

## BAB II SISTEM BASIS DATA

### A. Capaian Pembelajaran Mata Kuliah

Mahasiswa mampu menjelaskan konsep basis data, dan merancang basis data mulai dari tahap pemodelan data dengan model E-R sampai model relasional, dimana hasil rancangan memenuhi kaedah normalisasi.

### B. Sub-Capaian Capaian Pembelajaran Mata Kuliah

Mahasiswa mampu menjelaskan konsep sistem basis data

### C. Indikator – Indikator Pembelajaran

Mahasiswa mampu:

1. Menjelaskan tentang konsep sistem basis data
2. Menjelaskan tentang model data
3. Menjelaskan abstraksi data
4. Menjelaskan bahasa dalam basis data

### D. Materi

#### 1. Pendahuluan

Dalam bab ini, akan diperkenalkan beberapa konsep berkaitan dengan sistem basis data. Adapun materi yang akan disampaikan adalah definisi sistem basis data, Sistem Pengelola Basis Data (*Database Management System-DBMS*), jenis pengguna,).

#### 2. Sistem Basis Data

##### 2.1 Pengertian Sistem Basis Data

**Sistem** adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan suatu fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu. Basis data adalah sebuah obyek yang pasif/mati. Basis data ada karena ada proses pembuatnya dan akan digunakan oleh program/aplikasi (*software*) sebagai pengelola basis data tersebut. Gabungan dari basis data dan aplikasi sebagai pengelolanya dapat mewujudkan sebuah sistem. Oleh karena itu, secara umum **sistem basis data** adalah sistem yang terdiri atas

kumpulan tabel yang saling berhubungan dalam sebuah sistem data di sebuah sistem komputer dan sekumpulan program yang memungkinkan beberapa pemakai dan/atau program lain untuk mengakses dan memanipulasi tabel-tabel tersebut. Atau dengan kata lain, sistem basis data adalah basis data dengan perangkat pengelolanya yaitu DBMS.

Sedangkan menurut Silberschatz, Korth, & Sudarshan (2020), menyatakan bahwa sistem basis data adalah kumpulan data yang saling terkait dan satu set program yang memungkinkan pengguna untuk mengakses dan memodifikasi data ini. Tujuan utama dari sistem basis data adalah untuk menyediakan pengguna dengan pandangan abstrak dari data. Artinya, sistem menyembunyikan detail tertentu tentang bagaimana data disimpan dan dipelihara.

Menurut Fathansyah (2018), dalam sebuah sistem basis data, terdapat komponen-komponen utama yang terlibat, yaitu:

1. Perangkat Keras (*Hardware*)
2. Sistem Operasi (*Operating System*)
3. Basis Data (*Basis data*)
4. Sistem Pengelola Basis data (*Database magement System - DBMS*)
5. Pengguna (*User*)

## 2.2 Komponen Sistem Basis data

Berikut ini dijelaskan lebih detil mengenai setiap komponen dalam sistem basis data.

### 1) Perangkat Keras (*Hardware*)

Komponen perangkat keras dalam sebuah sistem basis data merupakan komponen dasar yang terdiri dari:

- Komputer dengan kelengkapannya (jika satu untuk yang bersifat *stand alone* atau lebih dari satu untuk sistem jaringan)
- Memori sekunder yang *on-line (harddisk)*
- Memori sekunder yang *off-line (removable disk: flashdisk, harddisk external, CD)* untuk keperluan backup data
- Media/perangkat komunikasi (untuk sistem jaringan)

### 2) Sistem Operasi (*Operating Sistem*)

Secara sederhana, sistem operasi merupakan program yang berfungsi untuk mengaktifkan/ memungsikan sistem komputer, mengendalikan seluruh sumber

daya (*resource*) dalam komputer dan melakukan operasi-operasi dasar dalam komputer (operasi *input-output*, pengelolaan file, dan lain-lain). Sistem operasi diinstal dalam komputer. Adapun jenis-jenis sistem operasi yang banyak digunakan adalah *MS-Dos*, *MS-Windows*, *Unix*, *Xenix*, *Sun-Solaris*, *Linux*, *MS-Windows NT*, *Apple Mac OS*, *Macintosh*, *OS/2*. Program pengelola basis data hanya dapat aktif jika sistem operasi yang dikehendaki (sesuai) telah aktif.

### 3) Basis Data (*Database*)

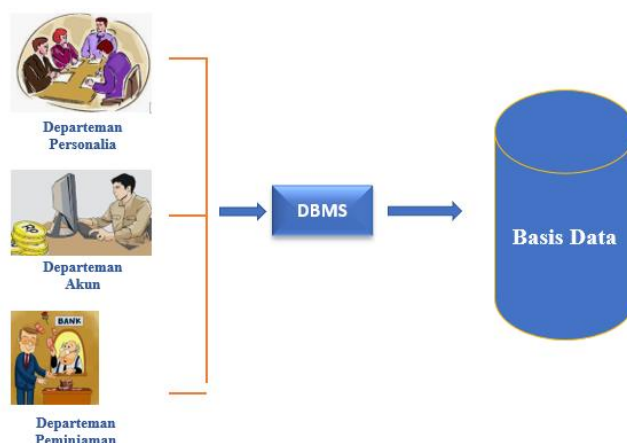
Sebuah sistem basis data dapat memiliki beberapa basis data. Setiap basis data dapat berisi /memiliki sejumlah obyek basis data (seperti tabel, indeks, dan lain-lain). Disamping dapat menyimpan data, setiap basis data juga dapat menyimpan definisi struktur (baik untuk basis data maupun untuk obyek-obyeknya secara detil).

### 4) Sistem Pengelola Basis data (*Database Management System - DBMS*)

Pengelolaan basis data secara fisik tidak dilakukan oleh pengguna secara langsung, tetapi ditangani oleh sebuah perangkat lunak/sistem yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. Sistem tersebut adalah sistem pengelola basis data (DBMS). Ia juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan/konsistensi data, dan sebagainya.

Menurut Watt & Eng (2014), sistem pengelola basis data adalah kumpulan program yang memungkinkan pengguna untuk membuat dan memelihara basis data serta mengontrol semua akses ke basis data tersebut. Tujuan utama dari DBMS adalah untuk menyediakan lingkungan yang nyaman dan efisien bagi pengguna untuk mengambil dan menyimpan data. Perangkat lunak yang termasuk DBMS seperti *dBase*, *FoxBase*, *MS-Access*, *Borland-Paradox*, *Borland-Interbase*, *MS-SQL Server*, *MySQL*, *SyBase*, *CA-Open Ingres*, *Informix*, *firebird*, *teradata*, *postgreSQL* dan *Oracle*.

Sebagai contoh sebuah sistem perbankan tradisional yang menggunakan pendekatan basis data yang ditunjukkan pada Gambar 2.1. Dalam contoh bank ini, DBMS digunakan oleh Departemen Personalia, Departemen Akun dan Departemen Pinjaman untuk mengakses basis data perusahaan bersama.



Gambar 2.1 Sistem Pengelolaan Basis data sebuah Bank

Menurut Elmasri & Navathe (2015), bahwa ada beberapa keuntungan ketika menggunakan DBMS dan kemampuan yang harus dimiliki oleh DBMS yang baik. DBA harus memanfaatkan kemampuan ini untuk mencapai berbagai tujuan yang terkait dengan desain, administrasi, dan penggunaan basis data multipengguna yang besar. Adapun keuntungan dari penggunaan DBMS, yaitu

- 1) Mampu mengontrol redundansi (*controlling redundancy*)
- 2) Mampu membatasi akses yang tidak sah (*restricting unauthorized access*)
- 3) Dapat menyediakan penyimpanan yang persisten untuk objek program (*providing persistent storage for program objects*)
- 4) Menyediakan struktur penyimpanan dan teknik pencarian untuk pemrosesan kueri yang efisien (*providing storage structures and search techniques for efficient query processing*)
- 5) Menyediakan pencadangan dan pemulihan (*providing backup and recovery*)
- 6) Menyediakan beberapa antarmuka pengguna (*providing multiple user interfaces*)
- 7) Mewakili hubungan kompleks antara data (*representing complex relationships among data*)
- 8) Menegaskan batasan integritas (*enforcing integrity constraints*)
- 9) Mengizinkan inferensi dan tindakan menggunakan aturan dan trigger (*permitting inferencing and actions using rules and triggers*)
- 10) Implikasi tambahan menggunakan pendekatan basis data (*additional implications of using the database approach*)

## 5) Pengguna (*User*)

Untuk basis data yang kecil, seperti daftar alamat yang dibahas, satu orang biasanya mendefinisikan, membangun, dan memanipulasi basis data, dan tidak ada pembagian. Namun, dalam organisasi besar, banyak orang yang terlibat dalam desain, penggunaan, dan pemeliharaan basis data besar dengan ratusan atau ribuan pengguna. Menurut Elmasri & Navathe (2016), ada 4 (empat) jenis pengguna yang berinteraksi dengan sistem basis data, yaitu:

### a) Administrator Basis Data (*Database Administrators-DBA*)

Dalam sebuah organisasi, banyak orang menggunakan sumber daya yang sama. Ada kebutuhan bagi administrator utama untuk mengawasi dan mengelola sumber daya tersebut. Dalam lingkungan basis data, sumber daya utama adalah basis data itu sendiri, dan sumber daya sekunder adalah DBMS dan perangkat lunak terkait. Mengelola sumber daya ini adalah tanggung jawab administrator basis data (DBA). DBA bertanggung jawab untuk memberikan wewenang melakukan akses ke basis data, mengoordinasikan dan memantau penggunaannya, dan memperoleh sumber daya perangkat lunak dan perangkat keras sesuai kebutuhan. DBA juga bertanggung jawab atas masalah seperti pelanggaran keamanan dan waktu respons sistem yang buruk.

Menurut Silberschatz, Korth, & Sudarshan (2020), fungsi dari DBA, yaitu

- Mendefinisikan skema. DBA membuat skema basis data asli dengan mengeksekusi satu set pernyataan definisi data pada tahap *Data Definition Languages* (DDL).
- Pendefinisian struktur penyimpanan dan metode pengaksesan. DBA dapat menentukan beberapa parameter yang berkaitan dengan organisasi fisik data dan indeks yang akan dibuat.
- Modifikasi skema dan pengorganisasian secara fisik. DBA melakukan perubahan skema dan pengorganisasian secara fisik untuk mencerminkan perubahan kebutuhan organisasi, atau untuk mengubah organisasi fisik untuk meningkatkan kinerja.
- Pemberian otoritas/wewenang untuk pengaksesan data. Dengan memberikan berbagai jenis wewenang, administrator basis data dapat mengatur bagian mana dari basis data yang dapat diakses oleh berbagai pengguna. Informasi wewenang pengaksesan disimpan dalam struktur

sistem khusus yang dikonsultasikan oleh sistem basis data setiap kali pengguna mencoba mengakses data dalam sistem.

- Melakukan pemeliharaan rutin. Contoh kegiatan pemeliharaan rutin dari DBA adalah:

- Secara berkala mencadangkan (*backup*) basis data ke server jarak jauh (*online*), untuk mencegah hilangnya data jika terjadi bencana seperti banjir yang dapat merusak server secara fisik.
- Memastikan bahwa ruang penyimpanan kosong yang cukup tersedia untuk operasi normal, dan meningkatkan ruang penyimpanan sesuai kebutuhan.
- Memantau pekerjaan yang berjalan pada basis data dan memastikan bahwa kinerja tidak diturunkan oleh tugas yang sangat mahal/berat yang diajukan oleh beberapa pengguna.

#### b) Perancang Basis data (*Database Designers*)

Perancang basis data bertanggung jawab untuk mengidentifikasi data yang akan disimpan dalam basis data dan untuk memilih struktur yang sesuai untuk mewakili dan menyimpan data ini. Tugas-tugas ini sebagian besar dilakukan sebelum basis data benar-benar diimplementasikan dan diisi dengan data. Ini adalah tanggung jawab perancang basis data untuk berkomunikasi dengan semua calon pengguna basis data untuk memahami kebutuhan mereka dan untuk membuat desain/rancangan yang memenuhi persyaratan ini. Dalam banyak kasus, perancang biasanya merupakan staf DBA dan dapat diberi tanggung jawab oleh staf lain setelah basis data selesai dirancang. Perancang basis data biasanya berinteraksi dengan setiap kelompok pengguna yang potensial mengembangkan tampilan basis data yang memenuhi persyaratan data dan pemrosesan kelompok ini. Setiap tampilan kemudian dianalisis dan diintegrasikan dengan tampilan grup pengguna lainnya. Perancang basis data akhir harus mampu mendukung kebutuhan semua kelompok pengguna.

#### c) Pengguna Akhir (*End Users*)

Pengguna akhir adalah orang-orang yang pekerjaannya memerlukan akses ke basis data untuk membuat kueri, memperbarui, dan membuat laporan. Ada beberapa kategori pengguna akhir, yaitu:

- **Pengguna akhir biasa** (*Casual end users*) kadang-kadang mengakses basis data, tetapi mereka mungkin memerlukan informasi yang berbeda setiap

kali. Pengguna jenis ini menggunakan antarmuka kueri basis data yang canggih untuk menentukan permintaan mereka dan biasanya merupakan manajer tingkat menengah atau tinggi atau browser sesekali lainnya.

- **Pengguna akhir yang naif atau parametrik** (*Naive or parametric end users*) merupakan bagian yang cukup besar dari pengguna akhir basis data. Pekerjaan utama mereka berkisar pada kueri dan pembaruan basis data secara terus-menerus, menggunakan jenis kueri dan pembaruan standar—disebut transaksi kalengan—yang telah diprogram dan diuji dengan cermat. Banyak dari tugas ini sekarang tersedia sebagai aplikasi seluler untuk digunakan dengan perangkat seluler. Tugas yang dilakukan pengguna tersebut bervariasi.

Contohnya adalah:

- Pelanggan dan teller bank memeriksa saldo rekening dan memposting penarikan dan penyetoran.
  - Agen reservasi atau pelanggan untuk maskapai penerbangan, hotel, dan perusahaan persewaan mobil memeriksa ketersediaan untuk permintaan yang diberikan dan membuat reservasi.
  - Karyawan di stasiun penerima untuk perusahaan pelayaran memasukkan identifikasi paket melalui kode batang dan informasi deskriptif melalui tombol untuk memperbarui basis data pusat dari paket yang diterima dan dalam perjalanan.
  - Pengguna media sosial memposting dan membaca item di situs Web media sosial
- **Pengguna akhir yang canggih** (*Sophisticated end users*) termasuk insinyur, ilmuwan, analis bisnis, dan orang lain yang secara menyeluruh membiasakan diri dengan fasilitas DBMS untuk mengimplementasikan aplikasi mereka sendiri untuk memenuhi persyaratan kompleks mereka.
- **Pengguna mandiri** (*Standalone users*) memelihara basis data pribadi dengan menggunakan paket program siap pakai yang menyediakan antarmuka berbasis menu atau grafis yang mudah digunakan. Contohnya adalah pengguna paket perangkat lunak keuangan yang menyimpan berbagai data keuangan pribadi.

d) Analis Sistem dan Pemrogram Aplikasi (*Software Engineers*)

Analisis sistem menentukan persyaratan pengguna akhir (terutama bagi pengguna akhir yang naif dan parametrik), dan mengembangkan spesifikasi untuk transaksi standar yang memenuhi persyaratan ini. Pemrogram aplikasi (*programmer*) mengimplementasikan spesifikasi ini sebagai program; kemudian mereka menguji, menemukan *bug/error*, mendokumentasikan, dan memelihara jenis transaksi ini. Analisis dan pemrogram seperti itu—biasanya disebut sebagai pengembang perangkat lunak (*software developer*)—harus terbiasa dengan berbagai kemampuan yang disediakan oleh DBMS untuk menyelesaikan tugas mereka.

### 3. Model Data

#### 3.1 Definisi Model Data

Salah satu karakteristik mendasar dari pendekatan basis data adalah menyediakan beberapa tingkat abstraksi data. Abstraksi data umumnya mengacu pada penekanan rincian organisasi dan penyimpanan data, dan menyoroti fitur-fitur penting untuk pemahaman data yang lebih baik. Salah satu karakteristik utama dari pendekatan basis data adalah untuk mendukung abstraksi data sehingga pengguna yang berbeda dapat melihat data pada tingkat detail yang mereka inginkan. Menurut Elmasri & Navathe (2016), bahwa model data merupakan kumpulan konsep yang dapat digunakan untuk menggambarkan struktur basis data dan menyediakan sarana yang diperlukan untuk mencapai abstraksi data ini. Yang dimaksud dengan struktur basis data adalah tipe data, hubungan, dan batasan yang berlaku untuk data. Fathansyah (2018) mendefinisikan model data sebagai kumpulan perangkat konseptual untuk menggambarkan data, hubungan data, semantik (makna) data dan batasan data). Connolly & Begg (2015), mendefinisikan bahwa model data adalah kumpulan konsep yang terintegrasi untuk mendeskripsikan dan memanipulasi data, hubungan antar data, dan batasan pada data dalam suatu organisasi. Sedangkan Date (2019) memberikan dua definisi berkaitan dengan model data ini. Definisi pertama, model data adalah seperti bahasa pemrograman, yang konstruksinya dapat digunakan untuk memecahkan banyak masalah khusus tetapi dalam dan dari dirinya sendiri tidak memiliki hubungan langsung dengan masalah khusus semacam itu. Dan definisi kedua memberikan pengertian bahwa model data adalah seperti program khusus yang ditulis dalam bahasa pemrograman pada definisi pertama, dimana model tersebut menggunakan fasilitas yang disediakan oleh



model, dalam pengertian pertama istilah itu, untuk memecahkan beberapa masalah tertentu.

Sebagian besar model data juga menyertakan satu set operasi dasar untuk menentukan pengambilan dan pembaruan data pada basis data. Selain operasi dasar yang disediakan oleh model data, menjadi lebih umum untuk memasukkan konsep dalam model data untuk menentukan aspek dinamis atau perilaku aplikasi basis data. Hal ini memungkinkan perancang basis data untuk menentukan satu set operasi yang ditentukan pengguna yang valid yang diizinkan pada objek basis data.

### 3.2 Kategori Model Data

Banyak model data telah diusulkan, yang dapat dikategorikan sesuai dengan jenis konsep yang digunakan untuk menggambarkan struktur basis data.

Ada dua buah jenis model data, yaitu model data tingkat tinggi atau model data konseptual, dan model data tingkat rendah atau model data fisik.

**a) Model data tingkat tinggi atau model data konseptual** memberikan konsep yang mirip dengan cara banyak pengguna melihat data.

Contohnya adalah model *entity-relationship* (Model ER), yang menggunakan konsep utama seperti entitas, atribut, dan relasi. Entitas mewakili objek dunia nyata seperti karyawan atau proyek. Entitas memiliki atribut yang mewakili properti seperti nama, alamat, dan tanggal lahir karyawan. Sebuah relasi mewakili sebuah hubungan antara entitas; misalnya, seorang karyawan mengerjakan banyak proyek. Ada relasi antara karyawan dan masing-masing proyek.

**b) Model data tingkat rendah atau model data fisik** memberikan konsep yang menjelaskan detail tentang bagaimana data disimpan di media penyimpanan komputer (*harddisk*). Konsep yang diberikan oleh model data fisik umumnya dimaksudkan untuk spesialis komputer, bukan untuk pengguna akhir.

Di antara dua jenis model data ini ada dikenal dengan **model data representasional** (atau **implementasi**), yang memberikan konsep yang dapat dengan mudah dipahami oleh pengguna akhir tetapi tidak terlalu jauh dari cara data diatur dalam penyimpanan komputer. Model data representasional menyembunyikan banyak detail penyimpanan data pada tempat penyimpanan tetapi dapat diimplementasikan pada sistem komputer secara langsung. Model data representasi atau implementasi adalah model yang paling sering digunakan dalam DBMS komersial tradisional contohnya MySQL. Ini termasuk

model data relasional yang banyak digunakan, serta apa yang disebut model data lama—model jaringan dan hierarki—yang telah banyak digunakan di masa lalu. Model data representasional mewakili data dengan menggunakan struktur record dan karenanya kadang-kadang disebut **model data berbasis record**.

Kita dapat menganggap **model data berbasis objek** sebagai contoh dari keluarga baru model data tingkat tinggi yang lebih dekat dengan model data konseptual. Model data objek juga sering digunakan sebagai model konseptual tingkat tinggi, khususnya dalam domain rekayasa perangkat lunak.

### 3.3 Abstraksi Data

Agar sistem/aplikasi dapat digunakan, oleh karena itu sistem harus mengambil data secara efisien. Kebutuhan akan efisiensi telah mendorong pengembang sistem basis data untuk menggunakan struktur data yang kompleks untuk mewakili data dalam basis data. Karena banyak pengguna sistem basis data tidak terlatih dengan komputer, pengembang menyembunyikan kerumitan yang ada dari pengguna melalui beberapa tingkat abstraksi data, untuk menyederhanakan interaksi pengguna dengan sistem/aplikasi.

Ada tiga level dalam abstraksi data, yaitu

a) Level Fisik (*Physical Level*)

Level fisik merupakan level abstraksi terendah. Level ini menggambarkan bagaimana data sebenarnya disimpan. Pada level ini pengguna melihat data sebagai gabungan dari struktur dan datanya sendiri. Pengguna juga memiliki kompetensi untuk mengetahui bagaimana gambaran fisik dari penyimpanan data. Pada level ini, pengguna berurusan dengan data sebagai teks, angka, atau bahkan melihatnya sebagai himpunan bit data.

b) Level Logis/Konseptual (*Logical Level*)

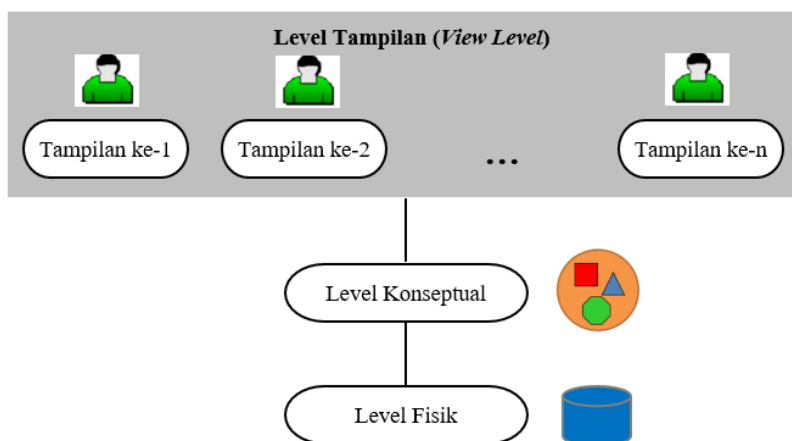
Level abstraksi berikutnya yang lebih tinggi level konseptual. Level ini menjelaskan tentang data apa yang disimpan dalam basis data, dan hubungan apa yang ada di antara data tersebut. Level ini menggambarkan seluruh basis data dalam jumlah kecil struktur yang relatif sederhana. Meskipun implementasi struktur sederhana pada level ini mungkin melibatkan struktur level fisik yang kompleks, akan tetapi pengguna pada level konseptual tidak perlu menyadari kompleksitas ini. Ini disebut sebagai kebebasan data fisik. Administrator basis

data, yang harus memutuskan informasi apa yang akan disimpan dalam basis data, menggunakan tingkat abstraksi logis/konseptual.

c) Level Tampilan (*View Level*)

Level tampilan merupakan level abstraksi tertinggi. Level ini menggambarkan sebagian dari keseluruhan basis data. Meskipun pada level logis menggunakan struktur yang lebih sederhana, kompleksitas tetap ada karena berbagai informasi yang disimpan dalam basis data yang besar. Banyak pengguna sistem basis data tidak membutuhkan semua informasi ini, sebaliknya, mereka hanya perlu mengakses sebagian dari basis data. Adanya level tampilan ini adalah untuk menyederhanakan interaksi pengguna dengan sistem/aplikasi. Sistem dapat menyediakan banyak tampilan untuk basis data yang sama

Pada Gambar 2.5 ditunjukkan hubungan yang ada antara ketiga level abstraksi data.



Gambar 2.5. Hubungan antara ketiga level abstraksi data

#### 4 Bahasa Basis Data

Sebelumnya telah dibahas berbagai pengguna yang didukung oleh DBMS. DBMS harus menyediakan bahasa yang sesuai untuk setiap kategori pengguna. Pada bagian ini akan dibahas mengenai jenis bahasa yang disediakan oleh DBMS.

Setelah rancangan basis data selesai dan DBMS dipilih untuk mengimplementasikan dbasis data, langkah pertama adalah menentukan skema konseptual dan internal untuk database dan pemetaan di antara keduanya. Dalam DBMS, dikenal ada tiga jenis bahasa yaitu *Data Definition Language* (DDL), *Data Control Language* (DCL), dan *Data Manipulation Language* (DML).

#### 4.1 Data Definition Language (DDL)

DDL adalah bahasa dalam DBMS yang digunakan untuk mendefinisikan (membuat, memodifikasi & menghapus) sebuah basis data ataupun tabel. DDL berisi sejumlah perintah dasar, yaitu:

##### 1) Create

*Create* merupakan perintah untuk membuat atau mendefinisikan. Adapun bentuk-bentuk perintah *create* adalah:

- *Create Database*, yaitu perintah untuk membuat basis data baru
- *Create Function*, yaitu perintah untuk membuat fungsi pada basis data
- *Create Index*, adalah perintah untuk membuat index pada basis data
- *Create Table*, adalah perintah untuk membuat tabel pada basis data
- *Create Procedure*, adalah perintah untuk membuat prosedur pada basis data
- *Create Trigger*, adalah perintah untuk membuat trigger pada basis data

##### 2) Rename,

*Rename* digunakan untuk melakukan perubahan. Terdapat sebuah perintah yaitu *Rename Table* yang berfungsi untuk mengganti nama tabel.

##### 3) Alter, dan

*Alter* hanya memiliki satu perintah saja, yaitu *Alter Table* yang berguna untuk melakukan perubahan pada struktur tabel.

##### 4) Drop

*Drop* merupakan perintah yang digunakan untuk melakukan penghapusan, Ada beberapa bentuk perintah *Drop*, yaitu:

- *Drop Database*, yaitu perintah untuk menghapus basis data
- *Drop Function*, yaitu perintah untuk menghapus fungsi pada basis data
- *Drop Index*, yaitu perintah untuk menghapus index pada tabel dalam basis data
- *Drop Procedure*, yaitu perintah untuk menghapus prosedur pada basis data

- *Drop Tabel*, yaitu perintah untuk menghapus tabel pada basis

#### 4.2 *Data Manipulation Language (DML)*

DML merupakan bahasa dalam DBMS yang digunakan untuk melakukan pengolahan data pada tabel dalam sebuah basis data. Adapun perintah-perintah dasar pada DML ini, adalah:

- 1) *Select*, adalah perintah yang digunakan untuk menampilkan record data pada sebuah tabel
- 2) *Insert*, adalah perintah yang digunakan untuk menambahkan record data baru pada tabel
- 3) *Update*, adalah perintah yang digunakan untuk mengubah data pada tabel
- 4) *Delete*, adalah digunakan untuk menghapus record data pada tabel

#### 4.3 *Data Control Language (DCL)*

DCL merupakan bahasa dalam DBMS yang berfungsi untuk melakukan kontrol terhadap data atau obyek dalam sebuah basis data. DCL

Adapun perintah dasar dari DCL, adalah:

##### 1) *Grant*

*Grant* merupakan perintah yang berhubungan dengan hak akses terhadap pengguna basis data. Dengan menggunakan perintah ini, seorang administrator (DBA) dapat memberikan hak akses terhadap pengguna untuk melakukan manajemen/pengelolaan dan pengaksesan terhadap basis data.

Dengan hak akses tersebut memungkinkan pengguna yang ditunjuk dapat melakukan proses manajemen basis data dengan menggunakan perintah – perintah DML dan juga DDL.

##### 2) *Revoke*

*Revoke* merupakan kebalikan dari perintah *grant*. Perintah *revoke* ini digunakan oleh administrator untuk menarik atau mencabut hak akses yang sudah diberikan sebelumnya kepada pengguna. Pencabutan hak akses ini bisa disebabkan oleh berbagai macam kondisi, tergantung keputusan dari administrator.

Pada intinya, fungsi dari DCL atau fungsi pengontrolan ini bisa dilakukan oleh seorang administrator, yang bertanggung jawab terhadap basis data dari sebuah perusahaan ataupun institusi.

## E. Rangkuman

**Sistem** adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan suatu fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu. Sedangkan **sistem basis data** adalah sistem yang terdiri atas kumpulan tabel yang saling berhubungan dalam sebuah sistem data di sebuah sistem komputer dan sekumpulan program yang memungkinkan beberapa pemakai dan/atau program lain untuk mengakses dan memanipulasi tabel-tabel tersebut. Dalam sistem basis data, terdapat komponen-komponen utama yang terlibat, yaitu

1. Perangkat Keras (*Hardware*)
2. Sistem Operasi (*Operating System*)
3. Basis Data (*Basis data*)
4. Sistem Pengelola Basis data (*Database management System* - DBMS)
5. Pengguna (*User*)

Pada bab ini juga dibahas berkaitan model data yang meliputi pengertian model data, serta pengkategorianya. Berdasarkan jenis konsep yang digunakan untuk menggambarkan struktur basis data, model data dikategorikan menjadi dua yaitu model data tingkat tinggi atau model data konseptual, dan model data tingkat rendah atau model data fisik.

Selanjutnya pembahasan mengenai abstraksi data dibagian 3.3. Ada tiga tingkatan/level dalam abstraksi data, yaitu level fisik, level konseptual, dan level tampilan. Di setiap level, memiliki fungsi-fungsi tersendiri berkaitan pengaksesan data dalam basis data.

Terakhir dalam bab ini dibahas mengenai bahasa basis data. Ada tiga jenis bahasa yang berperan dalam basis data, yaitu *Data Definition Language* (DDL), *Data Control Language* (DCL), dan *Data Manipulation Language* (DML). Di setiap jenis bahasa ini memiliki perintah-perintah dasar yang digunakan untuk mengakses data dalam basis data.

## F. Latihan Soal

1. Coba definisikan sistem basis data menggunakan bahasa sendiri!

2. Sebutkan komponen-komponen utama yang terlibat dalam sistem basis data!
3. Sebutkan minimal 5 buah contoh yang termasuk dalam sistem operasi!
4. Sebutkan minimal 5 buah contoh DBMS!
5. Sebutkan tiga buah keuntungan dari penggunaan DBMS!
6. Model data dikategorikan menjadi dua, coba sebutkan!
7. Ada tiga level dalam abstraksi data, silakan sebutkan!
8. Sebutkan perintah-perintah dasar yang termasuk dalam DDL!

## **G. Referensi**

- Bagui, Sikha Saha & Richard Walsh Earp. 2023. Database Design Using Entity-Relationship Diagrams. Third Edition. New York: CRC Press
- Connolly, Thomas M., & Carolyn E. Begg. 2015. Database Systems: A Practical Approach to Design, Implementation, and Management. England: Pearson.
- Dantes, Gede Rasben, Komang Setemen, Ni Wayan Marti, I Ketut Resika Arthana, Kadek Surya Mahedy, dan Putu Hendra Suputra. 2017. Pengantar Basis Data. Depok: Rajawali Press
- Date, C. J. 2019. Database Design and Relational Theory: Normal Forms and All That Jazz. Second Edition. USA: Apress
- Elmasri, Ramez & Shamkant B. Navathe. 2016. Fundamental of Database Systems. Seventh Edition. United State: Pearson.
- Fathansyah. 2018. Basis Data. Revisi Ketiga. Bandung: Informatika.
- Silberschatz, A., Hendry F. Korth, & S. Sudarshan. 2020. Database System Concepts. Seventh Edition. New York: McGraw-Hill Education.
- Watt, Adrienne & Nelson Eng. 2014. Database Design. Second Edition. Victoria: BCCampus.