**Perancangan Basis Data**

**Pengertian Basis Data**

Basis data adalah penyimpanan kumpulan informasi secara sistematik dalam sebuah komputer sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil query basis data disebut sistem manajemen basis data (*database management system, DBMS*).

DBMS memiliki karakteristik, sebagai berikut :

* Software program
* Supplements operating sistem
* Manages data
* Queries data and generates reports
* Data security

Sedangkan sistem adalah sebuah tatanan yang terdiri atas sejumlah komponen fungsional yang saling berhubungan dan secara Bersama – sama bertujuan untuk memenuhi suatu proses pekerjaan. Sehingga bisa dikatakan bahwa sistem basis data adalah sistem yang terdiri atas kumpulan file – file yang saling berhubungan dan dikelola oleh program (DBMS) yang memungkinkan beberapa pemakai dan atau program lain yang memiliki otoritas untuk mengakses dan manipulasi data.

DBMS (*database Management System)* memiliki beberapa fungsi, diantaranya :

1. *data storage, retrieval, and update*

sebuah DBMS harus melengkapi / menyediakan pengguna dengan kemampuan penyimpanan, penelusuran kembali, dan mengubah data dalam database

1. *a user accessible catalog*

sebuah DBMS harus menyediakan catalog yang mendeskripsikan lokasi penyimpanan data dan dapat diakses oleh pengguna

1. *transaction support*

DBMS harus menyediakan sebuah mekanisme yang akan menjamin setiap dari semua kegiatan mengubah yang berhubungan dengan transaksi maupun tidak

1. *concurrency control service*

DBMS harus menyediakan sebuah mekanisme untuk menjamin bahwa database dapat diubah dengan benar ketika beberapa pengguna mengubah database pada waktu yang bersamaan

1. *recovery services*

DBMS harus menyediakan sebuah mekanisme untuk memperbaiki basis data yang rusak karena sesuatu kejadian

1. *authorization services*

DBMS harus menyediakan sebuah mekanisme untuk menjamin bahwa hanya pengguna yang diberi otoritas yang dapat mengakses basis data

1. *support for data communication*

DBMS harus mampu berintegrasi dengan software komunikasi

1. *integrity services*

DBMS harus menyediakan sebuah cara untuk menjamin bahwa data dalam basis data dan perubahan data, keduanya mengikuti aturan – aturan yang tepat

1. *services to promote data independence*

DBMS harus meliputi fasilitas – fasilitas yang mendukung program – program independsi dari struktur basis data actual

1. *utility services*

DBMS harus menyediakan satu set fasilitas pelayanan

Kelebihan pemakaian DBMS adalah, sebagai berikut :

* Data berdiri sendiri (*data independence*)
* Pengaksesan data efisien (*efficient data access*)
* Integritas data dan keamanan terjamin (*data integrity and security*)
* Administrasi data (*data administration*)
* Dapat diakses bersamaan (*concurrent access*)
* Recovery saat terjadi kegagalan (*crash recovery*)
* Mengurangi waktu pembangunan aplikasi (*reduced application development time*)

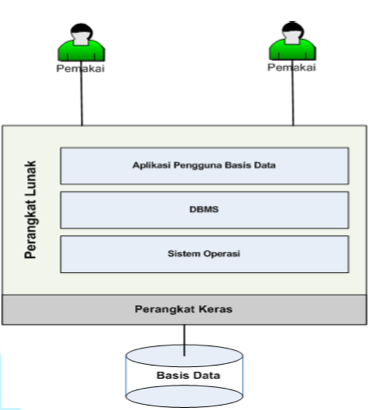
Kekurangan pemakaian DBMS adalah sebagai berikut :

* Merupakan software yang kompleks sehingga user harus mengerti fungsi – fungsi yang ada untuk mendapatkan hasil rancangan yang baik
* Membutuhkan tempat penyimpanan dan memory yang cukup besar untuk instalasinya
* Biaya DBMS yang tergantung pada tipe DBMS yang digunakan
* Biaya tambahan terhadap kebutuhan perangkat keras
* Biaya tambahan lain seperti pelatihan dan spesialisasi untuk menjalankan sistem yang baru
* Performa
* Dampak kegagalan lebih besar

**Komponen sistem basis data**

Komponen – komponen pada sebuah sistem basis data, antara lain :

* Perangkat keras
* Sistem operasi
* Basis data
* DBMS (*database management system*)
* Pemakai
* Aplikasi lain



Gambar 1 komponen DBMS

**Abstraksi Data**

Tujuan utama dari sistem basis data adalah untuk menyediakan fasilitas untuk view data secara abstrak bagi penggunanya. Namun bagaimana sistem menyimpan dan mengelola data. Abstraksi data merupakan level dalam bagaimana melihat data dalam sebuah sistem basis data.

Level abstraksi data dibagi menjadi 3, yaitu :

1. **Level fisik**

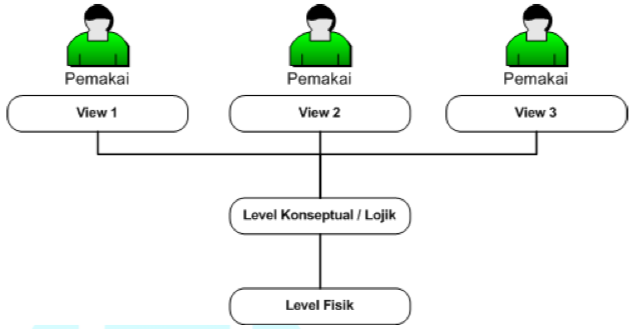
Level fisik merupakan level terendah pada abstraksi data yang menunjukkan bagaimana sesungguhnya data disimpan. Pada level pemakai melihat data sebagai gabungan dari struktur dan datanya sendiri.

1. **Level Logic**

Level logic merupakan level berikutnya pada abstraksi data, menggambarkan data apa yang disimpan pada basis data dan hubungan apa saja yang ada diantara data

1. **Level view**

Level view merupakan level tertinggi dari abstraksi data yang hanya menunjukkan sebagian dari basis data. Banyak user dalam basis data tidak akan terlibat dengan semua data atau informasi yang ada atau yang disimpan. Para user umumnya hanya membutuhkan sebagian data atau informasi dalam basis data yang berada di antramuka user.



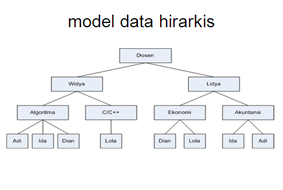
Gambar 2 level abstraksi data

**Model basis data**

Terdapat beberapa model basis data yang digunakan, diantaranya :

1. **Model *Hierarchical***

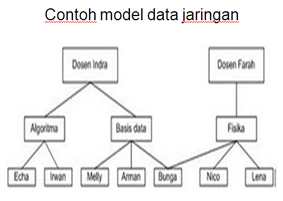
Model *hierarchical* memiliki bentuk struktur pohon dimana *field* hanya memiliki satu buah induk (*parent*), masing – masing parent memiliki banyak *child* (anak). Model ini memiliki kecepatan yang baik.



Gambar 3 Model Hierarchical

1. **Model *Network***

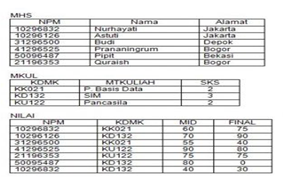
Model *network* merupakan *relationship* dibuat menggunakan *linked list (pointer).* Berbeda dengan model *hierarchical* satu anak dapat memiliki beberapa induk. Model ini memiliki fleksibilitas yang tinggi.



Gambar 4 model *network*

1. **Model *Relational***

model *relational* adalah model ini direpresentasikan dalam tabel dua dimensi, tabel – tabel tersebut memiliki hubungan yang disebut dengan relasi. Model ini memiliki fleksibilitas dan kecepatan yang tinggi.



Gambar 5 model *relational*

**Pengertian Perancangan Basis Data**

Perancangan basis data adalah proses untuk menentukan isi dan pengaturan data yang dibutuhkan untuk mendukung berbagai rancangan sistem

**Tujuan Perancangan Basis Data**

Pada perancangan basis data terdapat beberapa tujuan, diantaranya adalah :

1. Untuk memenuhi informasi yang berisikan kebutuhan – kebutuhan user secara khusus dan aplikasi – aplikasinya
2. Memudahkan pengertian struktur informasi
3. Mendukung kebutuhan – kebutuhan pemrosesan dan beberapa objek penampilan (*response time, processing time,* dan *storage space* ).

**Tahapan perancangan basis data**

Dalam pembuatan / perancangan basis data, terdapat 6 fase proses perancangan basis data, diantaranya :

1. Pengumpulan data dan analisis

Proses identifikasi dan analisis kebutuhan – kebutuhan data disebut dengan pengumpulan data dan Analisa. Untuk menentukan kebutuhan – kebutuhan suatu sistem basis data. Melakukan peninjauan dokumentasi yang ada. Serta melakukan analisis lingkungan operasi dan pemrosesan data dan dapat juga membuat daftar pertanyaan dalam melakukan wawancara pada proses pengumpulan data

1. Perancangan basis data secara konseptual

Perancangan basis data secara konseptual dimana proses ini membuat / merancang skema konseptual dan merancangan transaksi yang akan terjadi di dalam database / basis data

1. Pemilihan DBMS

Ada beberapa factor dalam pemilihan DBMS, diantaranya adalah :

1. Factor teknis

Contoh factor Teknis adalah :

Memilih tipe model data (hirarki, jaringan atau relational), struktur penyimpanan dan jalur pengaksesan yang didukung sistem manajemen database, interface dan programmer, dan bahasa query

1. Factor ekonomi dan politik organisasi

factor – factor ekonomi, meliputi : biaya penyediaan hardware dan software, biaya konversi pembuatan database, biaya personalia, dll

factor – factor politik organisasi.

1. Perancangan basis data secara logika (*data model mapping*)

Perancangan basis data secara logika, meliputi :

1. Pemetaan (transformasi data)

Transformasi yang tidak tergantung pada sistem, pada tahap ini transformasi tidak mempertimbangkan karakteristik yang spesifik atau hal – hal khusus yang akan dipublikasikan pada sistem manajemen database

1. Penyesuaian skema ke DBMS

Penyesuaian skema yang dihasilkan dari tahap pemetaan untuk dikonfirmasikan pada bentuk implementasi yang spesifik dari suatu model data seperti yang digunakan oleh sistem manajemen database yang terpilih.

1. Perancangan basis data secara fisik

Perancangan basis data secara fisik, meliputi :

1. ***Response time***

Waktu transaksi database selama eksekusi untuk menerima respon

1. ***Space utility***

Jumlah ruang penyimpanan yang digunakan oleh database file dan struktur jalur pengaksesannya

1. ***Transaction throughtput***

Merupakan nilai rata – rata transaksi yang dapat di proses permenit oleh sistem database dan merupakan parameter kritis dari sistem transaksi

1. Implementasi sistem basis data

**Komponen – komponen basis data antara lain :**

1. **Entitas**

Entity relationship adalah jaringan yang menggunakan susunan data yang disimpan dari sistem secara abstrak. Entity relationship model terdiri dari unsur – unsur entity dan relationship antara entity – entity tersebut.

Contoh :

1. Semua pelanggan, atau pelanggan saja dengan entitas adi, ryan, endah dan seterusnya
2. Semua mobil atau mobil apa saja dengan entitas mobil kijang, honda dan lain – lain
3. **Relasi**

Relasi adalah sekumpulan hubungan antar entitas yang memiliki arti. Relasi digambarkan dengan sebuah garis yang menghubungkan entitas – entitas yang saling berhubungan.

1. **Atribut**

Setiap entitas pasti memiliki atributa yang mendeskripsikan karakteristik dari entitas tersebut. Penentuan / pemilihan atribut – atributa yang relevan bagi sebuah entitas merupakan hal penting lainnya dalam pembentukan model data. Penetapan atribut bagi sebuag entitas umumnya memang didasarkan pada fakta yang ada. Tetapi tidak selalu seperti itu. Istilah atribut sebenarnya identic dengan pemakaian kolom data.

Contoh :

1. Entitas pelanggan memiliki atribut : kd\_pelanggan, nama\_pelanggan, alamat, dan no\_telepon
2. Entitas mahasiswa memiliki atribut : nim, nm\_mahasiswa, alamat, no\_telepon
3. **Kunci (*key*)**

Terdapat beberapa macam key yang dapat diterapkan pada suatu tabel, diantaranya :

1. ***Super key***

Super key merupakan himpunan dari satu atau lebih entitas yang dapat digunakan untuk mengidentifikasi secara unik sebuah entitas dalam entitas set.

1. ***Candidate key***

Candidate key merupakan kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah tabel secara unik. Sebuah candidate key tidak boleh berisi atribut atau kumpulan super key yang lain.

1. ***Primary key***

Primary key merupakan satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian spesifik tapi juga dapat mewakili setiap kejadian dari suatu entity

1. ***Alternate key***

Alternate key merupakan kunci candidate yang tidak dipakai sebagai primary key. Sering kali kunci alternative ini dipakai sebagai kunci pengurutan dalam pembuatan laporan.

1. ***Foreign key***

Foreign key adalah satu atribut atau satu set minimal atribut yang melengkapi satu hubungan yang menunjukkan ke induknya. Kunci tamu ditempatkan pada entity anak dan sama dengan induk kunci utama yang direlasikan.

***Entity relationship model***

*Entity relationship model* merupakan model untuk menjelaskan hubungan antar data dalam basis data berdasarkan suatu persepsi bahwa real word terdiri dari objek – objek dasar yang mempunyai hubungan atau relasi antara objek – objek.

E – R MODEL berisi ketentuan / aturan khusus yang harus dipenuhi oleh isi database. Aturan terpenting adalah MAPPING CARDINALITIES, yang memenuhi jumlah entity yang dapat dikaitkan dengan entity lainnya melalui *relationship – set.*

Symbol yang digunakan pada pembuatan E – R Model adalah,sebagai berikut :

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | Entitas |
|  | Relasi |
|  | Atribut |
|  | Menunjukkan adanya relasi |
|  | Weak entitas |
|  | Identifying Relationship |
|  | Atribut multivalue |
|  | Atribut primary key |
|  | Atribut derivative |
|  | Attribute composite |

Contoh E-R Model

Customer

Buku tabungan

memiliki

Gambar 6 contoh E-R Model

**Komponen entity relationship**

Berikut ini adalah komponen – komponen entity relationship, diantaranya adalah :

1. Entitas yaitu suatu kumpulan object atau sesuatu yang dapat dibedakan atau dapat diidentifikasikan secara unik. Dan kumpulan entitas yang sejenis disebut dengan entity set
2. Relationship yaitu hubungan yang terjadi antara satu entitas atau lebih
3. Atribut, kumpulan elemen data yang membentuk suatu entitas
4. Indicator tipe dibagi 2, yaitu :
5. Indicator tipe asosiatif object
6. Indicator tipe super tipe

**Entity set terbagi atas, diantaranya :**

1. Strong entity set yaitu entity set yang satu atau lebih atributnya digunakan oleh entity set lain sebagai key, digambarkan dengan persegi Panjang
2. Weak entity set, entity set yang bergantung terhadap strong entity set. Digambarkan dengan persegi Panjang bertumpuk

Contoh entity set :

Weak entity

milik

Tabungan

Pegawai

Contoh strong entity

Contoh :

*Strong entity set*

|  |  |
| --- | --- |
| **Nopeg** | **Nama** |
| 201507340 | Billy |
| 201507341 | Fuad |
| 201507342 | Nining |
| 201507343 | Fintri |

*Weak entity set transaction*

|  |  |  |  |
| --- | --- | --- | --- |
| Nopeg | Tanggungan | Tanggal\_lahir | Jenis\_Kelamin |
| 201507340 | Hafidz | 22 – 03 – 2006 | Laki – Laki |
| 201507341 | Reni | 13 – 05 – 1999 | Perempuan |
| 201507342 | Raffa | 21 – 06 – 2006 | Laki – Laki |
| 201507343 | Naya | 25 – 10 – 2006 | Perempuan |

Jenis – jenis atribut

Berikut ini adalah jenis – jenis atribut, diantaranya :

1. Atribut key -> atribut yang digunakan untuk menentukan suatu entity secara unik
2. Atribut simple -> atribut yang bernilai tunggal
3. Atribut multivalue -> atribut yang memiliki sekolompok nilai untuk setiap entitas.

Contoh :

Pegawai

1. Atribut composit -> suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai atri tertentu.

Contoh :

Pegawai

1. Atribut derivative -> suatu atribut yang dihasilkan dari atribut lain. Sehingga umur yang merupakan hasil dari kalkulasi antara tgl\_lahir dan tanggal\_lahir. Sehingga keberadaan atribut umur bergantung pada keberadaan atribut tgl\_lahir.

Contoh :

Pegawai

**Derajat Relationship**

Menjelaskan jumlah entitas yang berpartisipasi dalam suatu relationship. Berikut adalah macam – macam derajat relationship, diantaranya :

1. **Unary degree**

Pegawai

Lapor

1. **Binary degree**

Pegawai

Kerja

Departemen

1. **Ternary degree**

Pegawai

Kerja

Proyek

Kota

**Cardinality**

Banyaknya entitas sesuai dengan entitas lain melalui relationship / hubungan.

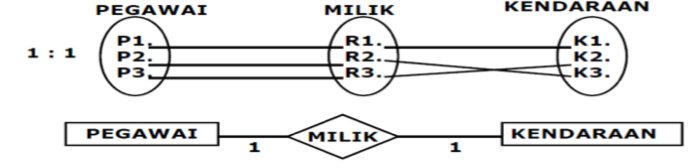
**Representasi dari entity set**

Entity set direpresentasikan dalam bentuk tabel dan nama yang unique. Setiap tabel terdiri dari sejumlah kolom, dimana masing – masing kolom diberi nama yang unique juga.

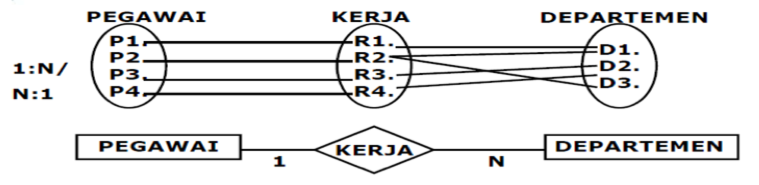
*Cardinality ratio constraint*, mejelaskan Batasan jumlah keterhubungan satu entity dengan entity lainnya.

Berikut ini jenis – jenis cardinality, sebagai berikut :

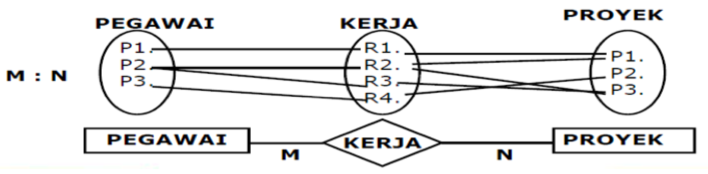
1. One to one (1 to 1)



1. Many to one atau one to many ( 1 to m ) atau (m to 1)



1. Many to many (m to n)



**Normalisasi**

Normalisasi merupakan proses pengelompokan elemen data menjadi tabel – tabel yang menunjukkan entitas dan relasi. Normalisasi adalah proses pengelompokan atribut – atribut dari suatu relasi sehingga memebentuk *WELL STRUCTURE RELATION*.

Keuntungan dari normalisasi, diantaranya :

1. Meminimalkan ukuran penyimpanan yang diperlukan utnuk menyimpan data
2. Meminimalkan resiko inkonsistensi data pada basis data
3. Meminimalkan kemungkinan anomaly pembaruan
4. Memaksimalkan stabilitas struktur data

***WELL STRUCTURE RELATION***

Adalah sebuah relasi yang jumlah kerangkapan datanya sedikit (*minimum amount of redundancy*), serta memberikan kemungkinan bagi user untuk melakukan INSERT, DELETE, dan MODIFY terhadap baris – baris data pada relation tersebut yang tidak berakibat terjadinya error atau inkonsistensi data, yang disebabkan oleh operasi – operasi.

**Bentuk – bentuk normalisasi**

Berikut ini bentuk – bentuk normalisasi, diantaranya :

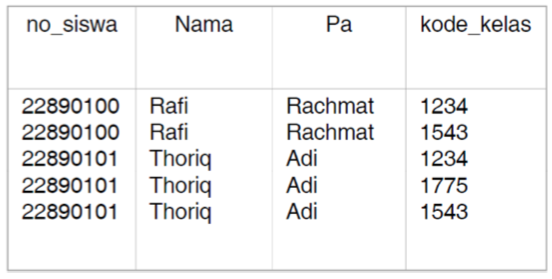
1. Bentuk normal ke satu (1NF / *first normal form*)

Suatu relasi 1NF jik dan hanya jika sifat dari setiap relasi atributnya bersifat atomik. Atomik adalah zat terkecil yang masih memiliki sifat induknya, bila dipecah lagi maka ia tidak memiliki sifat induknya.

Ciri – ciri 1NF, diantaranya :

1. Setiap data dibentuk dalam flat file, data dibentuk dalam satu record demi satu record nilai dari field berupa “atomic value”.
2. Tidak ada set attribute yang berulang atau bernilai ganda
3. Tiap field hanya satu pengertian

Contoh :



1. Bentuk normal ke dua(2NF / *second normal form*)

Bentuk normal kedua mempunyai syarat yaitu bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci haruslah bergantung secara fungsi pada kunci utama / *primary key* . sehingga untuk membentuk normal kedua haruslah sudah ditentukan kunci – kunci field. Kunci fiel haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya.

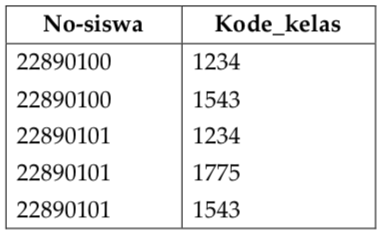
Contoh:

Relasi siswa pada 1NF terlihat bahwa kunci utama / *primary key* adalah nomor siswa. Nama siswa dan PA bergantung fungsi pada no\_siswa, tetapi kode\_kelas bukanlah fungsi dari siswa, maka file siswa dipecah menjadi 2 relasi. Dapat terlihat pada tabel berikut :

Relasi siswa



Dan relasi ambil\_kelas



1. Bentuk normal ke tiga (3NF / *thrid normal form*)

Untuk menjadi bentuk normal ketiga, maka relasi haruslah dalam bentuk normal kedua dan semua atribut buka primer tidak punya hubungan yang transitif. Setiap atribut bukan kunci haruslah bergantung hanya pada primary key dan pada primary key secara menyeluruh.

Contoh :

Bentuk normal kedua diatas termasuk juga bentuk normal ketiga karena seluruh atribut yang ada disitu bergantung penuh pada kunci primer.

1. Bentuk *Boyce-Codd Normal Form* (BCNF)

BCNF mempunyai bentuk yang lebih kuat dari bentuk normal ketiga. Untuk menjadi BCNF, rekasi harus dalam bentuk normal kesatu dan setiap atribut harus bergantung fungsi pada atribut superkey.

Contoh :

Relas seminar dengan ketentuan sebagai berikut :

1. Kunci primer adalah no\_siswa dan seminar
2. Siswa boleh mengambil satu atau dua seminar
3. Setiap siswa dibimbing oleh salah satu diantara 2 instruktur seminar
4. Setiap instruktur hanya boleh mengambil satu seminar saja

**Tabel relasi seminar**



Keterangan :

Bentuk relasi seminar adalah bentuk normal ketiga. Tetapi bukan bentuk BCNF karena nomor seminar masih bergantung fungsi pada instruktur, jika setiap instruktur dapat mengajar hanya pada satu seminar. Seminar bergantung fungsi pada satu atribut bukan superkey seperti yang disyaratkan ileh BCNF. Maka relasi seminar harus dipecah menjadi dua, seperti berikut :



1. Bentuk normal ke empat (4NF )

Relasi R adalah bentuk 4 NF jika dan hanya jika relasi tersebut juga termasuk BCNF dan semua ketergantungan multivalue adalah juga ketergantungan fungsional.

1. Bentuk normal ke lima (5NF)

Bentuk normal ke lima (5NF) disebut juga dengan PJNF (*projection join normal form*) dari bentuk normal ke empat (4NF) dilakukan dengan menghilangkan ketergantungan join yang bukan merupakan kunci kandidat.

**Daftar Pustaka**

Tim Dosen. 2016. Modul Perkuliahan Perancangan Basis data. Universitas Mercu buana.

Tim penyusun. 2009. Modul praktikum perancangan basis data. Bina sarana informatika.

Pasaribu, T. 2011. Materi Perancangan Basis data.