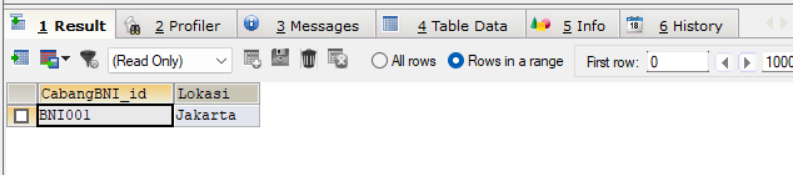


maka sintaks yang kita perlukan adalah:

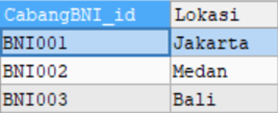
1. INSERT CabangBNI SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');
2. ADD CabangBNI SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');
3. ENTRY INTO CabangBNI SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');
4. **INSERT INTO CabangBNIBaru SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');**
5. INSERT INTO CabangBNI SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');

**Penjelasan:**

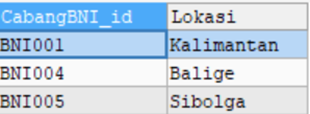
1. **INSERT INTO CabangBNIBaru SELECT \* FROM CabangBNI WHERE CabangBNI\_id IN ('BNI001','Jakarta');** :



1. Dari tabel t\_ CabangBNI berikut ini,



Akan diubah Lokasi pada CabangBNI\_id = BNI001 menjadi Kalimantan.

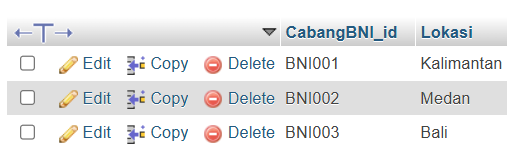


Maka sintaks yang diperlukan adalah:

1. **UPDATE CabangBNI SET Lokasi = 'Kalimantan' WHERE CabangBNI\_id='BNI001’;**
2. UPDATE CabangBNIBaru SET Lokasi = 'Kalimantan' WHERE CabangBNI\_id='BNI001';
3. ENTRY CabangBNIBaru SET Lokasi = 'Kalimantan' WHERE CabangBNI\_id='BNI001';
4. UPDATE CabangBNIBaru ADD Lokasi = 'Kalimantan' WHERE CabangBNI\_id='BNI001';
5. UPDATE CabangBNIBaru SET Lokasi = 'Kalimantan' WHERE CabangBNI\_id=’BNI001’;

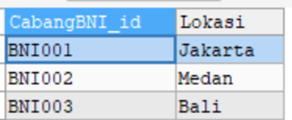
**Penjelasan:**

1. **UPDATE CabangBNI SET Lokasi = 'Kalimantan' WHERE CabangBNI\_id ='BNI001' :**

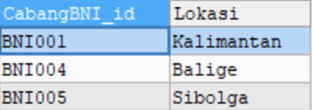
****

1. Perhatikan tabel berikut ini!

**t\_CabangBNI**



**t\_CabangBNIBaru**



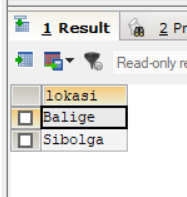
Sintaks yang diperlukan untuk membuat hasil berikut adalah:



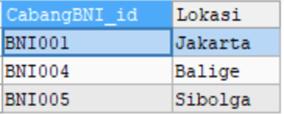
1. SELECT lokasi FROM CabangBNIBaru UNION (SELECT lokasi FROM CabangBNI)
2. SELECT lokasi FROM CabangBNIBaru NO (SELECT lokasi FROM CabangBNI)
3. SELECT lokasi FROM CabangBNIBaru EXCEPT (ADD lokasi FROM CabangBNI)
4. **SELECT lokasi FROM CabangBNIBaru EXCEPT (SELECT lokasi FROM CabangBNI)**
5. SELECT lokasi FROM CabangBNI EXCEPT (SELECT lokasi FROM CabangBNIBaru)

**Penjelasan:**

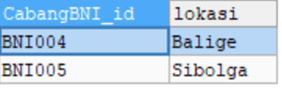
1. **SELECT lokasi FROM CabangBNIBaru EXCEPT (SELECT lokasi FROM CabangBNI):**



1. Dari tabel t\_ CabangBNIBaru berikut ini,



Akan dilakukan seleksi data lokasi dengan kriteria cabangBNI lebih besar dari BNI001 dengan hasil berikut ini:



maka sintaks kueri yang benar untuk menghasilkan hasil di atas adalah:

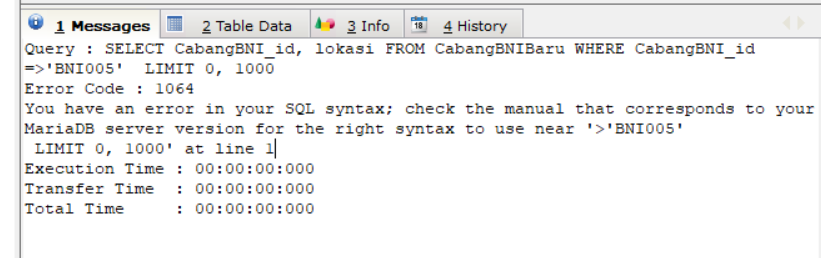
1. SELECT CabangBNI\_id, lokasi FROM CabangBNIBaru WHERE CabangBNI\_id >'BNI001'
2. SELECT CabangBNI\_id FROM CabangBNIBaru WHERE CabangBNI\_id >=’BNI004’
3. SELECT CabangBNI\_id FROM CabangBNIBaru WHERE CabangBNI\_id ='BNI004'
4. SELECT CabangBNI\_id, lokasi FROM CabangBNIBaru WHERE CabangBNI\_id =>'BNI005'
5. SELECT CabangBNI\_id, lokasi FROM CabangBNIBaru WHERE CabangBNI\_id >='BNI004'

Pernyataan yang benar adalah:

1. i dan iii
2. ii dan iv
3. i dan ii
4. **i dan v**
5. ii dan iii

**Penjelasan:**

1. **SELECT CabangBNI\_id, lokasi FROM CabangBNIBaru WHERE CabangBNI\_id =>'BNI005'**:



1. Jika terdapat adanya ketergantungan antara non key atribut, maka tabelnya harus dipecah merupakan syarat normalisasi?
2. BCNF
3. 1 NF
4. **3 NF**
5. 2 NF
6. UNF

**Penjelasan:**

Untuk memenuhi syarat normalisasi pada bentuk ketiga normal (3NF), tabel harus dipecah karena ketergantungan antara atribut non-kunci. Syarat utama 3NF adalah bahwa, selain memenuhi syarat 1NF dan 2NF, tidak ada ketergantungan transitif antara atribut non-kunci dan atribut kunci utama.

1. Normalisasi basisdata penting dilakukan untuk:
2. Menghindari kesalahan penghapusan data yang masih dibutuhkan ketika data lain dihapus (delete anomalies)
3. Meminimalisasi ketidakkonsistenan data ketika dilakukan modifikasi pada basisdata (update anomalies)
4. Memungkinkan penyimpanan data pada disk komputer.
5. Memanfaatkan penggunaan fitur sistem manajemen basisdata.
6. ii dan iv
7. i dan iii
8. ii dan iii
9. i dan iv
10. **i dan ii**

**Penjelasan:**

Memastikan integritas data dan mengurangi redundansi adalah dua tujuan utama normalisasi.

1. Perhatikan Gambar 1. Relasi antara student dan advisor berikut tidak normal karena:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | dosen | ruangan dosen | laboratorium 1 | laboratorium 2 | laboratorium 3 | departemen |
| 12 | Sari | 1114 | 101-01 | 101-03 | 101-05 | 51 |
| 13 | Frengki | 1113 | 101-02 | 101-04 | 101-06 | 52 |

Gambar 1. Relasi StudentAdvisor

1. Tidak ada atribut yang saling bergantung satu dengan yang lain.
2. **Atribut dosen tergantung kepada ruangan dosen**
3. Susah untuk menyimpan data dengan tipe data dari atribut yang tidak seragam.
4. Atribut Laboratorium 1, Laboratorium 2, dan Laboratorium 3 adalah kelompok atribut yang mengalami perulangan.
5. Mahasiswa dengan nomor 12 mengambil 3 kelas yang berbeda.

**Penjelasan:**

Karena setiap dosen memiliki ruangan tertentu yang ditugaskan untuk mereka, dan ruangan ini tidak berubah dari satu dosen ke dosen lainnya.

1. Sebuah relasi memenuhi 2NF apabila:
2. Relasi sudah memenuhi 1NF;
3. Atribut yang bukan kunci tergantung kepada atribut kunci;
4. Atribut yang bukan kunci tidak saling tergantung satu dengan yang lain;
5. Apabila memiliki composite key, tidak boleh ada atribut bukan kunci yang hanya tergantung sebagian saja kepada atribut pembentuk composite key.

Pernyataan yang benar adalah:

1. ii dan iii
2. i dan iii
3. ii dan iv
4. **i dan iv**
5. i dan ii

**Penjelasan:**

1. Diberikan relasi VendorOrder dengan atribut dan data masing-masing sebagai berikut.

**Vendor Order** (vendorno, orderno, vendorname, qty\_supplied, price/unit)



Gambar 2 Relasi VendorOrder

Relasi ini tidak memenuhi 2NF karena:

1. VendorOrder belum memenuhi 1NF.
2. VendorOrder memiliki composite key yaitu vendorno dan orderno.
3. Atribut vendorname tergantung kepada vendorno, yang merupakan salah satu bagian dari composite key.
4. Atribut qty\_supplied tergantung kepada orderno saja.
5. Atribut price/unit tidak tergantung kepada qty\_supplied.

Pernyataan yang benar adalah:

1. i dan ii
2. i dan iii
3. i, ii dan iii
4. hanya v
5. **iii dan iv**

**Penjelasan:**

iii) Atribut vendorname tergantung kepada vendorno, yang merupakan salah satu bagian dari composite key.

iv) Atribut qty\_supplied tergantung kepada orderno saja.

1. Sebuah relasi memenuhi 3NF apabila:
2. Relasi sudah memenuhi 2NF;
3. Atribut yang bukan kunci tidak saling tergantung satu dengan yang lainnya;
4. Atribut yang bukan kunci tidak bergantung pada sebagian saja atribut pada composite key;
5. Tidak memiliki composite key.

Pernyataan yang benar adalah:

1. i dan iii
2. **i dan ii**
3. ii dan iv
4. i dan iv
5. ii dan iii

**Penjelasan:**

1. Relasi dengan skema berikut

**Mahasiswa** (NIM, Nama, KodeFakultas, NamaFakultas, LamaStudi, Alamat)

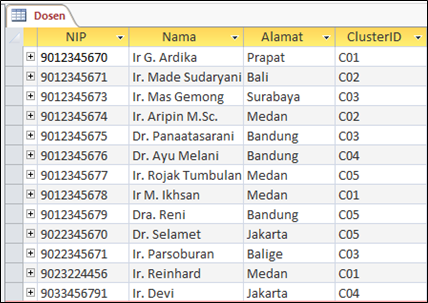
tidak memenuhi 3NF karena ...

1. Relasi tidak memiliki composite key.
2. Atribut Alamat bergantung kepada atribut kunci NIM.
3. **Atribut yang bukan kunci yaitu NamaFakultas bergantung kepada atribut KodeFakultas.**
4. Relasi memiliki lebih dari tiga (3) atribut yang bukan kunci.
5. Relasi tidak memenuhi 2NF.

**Penjelasan:**

Karena dalam 3NF, tidak boleh ada dependensi transitif; dalam hal ini, NamaFakultas (atribut non-kunci) bergantung pada KodeFakultas (atribut non-kunci), yang merupakan dependensi transitif.

1. Perhatikan tabel Dosen pada Gambar 6 berikut ini.



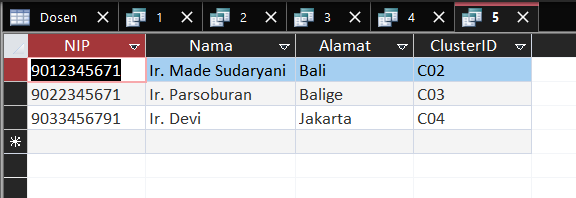
Gambar 6 Tabel Dosen

Sintaks SQL yang akan menghasilkan semua data dosen yang memiliki NIP diakhiri dengan angka 1 adalah:

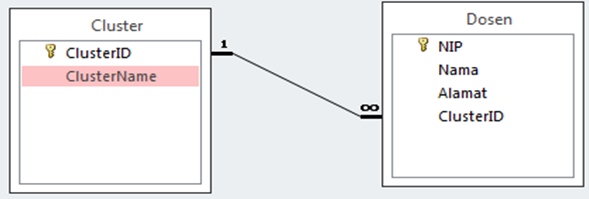
1. SELECT \* FROM Dosen WHERE NIP Like 0
2. SELECT \* FROM Dosen WHERE NIP Like \*????????0
3. SELECT \* FROM Dosen WHERE NIP = ‘0’
4. SELECT \* FROM Dosen WHERE NIP Like ?????????0
5. **SELECT \* FROM Dosen WHERE NIP Like ‘\*1’**

**Penjelasan:**

1. **SELECT \* FROM Dosen WHERE NIP Like ‘\*1’**



1. Jika tabel Dosen pada soal 14 memiliki relasi dengan tabel Cluster seperti pada Gambar 7 berikut.



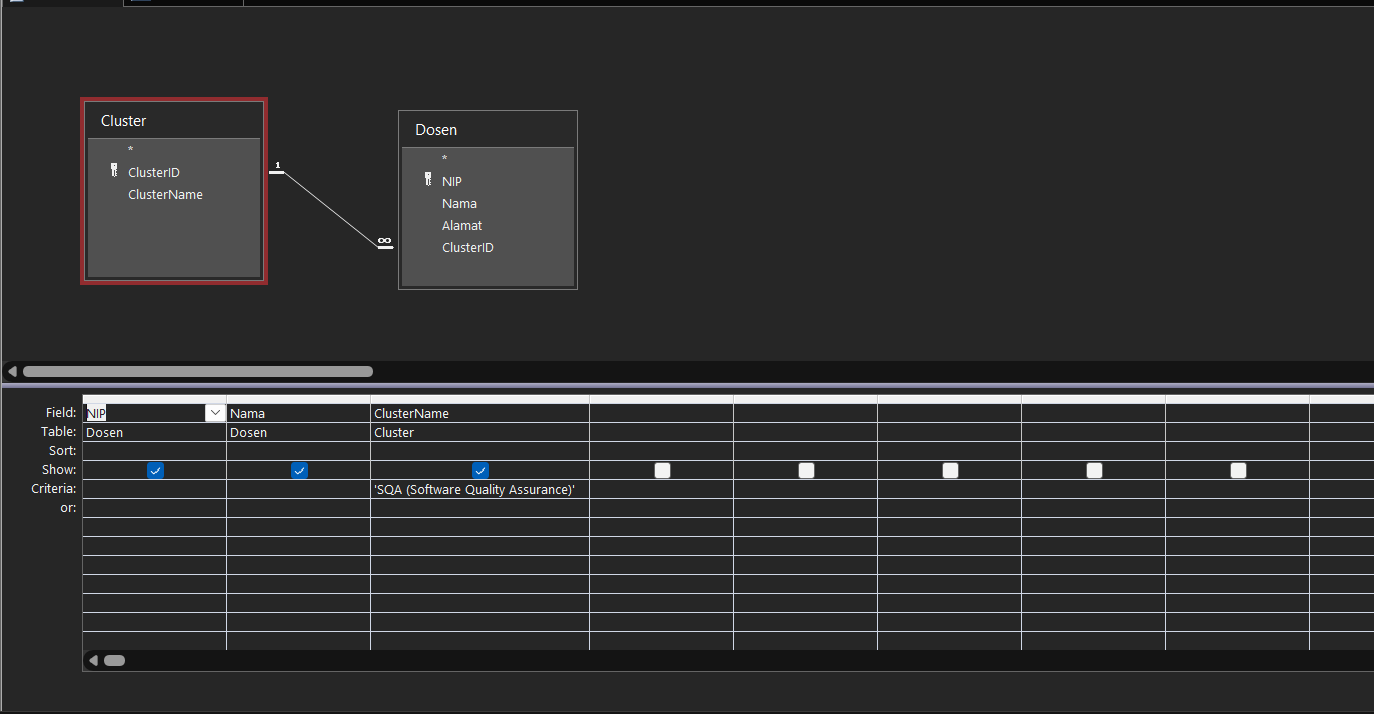
Gambar 7 Relasi Tabel Cluster - Dosen

Maka sintaks SQL berikut yang akan menghasilkan data dosen dengan atribut NIP, Nama, dan ClusterName yang berada pada ClusterName “SQA (Software Quality Assurance)” adalah:

1. SELECT NIP, Nama, ClusterName FROM Dosen INNER JOIN Cluster ON Cluster.ClusterID=Dosen.ClusterID WHERE ClusterID=”C01” and ClusterName=”SQA (Software Quality Assurance)
2. SELECT NIP, Nama, ClusterName FROM Dosen INNER JOIN Cluster ON Cluster.ClusterID=Dosen.ClusterID
3. SELECT NIP, Nama, ClusterName FROM Dosen INNER JOIN Cluster WHERE ClusterName=”SQA (Software Quality Assurance)”
4. **SELECT NIP, Nama, ClusterName FROM Dosen INNER JOIN Cluster ON Cluster.ClusterID=Dosen.ClusterID WHERE ClusterName=” SQA (Software Quality Assurance)”**
5. SELECT NIP, Nama, ClusterName From Dosen INNER JOIN WHERE ClusterID=” SQA (Software Quality Assurance)”

**Penjelasan:**

1. **SELECT NIP, Nama, ClusterName FROM Dosen INNER JOIN Cluster ON Cluster.ClusterID=Dosen.ClusterID WHERE ClusterName=” SQA (Software Quality Assurance)”**

****

Pada gambar tersebut, di bagian field Terdapat ClusterName yang mana table Cluster dan di dalamnya terdapat data SQA (Software Quality Assurance).