

# Introduction to Data and Database

20 Feb 2024



# Agenda

1. What is Data?
2. What is Database?
3. What is SQL and DBMS?
4. What is data types?
5. Data ""Lifecycle""
6. Google Collabs environment
7. Intro DuckDB / Google BigQuery / DBeaver (PostgreSQL)

# Pengertian data

da.ta

→ Tesaurus

1. *n* keterangan yang benar dan nyata: *pengumpulan -- untuk memperoleh keterangan tentang kehidupan petani*
2. *n* keterangan atau bahan nyata yang dapat dijadikan dasar kajian (analisis atau kesimpulan)
3. *n Komp* informasi dalam bentuk yang dapat diproses oleh komputer, seperti representasi digital dari teks, angka, gambar grafis, atau suara

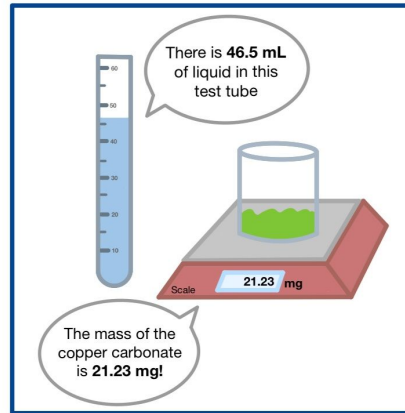
- menurut The Oxford Learner's Dictionary (2021) data adalah *"facts or information, especially when examined and used to find out things or to make decisions."*
- Data dapat berupa teks, pengamatan, angka, gambar, nomor, grafik, atau simbol.
- Sebagai contoh, data mungkin mencakup harga individu, berat, alamat, usia, nama, suhu, tanggal, atau jarak.

# Klasifikasi Data

- Data dapat berupa;
  - Data **kualitatif (qualitative)** adalah informasi deskriptif (itu mendeskripsikan sesuatu)
  - Data **kuantitatif (quantitative)** adalah informasi numerik (angka)

## Quantitative Data

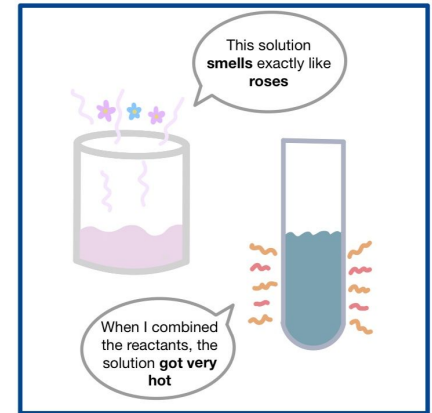
Based on numbers (measurement or counting)



VS

## Qualitative Data

Based on observations (involves the senses)

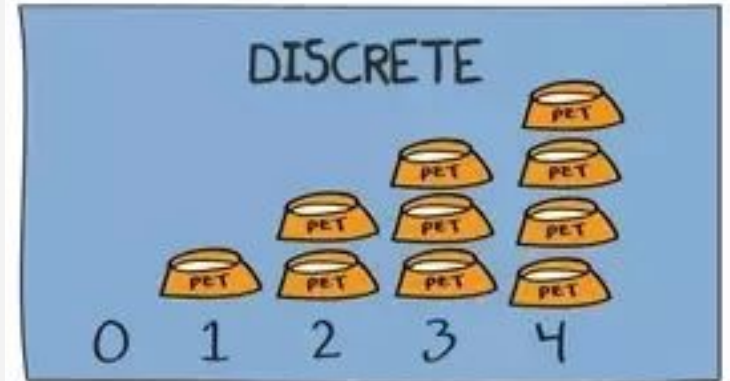


# Klasifikasi Data

Kriteria	Data Kualitatif	Data Kuantitatif
Definisi	Data yang menggambarkan karakteristik atau sifat.	Data yang menggambarkan jumlah, ukuran, atau hitungan.
Jenis Informasi	Deskriptif, tidak dapat dihitung.	Numerik, dapat dihitung.
Contoh	Warna, tekstur, aroma, opini, dan preferensi.	Tinggi, berat, umur, jumlah, harga, dan skor.
Tujuan	Memahami konsep, ide, atau pengalaman.	Mengukur atau menghitung fenomena.
Alat Pengumpulan Data	Wawancara, observasi, analisis dokumen.	Survei dengan skala numerik, pengukuran fisik.
Sifat Data	Subjektif, berdasarkan interpretasi.	Objektif, berdasarkan pengukuran.

# Klasifikasi Data

- Data **kuantitatif** dapat berupa:
  - Data **Diskrit** hanya dapat mengambil nilai tertentu (seperti bilangan bulat)
  - Data **Kontinu** dapat mengambil nilai berapapun (dalam suatu rentang)



# Klasifikasi Data

Kriteria	Data Diskrit	Data Kontinu
Definisi	Data yang hanya dapat mengambil nilai tertentu dan terpisah.	Data yang dapat mengambil nilai apapun dalam suatu rentang atau interval.
Nilai	Hanya bisa berupa bilangan bulat.	Bisa berupa bilangan bulat atau pecahan.
Contoh	Jumlah buku dalam sebuah tas, jumlah siswa dalam kelas.	Berat, tinggi, atau suhu yang dapat memiliki nilai pecahan.
Pengukuran	Tidak bisa dibagi lebih lanjut atau dipecahkan menjadi bagian yang lebih kecil tanpa kehilangan makna.	Dapat dibagi atau dipecahkan menjadi unit yang lebih kecil, sering kali tanpa batas.
Penggunaan	Cocok untuk menghitung jumlah dan melakukan analisis statistik yang berhubungan dengan frekuensi.	Cocok untuk pengukuran dan perhitungan yang memerlukan presisi tinggi.
Metode Analisis	Analisis frekuensi, modus, median.	Analisis varians, rata-rata, deviasi standar.
Sifat	Kuantitatif tapi dengan batasan jumlah nilai yang dapat diambil.	Kuantitatif tanpa batasan jumlah nilai yang dapat diambil.

# Data → Informasi

- Informasi adalah **data yang diproses atau diorganisir** dalam bentuk yang memberikan arti atau **nilai bagi penerima**. Informasi membantu dalam **pengambilan keputusan** dengan menyediakan konteks atau pemahaman yang lebih dalam mengenai suatu topik atau situasi.
- Data vs Informasi:
  - **Asal:** Data adalah titik awal yang belum diproses; informasi adalah hasil dari data yang diproses.
  - **Fungsi:** Data sendiri sering kali tidak cukup untuk pengambilan keputusan; informasi memberikan konteks dan pemahaman yang diperlukan untuk keputusan.
  - **Nilai:** Data bisa jadi tidak memiliki konteks atau arti tanpa analisis; informasi memberikan arti dan dapat langsung dimanfaatkan.



# Database

- Database adalah kumpulan **data yang tersimpan dalam sistem komputer**. Data tersebut terorganisir sedemikian rupa sehingga memudahkan dalam pencarian, pengambilan, dan pengelolaan informasi.
- Karakteristik Database:
  - **Terstruktur**: Data disimpan dalam tabel yang terorganisir, yang memudahkan dalam pengelolaan dan pengaksesan data.
  - **Dinamis**: Database dirancang untuk mudah diperbarui, dengan kemampuan untuk menambah, menghapus, atau mengubah data yang tersimpan.
  - **Dapat Diakses Secara Bersamaan**: Mendukung penggunaan oleh banyak pengguna sekaligus tanpa mengganggu integritas data.

# Database

- Mengapa Database Penting?
  - Efisiensi Operasional: Mempercepat proses bisnis dengan menyediakan akses cepat ke data yang dibutuhkan.
  - Pengambilan Keputusan: Membantu dalam pengambilan keputusan yang berbasis data melalui analisis data yang tersimpan.
  - Integritas Data: Menjaga konsistensi dan akurasi data melalui aturan dan batasan yang didefinisikan dalam database.
- Contoh Penggunaan Database:
  - Bisnis: Pengelolaan inventori, data pelanggan, dan transaksi penjualan.
  - Pendidikan: Menyimpan informasi siswa, nilai, dan kurikulum.
  - Kesehatan: Pengelolaan rekam medis pasien dan penjadwalan jadwal dokter.

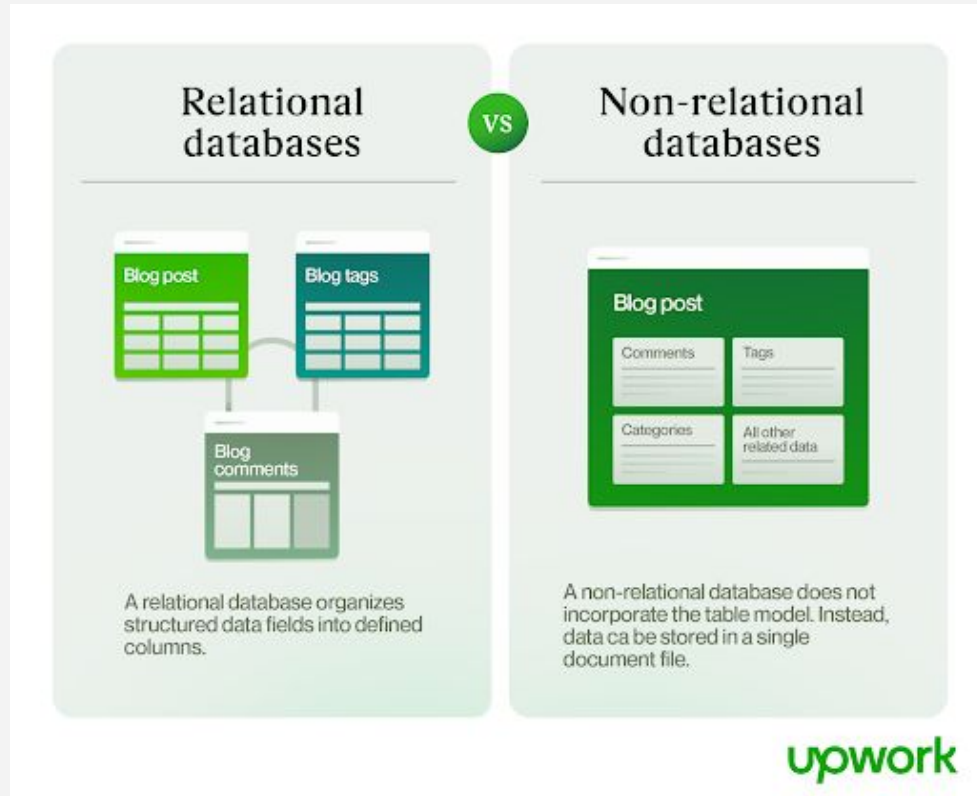
# Database Management System (DBMS)

- DBMS adalah **perangkat lunak yang digunakan untuk mengakses database**, serta untuk membuat, memproses, dan mengelola data di dalamnya. DBMS bertindak sebagai antarmuka antara database dan pengguna atau aplikasi lainnya.
- Fokus: Pada penyediaan **alat untuk mengelola dan mengakses data**. DBMS memungkinkan pengguna untuk membuat, membaca, memperbarui, dan menghapus data dalam database menggunakan query SQL atau antarmuka grafis.
- Contoh: MySQL, Oracle, Microsoft SQL Server, PostgreSQL, dll.
-

# Database vs DBMS

- **Peran dan Fungsi:** Database adalah tempat penyimpanan data, sedangkan DBMS adalah perangkat lunak yang memfasilitasi interaksi dengan data tersebut.
- **Interaksi:** Pengguna atau aplikasi tidak berinteraksi langsung dengan database; mereka menggunakan DBMS sebagai perantara untuk mengakses dan mengelola data.
- **Fitur:** DBMS menyediakan fitur seperti kontrol akses, manajemen transaksi, pencarian dan pengambilan data, serta dukungan untuk bahasa query seperti SQL. Database sendiri tidak memiliki fungsionalitas ini tanpa DBMS.
- **Keamanan dan Integritas Data:** DBMS menyediakan mekanisme untuk memastikan keamanan dan integritas data, termasuk kontrol akses, backup, dan pemulihan data. Database menyimpan data tetapi tidak secara langsung menangani aspek keamanan ini.

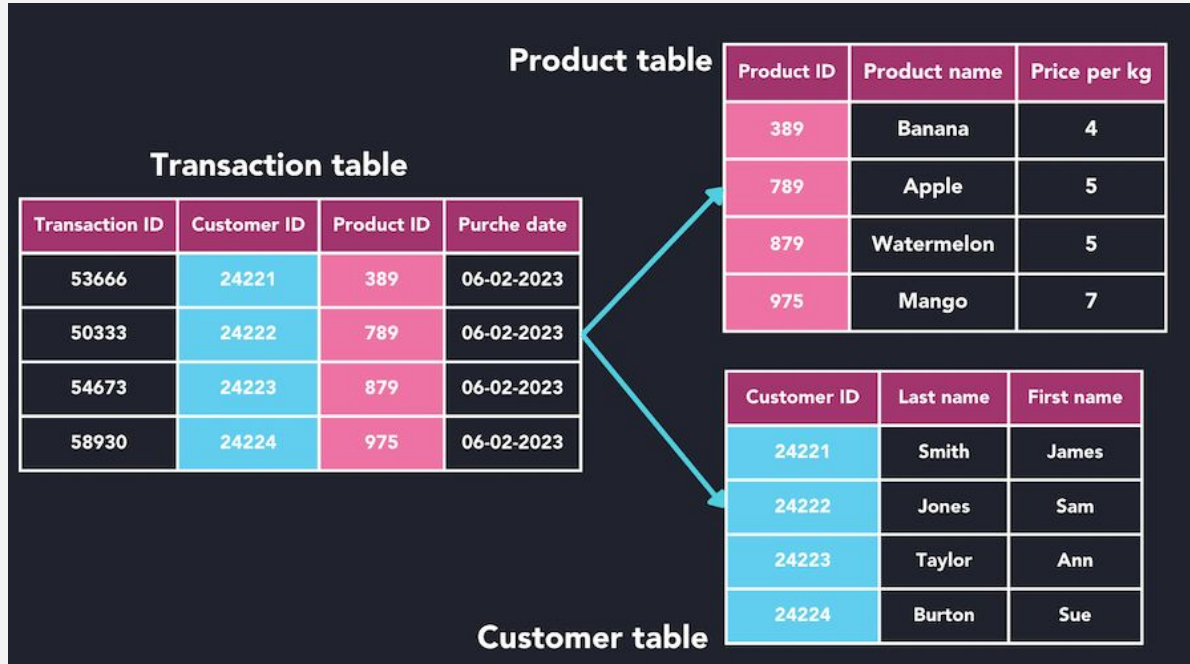
# Type Database



# Tipe Database : Relasional

- Menggunakan tabel untuk menyimpan data yang saling terkait.
- **Ciri Khas:**
  - Struktur data tabel dengan baris dan kolom.
  - Penggunaan *foreign key* untuk hubungan antar tabel.
- **Bahasa Query:** SQL (Structured Query Language).
- **Contoh:** MySQL, PostgreSQL, Oracle, Microsoft SQL Server.
- **Keunggulan:** Struktur yang ketat memudahkan pemodelan hubungan antar data.
- **Kekurangan:** Skalabilitas bisa menjadi tantangan untuk dataset yang sangat besar atau untuk aplikasi real-time.

# Type Database : Relasional



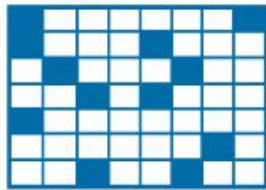
# Tipe Database : Non - Relasional

- Dirancang untuk penyimpanan data yang lebih fleksibel dan skalabilitas horizontal.
- **Ciri Khas:**
  - Mendukung berbagai model data seperti dokumen, kunci-nilai, wide-column, dan graf.
  - Tidak memerlukan skema tetap sebelum menyimpan data.
- **Contoh:** MongoDB (Dokumen), Redis (Kunci-Nilai), Cassandra (Wide-Column), Neo4j (Graf).
- **Keunggulan:** Fleksibilitas dan skalabilitas tinggi.
- **Kekurangan:** Kompleksitas dalam manajemen dan konsistensi data.

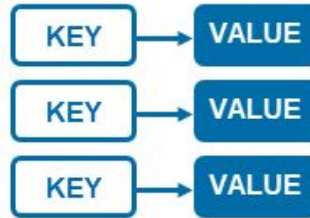


# Type Database : Non - Relasional

## Non-Relational Database Types



Column based



Key-value



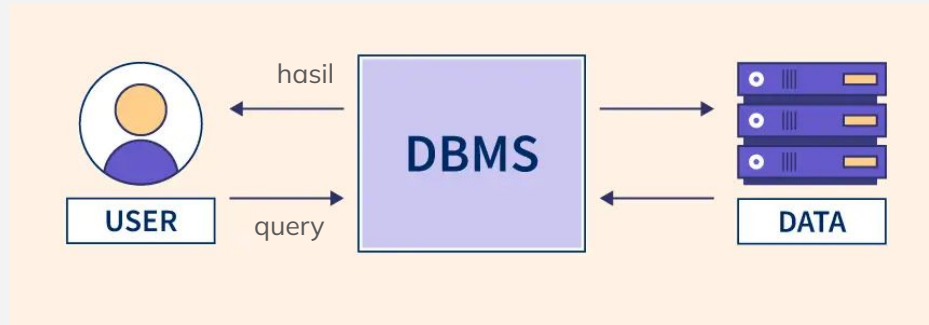
Graph



Document

# Pengertian Query

- Sebuah cara untuk mengambil atau memodifikasi data dari database menggunakan perintah tertentu.
- **Tujuan:** Memungkinkan pengguna atau aplikasi untuk berinteraksi dengan database untuk melakukan tugas-tugas seperti pencarian, pembaruan, penghapusan, dan penyisipan data.
- Query dieksekusi oleh Database Management System (DBMS) yang kemudian menginterpretasikan perintah dan mengambil atau memodifikasi data sesuai permintaan.



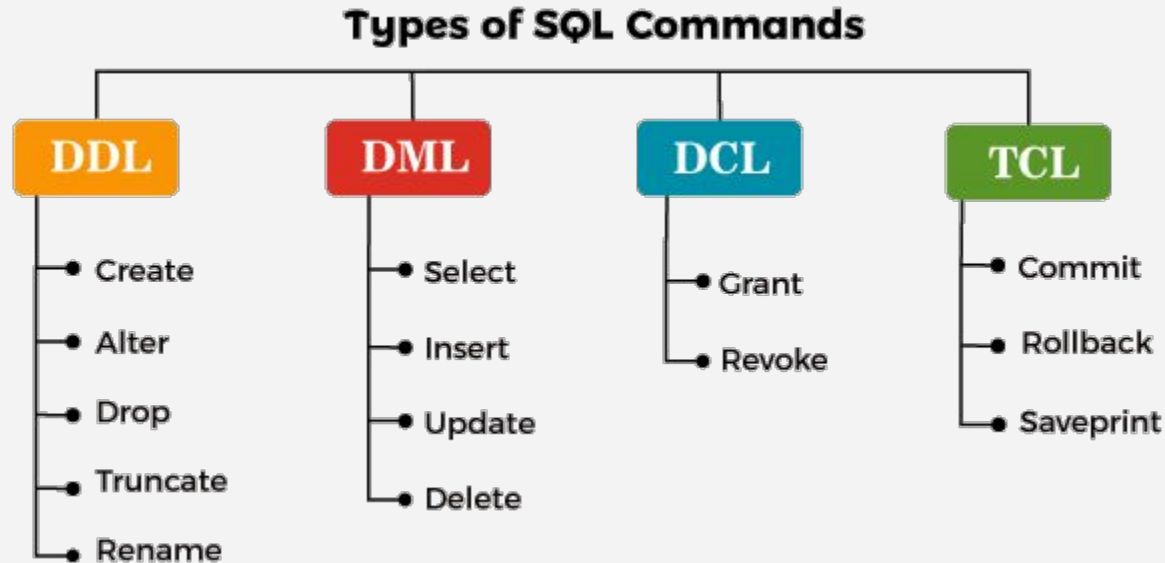
# Bahasa Query Umum: SQL

- SQL (Structured Query Language): Bahasa standar untuk melakukan query pada database relasional.
- Contoh Perintah SQL: SELECT, INSERT, UPDATE, DELETE.
- Contoh Query: `SELECT * FROM users WHERE age > 18;`

# Jenis Query

- DDL (Data Definition Language):
  - Fungsi: Mendefinisikan, mengubah, dan menghapus struktur data dan skema.
  - Contoh: CREATE, ALTER, DROP.
- DML (Data Manipulation Language):
  - Fungsi: Memanipulasi data dalam database (menambah, mengupdate, menghapus data).
  - Contoh: SELECT, INSERT, UPDATE, DELETE.
- DCL (Data Control Language):
  - Fungsi: Mengatur hak akses pengguna terhadap data.
  - Contoh: GRANT, REVOKE.
- TCL (Transaction Control Language):
  - Fungsi: Mengelola transaksi dalam database, seperti memulai, menggulung balik, dan mengkonfirmasi transaksi.
  - Contoh: COMMIT, ROLLBACK, SAVEPOINT.

# Bahasa Query Umum: SQL



# Tipe Data

- Tipe data dalam database menentukan jenis nilai yang dapat disimpan dalam kolom.
- Pemilihan tipe data yang tepat esensial untuk efisiensi penyimpanan, kecepatan akses, dan integritas data.
- Definisi kolom tabel:
  - Nama kolom
  - Tipe data dari kolom
- contoh: **student\_name** dan tipe datanya **char(50)** yang berarti dapat menyimpan string karakter hingga 50 karakter.
- Tipe data terutama diklasifikasikan menjadi tiga kategori untuk setiap database.
  - Tipe Data String
  - Tipe Data Numerik
  - Tipe Data Tanggal dan Waktu

# Tipe Data : String

- CHAR: Panjang tetap dengan maksimum 255 karakter.
- VARCHAR: Panjang variabel dengan maksimum 65,535 karakter.
- TEXT: Untuk teks panjang dengan maksimum 65,535 karakter (TEXT), 16,777,215 karakter (MEDIUMTEXT), atau 4,294,967,295 karakter (LONGTEXT).
- BINARY dan VARBINARY: Mirip dengan CHAR dan VARCHAR, tetapi untuk data biner.
- BLOB: Untuk data biner besar dengan ukuran maksimum yang sama dengan TEXT.

<https://dev.mysql.com/doc/refman/8.0/en/data-types.html>

# Tipe Data : Numerik

- INT: Bilangan bulat, dengan atau tanpa tanda.
- DECIMAL (M,D): Bilangan desimal tetap, cocok untuk nilai keuangan.
- FLOAT dan DOUBLE: Bilangan desimal dengan presisi tunggal dan ganda.
- BIT: Menyimpan nilai biner (0 atau 1).

<https://dev.mysql.com/doc/refman/8.0/en/data-types.html>

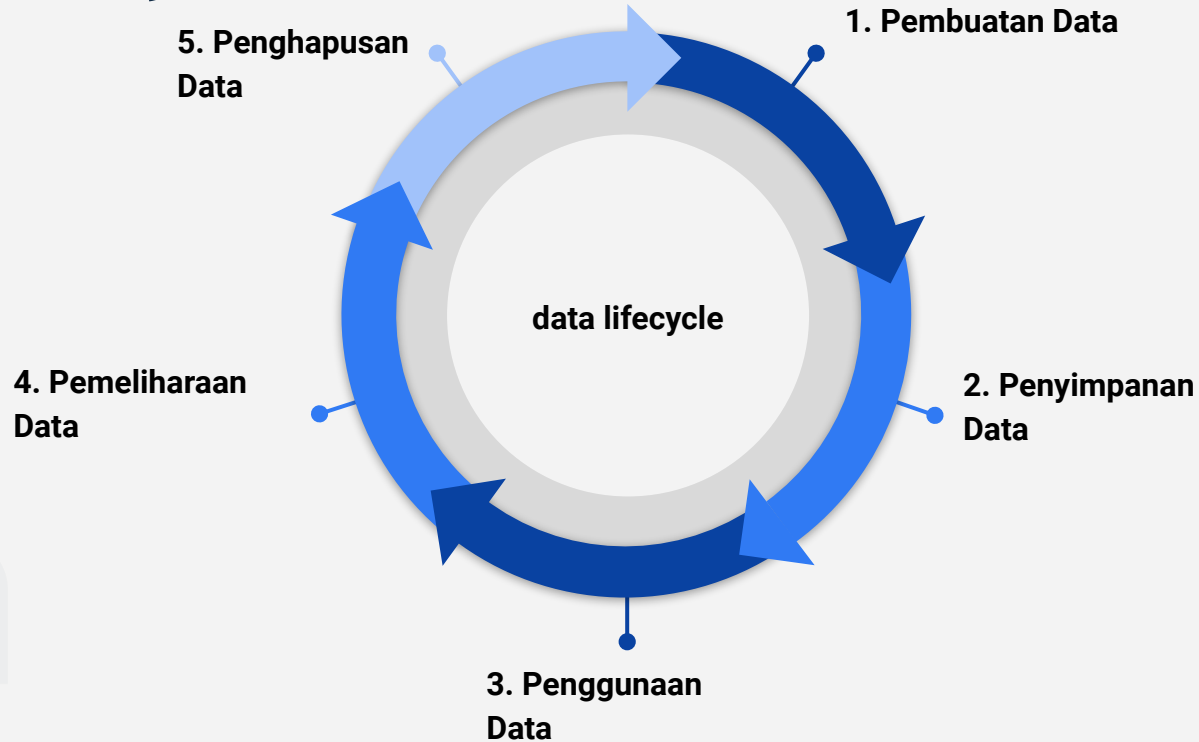


# Tipe Data : Numerik

- DATE: Format tanggal (YYYY-MM-DD).
- TIME: Format waktu (HH:MM:SS).
- DATETIME: Gabungan tanggal dan waktu (YYYY-MM-DD HH:MM:SS).
- TIMESTAMP: Tipe data yang mirip dengan DATETIME, tetapi digunakan untuk merekam waktu suatu baris ditambahkan atau diubah.
- YEAR: Menyimpan tahun.

<https://dev.mysql.com/doc/refman/8.0/en/data-types.html>

# Data Lifecycle



# Data Lifecycle

- Tahap 1 - Pembuatan Data
  - Data dihasilkan melalui berbagai cara, seperti input pengguna, transaksi bisnis, atau pengumpulan data sensor.
  - Penting untuk menentukan struktur dan tipe data pada tahap ini.
- Tahap 2 - Penyimpanan Data
  - Data disimpan dalam sistem penyimpanan seperti database, file system, atau cloud storage.
  - Penentuan lokasi dan format penyimpanan mempengaruhi aksesibilitas dan keamanan data.
- Tahap 3 - Penggunaan Data
  - Data digunakan untuk analisis, laporan, atau sebagai dasar pengambilan keputusan.
  - Penggunaan data yang efisien memerlukan tools analitik dan sistem manajemen data yang baik.

# Data Lifecycle

- Tahap 4 - Pemeliharaan Data
  - Meliputi proses backup, pembaruan, dan pembersihan data untuk menjaga kualitas dan relevansi data.
  - Pemeliharaan yang rutin membantu mengurangi redundansi dan memastikan integritas data.
- Tahap 5 - Arsip atau Penghapusan Data
  - Data yang tidak lagi relevan atau telah melewati masa retensi ditentukan diarsipkan atau dihapus.
  - Penghapusan atau pengarsipan harus dilakukan dengan cara yang aman untuk mencegah kebocoran informasi.

**Tools:**  
**Google Colab**  
**<https://colab.google/>**

# Google Colab

- Google Colab, atau Google Colaboratory, adalah sebuah layanan gratis dari Google yang memungkinkan siapa saja untuk menulis dan menjalankan kode Python melalui browser. Ini sangat berguna untuk pembelajaran mesin, analisis data, dan pendidikan.



# Google Colab Fitur

- Lingkungan Jupyter notebook berbasis cloud tanpa instalasi lokal.
- Akses gratis ke GPU dan TPU untuk komputasi cepat.
- Kolaborasi real-time mirip Google Docs.
- Integrasi dengan Google Drive untuk penyimpanan mudah.
- Dukungan untuk banyak library Python populer.

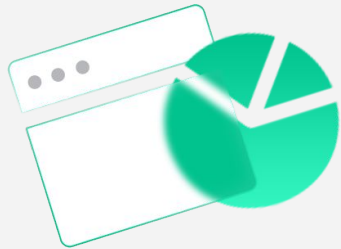
# **Tools:** **DBeaver DBMS**



# DBeaver

- DBeaver Community adalah alat database lintas platform gratis untuk pengembang, administrator basis data, analis, dan semua orang yang bekerja dengan data. Ini mendukung semua basis data SQL populer seperti MySQL, MariaDB, PostgreSQL, SQLite, dan lainnya.





# Terima Kasih

