By Muggy David

**Introduction to Data and Database**

1. Data

Informasi berupa fakta atau angka (berupa absolut), dikumpulkan untuk diolah untuk membantu membuat decision-making atau informasi dalam bentuk elektronik yang bisa disimpan dan digunakan dalam komputer.

Format: text, spreadsheet, image, audio, video, etc.

1. Categorical data

* Nominal

Sesuatu yang bisa diklasifikasi (gender, name, cities)l

* Ordinal

Klasifikasi yang berbobot (seperti contoh “kelas: ada SD, SMP, SMA” atau “Ranking: ada rank 1, rank 2, rank 3, dst”)

1. Numerical data

* Discret (count/sesuatua yang dijumlah): cth: jumlah anak
* Continuous (measurement: interval: tidak memiliki titik “0” yang absolut [cth: temperature memiliki titik “0” yang berbeda2] atau rasio: memiliki titik “0” yang absolut. [tinggi/berat])

* Raw Data: bisa didapatkan dari survei/observasi/bahkan megenarate data sendiri.
* Informasi: Raw data tersebut bisa diberikan konteks (bisnis/teknikal) terlebih dahulu, sehingga bisa menjadi informasi.
* Knowledge: diberi meaning shingga menjadi pengetahuan.
* Wisdom: diberikan insight, knowledge ini bisa menjadi wisdom. Pengetahuan A dan pengetahuan B bisa menjadi satu kesatuan.

1. Database

Kumpulan data yang terorganisir dan diakses secara elektronik.

Keunggulan Database:

* Mengurangi redundansi: database membedakan satu row dengan row lain, sehinggan terdapat id yang berbeda-beda.
* Memiliki aturan yang berlapis-berlapis sehingga integritasnya sangat tinggi. Data sudah pasti konsisten mengikuti apa yang sudah di set di awal (ctoh: kolom tanggal sudah pasti berisi tanggal dari awal hingga akhir)
* Data yang sudah tersentral
* Memiliki penjagaan yang berlapis dan menjaga privasi data. Seperti cth: dalam 1 perusahaan tidak semua orang bisa akses ke semua file. Spt cth, tim marketing tidak bisa akses file finance,
* Kita bisa mengakses gambaran besa, karena semuanya tersentral.
* Sudah tersiapkan backup dan recovernya, tinggal diakukan backup/recovery secara manual atau terjadwal.

1. SQL n DBMS

DBMS: Database Management System = software yang di desain untuk store, retrieve, define,

SQL: Structured Query Language = bahasa pemograman yang didesain untuk memanage data yang terdapat di DBMS

DDL: Data Definition Language =

* Database > Schema > Table > Kolom
* Perintah SQL untuk mendefine Schema
* Kita bisa CREATE, DROP, ALTER, TRUNCATE, COMMENT, RENAME dalam Schema, perintah ini bisa kita lakukan ke file SQLnya

DML:Data Manipulation Language =

* Digunakan untuk memanipulasi
* SELECT, INSERT, UPDATE, DELETE

Database (Structured)

Pro:

* Lebih mudah implementasi di Machine learning
* Lebih banyak digunakan oleh orang bisnis/end-user, datanya sudah bisa dipakai

Cons:

* Hanya ke ralational database only

Tools:

* MySQOL, MS SQL,

Database (Unstructured)

Menyimpan sesuatu yang sifatnya bukan tabular (seperti menyimpan audio, gambar, atau video)

Pro:

* Waktu proses yang lebih cepat
* Lebih hemat storage dalam data lake

Cons:

* Memerlukan skill yang lebih teknikal
* Memerlukan tools yang lebih spesifik

Tools:

* MongoDB, DynamoDB, Hadoop, etc

1. Data Types

Data types bisa berbeda-beda tergantung DBMS yang dipilih.

Numerik: angka, bisa dilakukan aritmatika (cth: int dikali float. Tetapi harus hati-hati saat pembagian, cth: int:int bisa jadi float, 5 /2=2.5, hasilnya akan dibaca 2, o.5 akan dipaksa hilang)

* Int
* Bigint
* Smallint
* Float: desimal

Date/Time: tanggal/waktu

Char:

* Char
* Varchar
* text

Cth kasus:

Barcode/plat nomor berisi dari kata dan angka.

Cth: plat nomor B 2562 UKV => terdiri dari string dan int.

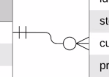
Tidak perlu bingung, plat nomor bisa kita paksa jadi string saja.

Database - Data Models

Relationship Cardinality:

* One to one (1:!):
* One to many (1:M) / Many to one
* Many to many (M:N)

Participan Constraint

 = Constrain

Mandatory:

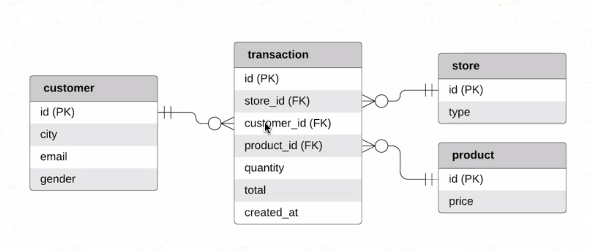
* Data customer sudah pasti ada di transaction, tapi di customer belum tentu melakukan transaksi

Optional:

* Alks

Entity Types:

* Strong: bisa berdiri sendiri
* Weak: tidak bisa berdiri sendiri
* Associate: dibuat oleh entitas lain



**PK (Primary Key)**: ID yang mebedakan satu row dengan row lain, ketika PK satu dipindahkan ke tabel lain, akan berubah menjadi **FK (Foreign Key)**. ID digunkan untuk menghubungkan antar tabel.

Untuk mengefisiensikan database dilakukan normalisasi.

1. Data “lifecycle”

Data itu prosesnya benar2 tertata

1. Generation
2. Collection
3. Processing: data cleansing (jika dibutuhkan)
4. Storage: data digudangkan/di warehouse atau di data lake.
5. Management: data dimanage / didistribusikan ke untuk reporting, untuk machine learning.
6. Analysis: data di analisis
7. Visualization: data di visualisasikan untuk mempresentasikan hasilnya.
8. Interpretation hasilnya di interpreatsi/diterapkan ke bisnis/ke internal perusahaan

Data lifecycle tidak akan berhenti, jadi ketika sudah di tahap ke-8 ada kmungkinan untuk kembali ke tahap ke-1.

1. Google Collabs Environment
2. DBeaver Installment