

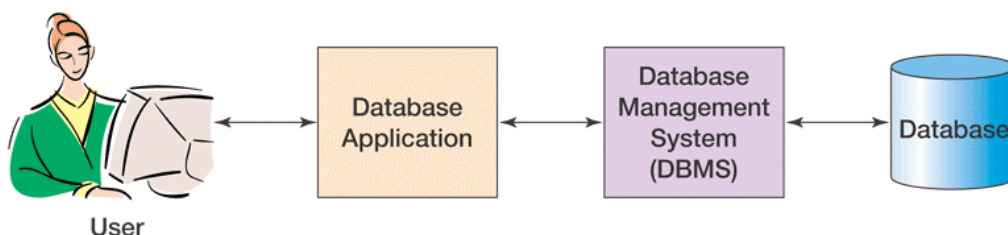
<b>Mata Kuliah</b>	<b>:</b>	<b>ICT Literacy</b>
<b>Bobot Sks</b>	<b>:</b>	<b>2 sks</b>
<b>Dosen Pengembang</b>	<b>:</b>	<b>Cian Ramadhona Hassolthine, S.Kom., M.Kom</b>
<b>Dosen Tutor</b>		
<b>Capaian Pembelajaran Mata Kuliah</b>	<b>:</b>	<b>Mahasiswa memahami konsep basis data</b>
<b>Kompetensi Akhir Di Setiap Tahap (Sub-Cpmk)</b>	<b>:</b>	<b>Mahasiswa dapat memahami konsep basis data dan dapat memahami tahapan dalam daur hidup pengembangan sistem basisdata</b>
<b>Minggu Perkuliahan Online Ke-</b>	<b>:</b>	<b>4</b>

## Pengantar Basis Data

### Konsep Basis Data

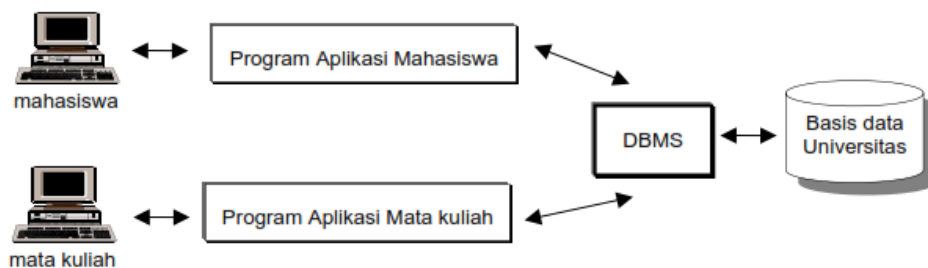
- **Data :**  
Representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, mahasiswa, pembeli), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.
- **Basis Data :**  
Sekumpulan data yang terintegrasi yang diorganisasikan untuk memenuhi kebutuhan para pemakai di dalam suatu organisasi.
- **DBMS (Database Management System) :**  
Perangkat Lunak yang menangani semua pengaksesan ke basis data.

Sistem Basis Data terdiri dari basis data dan DBMS.



## Sistem Basis Data

Seiring dengan berjalannya waktu, lambat laun sistem pemrosesan file mulai ditinggalkan karena masih bersifat manual, yang kemudian dikembangkanlah system pemrosesan dengan pendekatan basis data.



Keterangan :

Mhs ( Npm, nama, alamat, tgl\_lahir )

Mt\_kul ( kd\_mk, nama\_mk,sks )

## Istilah – Istilah Dasar Basis Data

- Enterprise

Suatu bentuk organisasi seperti : bank, universitas, rumah sakit, pabrik, dsb. Data yang disimpan dalam basis data merupakan data operasional dari suatu enterprise.

Contoh data operasional : Data keuangan, data mahasiswa, data pasien

- Entitas

Suatu obyek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data.

Contoh Entitas dalam lingkungan bank terdiri dari : Nasabah, Simpanan, Hipotik

Contoh Entitas dalam lingkungan universitas terdiri dari : Mahasiswa, mata kuliah

Kumpulan dari entitas disebut Himpunan Entitas

Contoh : semua nasabah, semua mahasiswa

- Atribut (Elemen Data)

Karakteristik dari suatu entitas.

Contoh : Entitas Mahasiswa atributnya terdiri dari Npm, Nama, Alamat, Tanggal lahir.

- Nilai Data (*Data Value*)

Isi data / informasi yang tercakup dalam setiap elemen data.

Contoh Atribut Nama Mahasiswa dapat berisi Nilai Data : Diana, Sulaeman, Lina

- Kunci Elemen Data (*Key Data Element*)

Tanda pengenal yang secara unik mengidentifikasi entitas dari suatu kumpulan entitas.

Contoh Entitas Mahasiswa yang mempunyai atribut-atribut npm, nama, alamat,

tanggal lahir menggunakan Kunci Elemen Data npm.

- Record Data

Kumpulan Isi Elemen data yang saling berhubungan.

Contoh : kumpulan atribut npm, nama, alamat, tanggal lahir dari Entitas Mahasiswa berisikan : "10200123", "Sulaeman", "Jl. Sirsak 28 Jakarta", "8 Maret 1983".

## Kelebihan dan Kelemahan Sistem Basis Data

### Kelebihan Sistem Basis Data

- Terkontrolnya kerangkapan data

Dalam basis data hanya mencantumkan satu kali saja field yang sama yang dapat dipakai oleh semua aplikasi yang memerlukannya.

- Terpeliharanya keselarasan (kekonsistenan) data

Apabila ada perubahan data pada aplikasi yang berbeda maka secara otomatis perubahan itu berlaku untuk keseluruhan

- Data dapat dipakai secara bersama (shared)

Data dapat dipakai secara bersama-sama oleh beberapa program aplikasi (secara batch maupun on-line) pada saat bersamaan.

- Dapat diterapkan standarisasi  
Dengan adanya pengontrolan yang terpusat maka DBA dapat menerapkan standarisasi data yang disimpan sehingga memudahkan pemakaian, pengiriman maupun pertukaran data.
- Keamanan data terjamin  
DBA dapat memberikan batasan-batasan pengaksesan data, misalnya dengan memberikan password dan pemberian hak akses bagi pemakai (misal : modify, delete, insert, retrieve)
- Terpeliharanya integritas data  
Jika kerangkapan data dikontrol dan kekonsistenan data dapat dijaga maka data menjadi akurat
- Terpeliharanya keseimbangan (keselarasan) antara kebutuhan data yang berbeda dalam setiap aplikasi  
Struktur basis data diatur sedemikian rupa sehingga dapat melayani pengaksesan data dengan cepat
- Data independence (kemandirian data)  
Dapat digunakan untuk bermacam-macam program aplikasi tanpa harus merubah format data yang sudah ada

#### Kelemahan Sistem Basis Data

- Memerlukan tenaga spesialis
- Kompleks
- Memerlukan tempat yang besar
- Mahal

## Pengguna Basis Data

### *System Engineer*

---

Tenaga ahli yang bertanggung jawab atas pemasangan Sistem Basis Data, dan juga mengadakan peningkatan dan melaporkan kesalahan dari sistem tersebut kepada pihak penjual.

### *Database Administrator (DBA)*

---

Tenaga ahli yang mempunyai tugas untuk mengontrol sistem basis data secara keseluruhan, meramalkan kebutuhan akan sistem basis data, merencanakannya, dan mengaturnya.

Tugas DBA :

- Mengontrol DBMS dan *software-software*
- Memonitor siapa yang mengakses basis data
- Mengatur pemakaian basis data
- Memeriksa *security, integrity, recovery dan concurency*

Program Utilitas yang digunakan oleh DBA :

- *Loading Routines* : Membangun versi utama dari basis data
- *Reorganization Routines* : Mengatur / mengorganisasikan kembali basis data
- *Journaling Routines* : Mencatat semua operasi pemakaian basis data
- *Recovery Routines* : Menempatkan kembali data, sebelum terjadinya kerusakan
- *Statistical Analysis Routines* : Membantu memonitor kehandalan sistem

### *End User (Pemakai Akhir)*

---

Ada beberapa jenis (tipe) pemakai terhadap suatu sistem basis data yang dapat dibedakan berdasarkan cara mereka berinteraksi terhadap sistem :

- Programmer aplikasi  
Pemakai yang berinteraksi dengan basis data melalui = Language (DML), yang disertakan (*embedded*) dalam program yang ditulis pada bahasa pemrograman induk (seperti C, pascal, cobol, dll)
- Pemakai Mahir (*Casual User*)  
Pemakai yang berinteraksi dengan sistem tanpa menulis modul program. Mereka menyatakan query (untuk akses data) dengan bahasa query yang telah disediakan oleh suatu DBMS
- Pemakai Umum (*End User / Naïve User*)  
Pemakai yang berinteraksi dengan sistem basis data melalui pemanggilan satu program aplikasi permanen (*executable program*) yang telah ditulis (disediakan) sebelumnya
- Pemakai Khusus (*Specialized/Sophisticated User*)  
Pemakai yang menulis aplikasi basis data non konvensional, tetapi untuk keperluan-keperluan khusus seperti aplikasi AI, Sistem Pakar, Pengolahan Citra, dll, yang bisa saja mengakses basis data dengan atau tanpa DBMS yang bersangkutan.

## Database Management Systems (DBMS)

### Bahasa Dalam DBMS

DBMS (Database Management systems) adalah kumpulan program yang mengkoordinasikan semua kegiatan yang berhubungan dengan basis data. Dengan adanya berbagai tingkatan pandangan dalam suatu basis data maka untuk mengakomodasikan masing-masing pengguna dalam piranti lunak manajemen basis data biasanya terdapat bahasa-bahasa tertentu yang disebut Data Sub language.

Data sub language adalah subset bahasa yang dipakai untuk operasi manajemen basis data. Dalam penggunaan biasanya dapat ditempelkan (*embedded*) pada bahasa tuan rumah (Cobol, PL/1, dsb). Secara umum maka setiap pengguna basis data memerlukan bahasa yang dipakai sesuai tugas dan fungsinya.

Dalam basis data secara umum dikenal 2 data sub language :

- **Data Definition Language (DDL)**  
Bahasa yang digunakan dalam mendefinisikan struktur atau kerangka dari basis data, di dalamnya termasuk record, elemen data, kunci elemen, dan relasinya
- **Data Manipulation Language (DML)**  
Bahasa yang digunakan untuk menjabarkan pemrosesan dari basis data, fasilitas ini diperlukan untuk memasukkan, mengambil, mengubah data. DML dipakai untuk operasi terhadap isi basis data

Ada 2 jenis DML :

- **Procedural DML**  
Digunakan untuk mendefinisikan data yang diolah dan perintah yang akan dilaksanakan.
- **Non Procedural**  
Digunakan untuk menjabarkan data yang diinginkan tanpa menyebutkan bagaimana cara pengambilannya.

## Tugas DBMS

---

DBMS mempunyai tugas untuk menangani semua bentuk akses kepada basis data, secara konsep :

- Pengguna menyatakan permintaan akses menggunakan DBMS
- DBMS menangkap dan menginterpretasikan
- DBMS mencari :
  - Eksternal / conceptual mapping
  - Conceptual schema
  - Konseptual / internal mapping
  - Internal schema
- DBMS melaksanakan operasi yang diminta terhadap basis data tersimpan. Proses 1 s/d 4 dapat dilakukan secara interactive atau dicompile dulu.

### Model Data

Model data adalah kumpulan konsep yang terintegrasi yang menggambarkan data, hubungan antara data dan batasan-batasan data dalam suatu organisasi. Fungsi dari sebuah model data untuk merepresentasikan data sehingga data tersebut mudah dipahami.

Untuk menggambarkan data pada tingkat eksternal dan konseptual digunakan model data berbasis objek atau model data berbasis record.

- **Model Data Berbasis Objek**

Model data berbasis objek menggunakan konsep entitas, atribut dan hubungan antar entitas. Beberapa jenis model data berbasis objek yang umum adalah :

- Entity-relationship
- Semantic
- Functional
- Object-oriented

- **Model Data Berbasis Record**

Pada model data berbasis record, basis data terdiri dari sejumlah record dalam bentuk yang tetap yang dapat dibedakan dari bentuknya. Ada 3 macam jenis model data berbasis record yaitu :

- Model data relasional (relational)
- Model data hierarkhi (hierarchical)
- Model data jaringan (network)

### Fungsi DBMS

---

Layanan-layanan yang sebaiknya disediakan oleh database management system adalah :

- **Penyimpanan, pengambilan dan perubahan data**

Sebuah DBMS harus menyediakan kemampuan menyimpan, mengambil dan merubah data dalam basis data.

- **Katalog yang dapat diakses pemakai**

Menyediakan sebuah katalog yang berisi deskripsi item data yang disimpan dan diakses oleh pemakai.

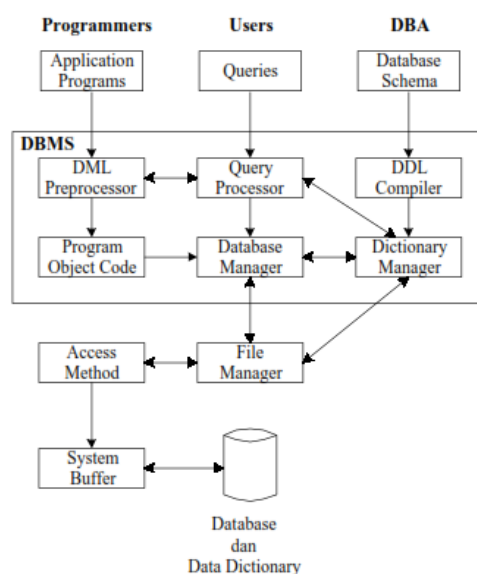
- **Mendukung Transaksi**

Menyediakan mekanisme yang akan menjamin semua perubahan yang berhubungan dengan transaksi yang sudah ada atau yang akan dibuat.



- Melayani kontrol concurrency  
Sebuah DBMS harus menyediakan mekanisme yang menjamin basis data terupdate secara benar pada saat beberapa pemakai melakukan perubahan terhadap basis data yang sama secara bersamaan.
- Melayani recovery  
Menyediakan mekanisme untuk mengembalikan basis data ke keadaan sebelum terjadinya kerusakan pada basis data tersebut.
- Melayani otorisasi  
Sebuah DBMS harus menyediakan mekanisme untuk menjamin bahwa hanya pemakai yang berwenang saja yang dapat mengakses basis data.
- Mendukung komunikasi data  
Sebuah DBMS harus mampu terintegrasi dengan software komunikasi.
- Melayani integrity  
Sebuah DBMS bertujuan untuk menjamin semua data dalam basis data dan setiap terjadi perubahan data harus sesuai dengan aturan yang berlaku.
- Melayani data independence  
Sebuah DBMS harus mencakup fasilitas untuk mendukung kemandirian program dari struktur basis data yang sesungguhnya.
- Melayani utility  
Sebuah DBMS sebaiknya menyediakan kumpulan layanan utility.

## Komponen DBMS



- **Query Processor**  
Komponen yang merubah bentuk query ke dalam instruksi tingkat rendah ke database manager
- **Database Manager**  
Database manager menerima query dan menguji skema eksternal dan konseptual untuk menentukan apakah record-record dibutuhkan untuk memenuhi permintaan. Kemudian DM memanggil file manager untuk menyelesaikan permintaan
- **File Manager**  
Memanipulasi penyimpanan file dan mengatur alokasi ruang penyimpanan pada disk.
- **DML Preprocessor**  
Modul yang merubah perintah DML embedded ke dalam program aplikasi dalam bentuk fungsi-fungsi yang memanggil dalam host language.
- **DDL Compiler**  
Merubah perintah DDL menjadi kumpulan tabel yang berisi metadata.
- **Dictionary Manager**  
Mengatur akses dan memelihara data dictionary. Data dictionary diakses oleh komponen DBMS yang lain.

#### **REFERENSI/DAFTAR PUSTAKA**

1. Juman, Kundang K. *Teknik Pencarian Fakta Dalam Perancangan Sistem Informasi*, 2012
2. Elmasri & Navathe. *Fundamental of Database Systems, 5th Edition, Chapter 4*, 2007
3. Bertalya. *MODEL EER (Enhanced Entity Relationship)*. 2008
4. Korth, H & Mc Graw Hill. *Database System Concept, 4th edition*. New York