

Teori Dasar Database

- Pengertian Database
- Sejarah Database
- Kategori Database
- Database Relasional (RDMS)
- Database Non-Relasional (NoSQL)
- Database Object-Oriented
- Struktur Data dalam Database
- Operasi dalam RDMS
- Profesi yang Berhubungan dengan Database
- DMS (Data Migration Service)

Pengertian Database

Database adalah sebuah sistem yang dirancang untuk menyimpan dan mengelola data dengan cara yang terstruktur, sehingga memungkinkan pengguna untuk menyimpan, mengambil, dan memanipulasi data dengan efisien. Secara umum, database digunakan untuk menyimpan informasi dalam jumlah besar secara terorganisir agar dapat diakses dan dikelola dengan mudah.

Beberapa konsep dasar dalam database meliputi:

1. **Data:** Informasi yang disimpan dalam database, seperti nama, alamat, atau transaksi.
2. **Database Management System (DBMS):** Perangkat lunak yang mengelola database, memungkinkan pengguna untuk menyimpan, mengambil, dan memanipulasi data. Contoh DBMS termasuk MySQL, PostgreSQL, Oracle, dan Microsoft SQL Server.
3. **Tabel:** Struktur dasar penyimpanan data dalam database yang terdiri dari baris dan kolom. Setiap kolom mewakili jenis data yang berbeda, sedangkan setiap baris mewakili satu set data yang lengkap.
4. **Query:** Perintah yang digunakan untuk mengambil atau memanipulasi data dari database, misalnya dengan menggunakan SQL (Structured Query Language).
5. **Relasi (Relational):** Model basis data yang paling umum, di mana data disusun dalam tabel yang saling terkait melalui kunci relasional, seperti dalam basis data SQL.

Database memiliki peran penting dalam berbagai aplikasi teknologi informasi modern, termasuk sistem manajemen perusahaan, situs web, aplikasi mobile, dan sistem informasi lainnya, karena menyediakan cara yang efisien untuk menyimpan dan mengelola informasi yang diperlukan untuk operasi bisnis dan organisasi.

Sejarah Database

Sejarah database melibatkan perkembangan konsep, teknologi, dan praktik dalam pengelolaan dan penyimpanan data. Berikut adalah gambaran umum tentang perkembangan sejarah database:

1. Model Awal (1960-an - 1970-an):

- Pada awal 1960-an, konsep dasar tentang basis data mulai muncul. Pada saat itu, sistem basis data umumnya terdiri dari file-file data yang diatur dalam hierarki atau jaringan.
- Pada pertengahan 1960-an, model data hierarki dikembangkan, diikuti oleh model jaringan yang lebih fleksibel. Contoh dari ini adalah sistem seperti IMS (Information Management System) dari IBM.

2. Model Relasional (1970-an):

- Pada tahun 1970, Edgar F. Codd mengusulkan model data relasional sebagai alternatif yang lebih bersih dan lebih terstruktur daripada model sebelumnya. Model ini didasarkan dirinya pada teori himpunan matematika dan menggunakan tabel (relasi) untuk menyimpan data.
- Pada tahun 1970-an, IBM memperkenalkan System R dan Oracle memperkenalkan produk basis data relasional pertama mereka. Ini adalah awal dari apa yang kemudian menjadi Sistem Manajemen Basis Data Relasional (RDBMS) modern.

3. Perkembangan SQL (1970-an - 1980-an):

- SQL (Structured Query Language) dikembangkan pada tahun 1970-an sebagai bahasa standar untuk mengakses dan mengelola data dalam basis data relasional.
- SQL menjadi bahasa yang dominan digunakan untuk kueri dan administrasi basis data relasional, dengan standar yang diadopsi oleh banyak vendor RDBMS.

4. Komersialisasi dan Proliferasi (1980-an - 1990-an):

- Pada tahun 1980-an, industri basis data mengalami pertumbuhan yang signifikan dengan munculnya banyak vendor RDBMS baru seperti Sybase, Informix, Microsoft SQL Server, dan PostgreSQL.
- Basis data relasional menjadi populer di berbagai industri untuk mengelola data transaksional dan analitis.

5. Internet dan Perkembangan Aplikasi Web (1990-an - 2000-an):

- Pada tahun 1990-an, ekspansi internet memicu permintaan akan sistem basis data yang dapat menangani volume data besar dan mendukung akses global.
- MySQL menjadi populer sebagai solusi basis data open-source yang cocok untuk aplikasi web dan e-commerce.

6. Perkembangan Big Data dan NoSQL (2000-an - Sekarang):

- Pada awal abad ke-21, ledakan data dari aplikasi web, media sosial, sensor, dan perangkat IoT memunculkan tantangan baru dalam pengelolaan dan analisis data.
- Basis data NoSQL (Not Only SQL) muncul sebagai alternatif untuk skenario di mana basis data relasional tradisional tidak efektif, memungkinkan skema yang fleksibel, replikasi yang mudah, dan skalabilitas horizontal.

7. **Cloud Computing dan Database as a Service (DaaS):**

- Munculnya cloud computing telah mengubah cara organisasi menyimpan, mengelola, dan mengakses data. Layanan basis data cloud (DaaS) seperti Amazon RDS, Google Cloud SQL, dan Azure SQL Database menawarkan solusi yang fleksibel dan dapat diskalakan untuk berbagai aplikasi.

Sejak munculnya model data relasional, database terus berkembang untuk mengatasi tantangan baru dalam pengelolaan data, termasuk skala yang besar, analisis real-time, keamanan data, dan integrasi dengan teknologi baru seperti AI dan machine learning. Perkembangan ini menunjukkan bahwa basis data terus menjadi bagian integral dari infrastruktur teknologi informasi modern.

Kategori Database

Database dapat dikelompokkan berdasarkan berbagai kriteria, seperti model data, cara penyimpanan, penggunaan, dan arsitektur distribusi. Berikut adalah beberapa pengelompokan umum database berdasarkan berbagai kriteria:

1. Berdasarkan Model Data:

- **Relasional:** Database yang menggunakan model relasional dengan tabel, baris, dan kolom.
- **Non-Relasional (NoSQL):** Database yang tidak mematuhi struktur relasional tradisional, seperti dokumen, grafik, key-value pairs, dan lain-lain.
- **Object-Oriented:** Database yang menyimpan data dalam bentuk objek yang sesuai dengan paradigma pemrograman berorientasi objek.

2. Berdasarkan Cara Penyimpanan:

- **Disk-based:** Database yang menyimpan data pada media penyimpanan seperti hard disk.
- **In-Memory:** Database yang menyimpan data dalam memori utama komputer untuk meningkatkan kinerja akses data.

3. Berdasarkan Penggunaan dan Tujuan:

- **Operational Database:** Database yang digunakan untuk mendukung operasi sehari-hari aplikasi dan sistem bisnis.
- **Data Warehouse:** Database yang digunakan untuk analisis dan pelaporan data dari berbagai sumber data operasional.
- **Time Series Database:** Database yang khusus dirancang untuk menyimpan dan mengelola data berdasarkan waktu.

4. Berdasarkan Arsitektur Distribusi:

- **Centralized Database:** Database yang terpusat di satu lokasi.
- **Distributed Database:** Database di mana data disimpan di beberapa lokasi fisik yang terhubung melalui jaringan komputer.
- **Replicated Database:** Database yang memiliki salinan data di beberapa lokasi untuk meningkatkan ketersediaan dan redundansi.

Setiap pengelompokan database mencerminkan cara mereka menyimpan, mengelola, dan memberikan akses terhadap data. Pemilihan jenis database yang tepat sangat tergantung pada kebutuhan aplikasi, skala penggunaan, dan karakteristik data yang akan disimpan dan diakses.

Database Relasional (RDMS)

Berikut adalah beberapa contoh dari database relasional yang umum digunakan:

1. **MySQL:**

- MySQL adalah sistem manajemen basis data relasional open-source yang sangat populer. MySQL digunakan secara luas untuk berbagai aplikasi web dan bisnis karena kinerjanya yang baik, skalabilitas, dan komunitas yang besar.

2. **PostgreSQL:**

- PostgreSQL adalah DBMS relasional open-source yang kuat dan sangat dapat disesuaikan. PostgreSQL mendukung fitur-fitur tingkat lanjut seperti transaksi, subquery, tipe data kustom, dan replikasi.

3. **Oracle Database:**

- Oracle Database adalah DBMS relasional komersial yang dikenal karena keandalannya, skalabilitasnya yang tinggi, dan berbagai fitur canggih seperti manajemen transaksi yang kuat dan kemampuan untuk mengelola data dalam skala besar.

4. **Microsoft SQL Server:**

- SQL Server adalah DBMS relasional dari Microsoft yang populer di lingkungan perusahaan. SQL Server mendukung berbagai fitur seperti analisis data, manajemen bisnis, dan integrasi dengan alat-alat Microsoft lainnya.

5. **SQLite:**

- SQLite adalah DBMS relasional ringan yang sering digunakan dalam aplikasi berbasis desktop dan mobile. SQLite dirancang untuk kecepatan dan efisiensi serta mudah diintegrasikan ke dalam aplikasi karena tidak memerlukan server terpisah.

6. **MariaDB:**

- MariaDB adalah fork dari MySQL yang dikembangkan oleh komunitas open-source setelah Oracle mengakuisisi MySQL. MariaDB kompatibel dengan MySQL dan menyediakan tambahan fitur serta peningkatan kinerja.

Setiap DBMS relasional ini memiliki karakteristik, keunggulan, dan kelemahan yang berbeda, dan pemilihan yang tepat tergantung pada kebutuhan spesifik aplikasi, skala penggunaan, dan persyaratan bisnis atau teknis lainnya.

Database Non-Relasional (NoSQL)

Berikut adalah beberapa contoh dari database non-relasional (NoSQL database) yang umum digunakan:

1. **MongoDB:**

- MongoDB adalah database dokument yang menggunakan format BSON (Binary JSON) untuk menyimpan data. MongoDB cocok untuk aplikasi yang membutuhkan fleksibilitas skema dan skalabilitas horizontal.

2. **Redis:**

- Redis adalah database key-value in-memory yang mendukung berbagai struktur data seperti string, hashes, lists, sets, dan sorted sets. Redis sering digunakan untuk caching dan memori data yang sering diakses.

3. **Cassandra:**

- Apache Cassandra adalah database kolom wide-column store yang dirancang untuk menangani volume data yang besar dan skalabilitas linier. Cassandra cocok untuk aplikasi dengan kebutuhan data yang besar dan distribusi geografis.

4. **Neo4j:**

- Neo4j adalah database grafik yang dirancang untuk menyimpan data dan relasi di antara entitas sebagai grafik. Neo4j cocok untuk aplikasi yang membutuhkan analisis jaringan sosial, rekomendasi, dan kecerdasan buatan.

5. **Amazon DynamoDB:**

- DynamoDB adalah layanan database NoSQL yang terkelola secara penuh dari AWS. DynamoDB menggunakan model data dokumen dan key-value dan dirancang untuk skala dan kinerja tinggi.

6. **Apache HBase:**

- Apache HBase adalah database kolom wide-column store yang berjalan di atas Hadoop Distributed File System (HDFS). HBase cocok untuk aplikasi yang membutuhkan konsistensi data tinggi dan penyimpanan data terstruktur besar.

7. **Riak:**

- Riak adalah database key-value yang dirancang untuk distribusi dan toleransi terhadap kegagalan. Riak cocok untuk aplikasi yang membutuhkan replikasi data otomatis dan tingkat ketersediaan yang tinggi.

8. **Elasticsearch:**

- Elasticsearch adalah mesin pencari dan analitik terdistribusi yang menyediakan penyimpanan dan pencarian dokumen yang cepat. Elasticsearch cocok untuk aplikasi yang memerlukan pencarian teks penuh dan analisis log.

Setiap jenis database non-relasional memiliki karakteristik, keunggulan, dan kelemahan yang berbeda, dan pemilihan yang tepat tergantung pada kebutuhan aplikasi spesifik, skala data, dan persyaratan kinerja.

Database Object-Oriented

Database yang berbasis objek (object-oriented database) adalah jenis database yang menyimpan data dalam bentuk objek, mirip dengan paradigma pemrograman berorientasi objek. Meskipun tidak sepopuler database relasional atau NoSQL, beberapa contoh database berbasis objek termasuk:

1. **ObjectStore (ObjectStore Database):**

- ObjectStore adalah database yang dirancang khusus untuk menyimpan dan mengakses objek-objek dalam aplikasi yang menggunakan paradigma pemrograman berorientasi objek.

2. **db4o (db4o Database):**

- db4o adalah database objek-objek yang bersifat open-source dan cocok untuk aplikasi-aplikasi yang menggunakan pemrograman berorientasi objek. Db4o menyimpan objek langsung ke dalam database tanpa memerlukan pemetaan relasional.

3. **ObjectDB (Object Database):**

- ObjectDB adalah database objek-objek yang mendukung bahasa pemrograman Java dan menyimpan objek-objek Java secara langsung ke dalam database, tanpa memerlukan konversi atau pemetaan relasional.

Database berbasis objek memiliki kelebihan dalam menyimpan objek-objek kompleks dengan struktur yang rumit, tetapi kurang umum digunakan dibandingkan dengan database relasional atau NoSQL. Pemilihan database berbasis objek tergantung pada kebutuhan aplikasi yang spesifik dan preferensi dalam penggunaan paradigma pemrograman berorientasi objek.

Struktur Data dalam Database

Struktur Data pada RDMS

Struktur data pada database mengacu pada cara data disusun, disimpan, dan diorganisir di dalam sebuah sistem manajemen basis data (DBMS). Struktur data ini penting karena memengaruhi efisiensi operasi database, performa kueri, dan kemampuan untuk menyimpan dan mengambil informasi dengan cepat. Berikut adalah beberapa komponen utama dari struktur data pada database:

1. **Tabel (Table):**

- Tabel adalah struktur utama dalam database yang digunakan untuk menyimpan data.
- Setiap tabel terdiri dari baris (record atau tuple) dan kolom (field atau attribute).
- Kolom-kolom ini mendefinisikan jenis data yang dapat disimpan di dalamnya, misalnya teks, angka, tanggal, atau biner.

2. **Kunci (Key):**

- Kunci adalah atribut atau kombinasi atribut yang memiliki nilai unik untuk setiap baris dalam tabel.
- **Primary Key:** Kunci utama yang unik mengidentifikasi setiap baris secara eksklusif.
- **Foreign Key:** Kunci yang menghubungkan satu tabel dengan tabel lain berdasarkan nilai yang sama.

3. **Indeks (Index):**

- Indeks adalah struktur data tambahan yang digunakan untuk mempercepat pencarian dan pengambilan data dari tabel.
- Indeks dapat dibuat pada satu kolom atau beberapa kolom untuk meningkatkan performa kueri SELECT.

4. **Relasi (Relationship):**

- Relasi mengacu pada hubungan antara tabel dalam basis data relasional.
- Jenis relasi utama meliputi one-to-one, one-to-many, dan many-to-many, yang masing-masing menentukan cara baris dalam satu tabel terhubung dengan baris dalam tabel lain.

5. **Skema (Schema):**

- Skema database mendefinisikan struktur tabel, relasi antartabel, kunci, indeks, dan batasan lainnya.
- Ini mencakup tipe data yang digunakan, batasan integritas referensial, dan aturan lain yang mengatur bagaimana data dapat dimasukkan, diperbarui, dan dihapus.

6. Tipe Data (Data Types):

- Setiap kolom dalam tabel memiliki tipe data yang menentukan jenis nilai yang dapat disimpan di dalamnya.
- Contoh tipe data termasuk VARCHAR (teks berukuran variabel), INT (bilangan bulat), DATE (tanggal), dan BLOB (binary large object).

7. Batasan (Constraints):

- Batasan diterapkan pada kolom atau tabel untuk memastikan keakuratan, konsistensi, dan integritas data.
- Contoh batasan meliputi NOT NULL (kolom tidak boleh kosong), UNIQUE (nilai harus unik), dan FOREIGN KEY (nilai harus ada di tabel lain).

8. View:

- View adalah hasil dari kueri yang terdefinisi secara logis yang disimpan sebagai objek dalam database.
- View memungkinkan pengguna untuk melihat data dari satu atau beberapa tabel dengan cara yang didefinisikan sebelumnya tanpa mengubah struktur dasar tabel.

9. Function:

- Function adalah objek atau program kecil dalam basis data yang dapat menerima parameter input, melakukan operasi tertentu, dan mengembalikan nilai sebagai hasil.
- Function biasanya digunakan untuk menghitung nilai berdasarkan logika tertentu atau untuk melakukan operasi yang dapat dipanggil dari kueri SQL atau bahasa pemrograman.
- Contoh: Anda dapat membuat function untuk menghitung usia berdasarkan tanggal lahir, atau untuk mengembalikan total penjualan dalam bulan tertentu.

10. Procedure:

- Procedure mirip dengan function, tetapi biasanya tidak mengembalikan nilai secara langsung.
- Procedure adalah kumpulan instruksi yang dapat menerima parameter input, melakukan operasi yang kompleks atau rutin, dan mungkin mempengaruhi data dalam tabel.
- Contoh: Anda dapat membuat procedure untuk memperbarui status pengiriman dalam database e-commerce, atau untuk mengirim email konfirmasi kepada pelanggan setelah pesanan diproses.

Struktur data yang baik dalam database adalah kunci untuk mengoptimalkan kinerja database, menjaga konsistensi data, dan menyediakan akses data yang efisien bagi pengguna dan aplikasi.

Jenis Struktur Data

Jenis-jenis struktur data yang umum digunakan dalam konteks penyimpanan dan pengelolaan data dalam basis data meliputi:

1. Tabel (Table):

- Tabel adalah struktur dasar dalam basis data relasional yang terdiri dari baris (record atau tuple) dan kolom (field atau attribute).

2. **Indeks (Index):**

- Indeks digunakan untuk meningkatkan kecepatan pencarian data dalam tabel dengan membuat struktur tambahan yang memetakan nilai kolom ke lokasi fisik di dalam basis data.

3. **B-tree:**

- B-tree adalah jenis struktur pohon pencarian yang sering digunakan untuk mengimplementasikan indeks dalam basis data. B-tree memungkinkan pencarian, penyisipan, dan penghapusan data dengan kompleksitas logaritmik.

4. **Hash Table:**

- Hash table digunakan untuk mengimplementasikan indeks hash di mana kunci (key) diubah menjadi indeks menggunakan fungsi hash. Ini memungkinkan pencarian data dengan kompleksitas waktu konstan dalam kasus terbaik.

5. **Inverted Index:**

- Inverted index adalah struktur data yang digunakan untuk memetakan kata kunci ke dokumen yang mengandung kata kunci tersebut. Ini sering digunakan dalam mesin pencari untuk memungkinkan pencarian teks yang efisien.

6. **Graph Database:**

- Graph database menggunakan struktur data grafik untuk menyimpan entitas dan hubungan antar entitas. Ini cocok untuk data dengan banyak hubungan kompleks, seperti jejaring sosial atau jaringan transportasi.

7. **Trie (Prefix Tree):**

- Trie adalah struktur data tree yang digunakan untuk penyimpanan kata kunci atau kueri teks yang dapat dicari dengan cepat berdasarkan prefix atau awalan kata.

Setiap jenis struktur data ini memiliki kelebihan dan kegunaan yang berbeda tergantung pada jenis data yang disimpan, pola akses data, dan kebutuhan operasional dalam sistem basis data.

Pemilihan struktur data yang tepat sangat penting untuk mengoptimalkan performa, efisiensi, dan skalabilitas sistem basis data.

Operasi dalam RDMS

Operasi-operasi dalam database meliputi berbagai tugas yang dapat dilakukan untuk mengelola dan memanipulasi data dalam basis data. Berikut adalah beberapa operasi utama dalam database:

1. **Create (Buat):**
 - Operasi untuk membuat basis data, tabel, indeks, view, atau prosedur.
2. **Read (Baca):**
 - Operasi untuk membaca atau mengambil data dari tabel atau basis data.
3. **Update (Perbarui):**
 - Operasi untuk mengubah atau memperbarui data yang ada dalam tabel.
4. **Delete (Hapus):**
 - Operasi untuk menghapus data dari tabel.
5. **Insert (Sisipkan):**
 - Operasi untuk menyisipkan data baru ke dalam tabel.
6. **Select (Pilih):**
 - Operasi untuk memilih data dari tabel berdasarkan kriteria tertentu menggunakan pernyataan SQL SELECT.
7. **Join (Gabung):**
 - Operasi untuk menggabungkan data dari dua atau lebih tabel berdasarkan kunci yang sesuai.
8. **Indexing (Pembuatan Indeks):**
 - Operasi untuk membuat indeks pada kolom-kolom tertentu dalam tabel, yang mempercepat proses pencarian data.
9. **Transaction Management (Manajemen Transaksi):**
 - Operasi untuk mengelola transaksi, yaitu kumpulan operasi yang dieksekusi sebagai satu kesatuan yang dapat dijamin untuk berhasil sepenuhnya atau gagal sepenuhnya.
10. **Backup and Restore (Cadangan dan Pulihkan):**
 - Operasi untuk membuat salinan data (backup) dan mengembalikan data dari backup jika diperlukan.
11. **Security Management (Manajemen Keamanan):**
 - Operasi untuk mengatur dan mengelola hak akses pengguna ke basis data, termasuk pengaturan peran, izin, dan audit.
12. **Optimization (Optimasi):**
 - Operasi untuk mengoptimalkan kueri-kueri SQL dan struktur basis data guna meningkatkan kinerja sistem secara keseluruhan.
13. **Replication (Replikasi):**
 - Operasi untuk menyalin dan mendistribusikan data dari satu basis data ke basis data lainnya, sering digunakan untuk meningkatkan ketersediaan dan redundansi data.

Setiap operasi ini merupakan bagian penting dari administrasi dan penggunaan basis data relasional, memungkinkan organisasi untuk mengelola data dengan efisien, konsisten, dan aman.

Profesi yang Berhubungan dengan Database

Beberapa profesi yang erat kaitannya dengan basis data (database) termasuk:

1. **Database Administrator (DBA):**

- Bertanggung jawab untuk merancang, mengelola, dan memelihara sistem basis data. Tugasnya meliputi instalasi perangkat lunak basis data, pemulihan bencana, keamanan data, tuning performa, dan pemeliharaan harian basis data.

2. **Data Engineer:**

- Fokus pada pengembangan dan pemeliharaan infrastruktur data yang kompleks, termasuk penyimpanan data, ETL (Extract, Transform, Load), data pipeline, dan integrasi data antar sistem.

3. **Data Analyst:**

- Mengumpulkan, mengelola, menganalisis, dan menginterpretasikan data untuk membantu pengambilan keputusan bisnis. Biasanya menggunakan kueri SQL dan alat analisis data untuk mengeksplorasi dan memvisualisasikan data.

4. **Data Scientist:**

- Menggabungkan keahlian dalam statistik, analisis data, dan pemrograman untuk mengeksplorasi data besar, menerapkan model prediktif, dan membuat wawasan bisnis dari data.

5. **Database Developer:**

- Spesialis dalam pengembangan aplikasi berbasis database, termasuk desain schema basis data, menulis kueri SQL kompleks, dan mengembangkan stored procedure untuk mengakses dan memanipulasi data.

6. **Database Architect:**

- Bertanggung jawab untuk merancang struktur basis data yang kompleks dan mengembangkan strategi untuk integrasi data, replikasi, skema partisi, dan arsitektur basis data yang scalable.

7. **Database Consultant:**

- Memberikan konsultasi kepada organisasi tentang pemilihan, implementasi, dan pengoptimalan sistem basis data. Mereka juga dapat membantu dalam migrasi data, pemulihan bencana, dan pemecahan masalah basis data.

8. **Database Manager:**

- Bertanggung jawab atas manajemen keseluruhan tim DBA dan strategi pengelolaan basis data di sebuah organisasi. Mereka memastikan kepatuhan terhadap kebijakan keamanan, perencanaan kapasitas, dan koordinasi antar tim teknis.

Setiap profesi ini memainkan peran penting dalam ekosistem teknologi informasi modern, membantu organisasi mengelola, menganalisis, dan menggunakan data secara efektif untuk

mencapai tujuan bisnis mereka.

DMS (Data Migration Service)

Dalam konteks migrasi database, DMS sering mengacu pada **Database Migration Service**. Ini adalah layanan yang disediakan oleh beberapa penyedia cloud seperti AWS (Amazon Web Services) dan Google Cloud untuk membantu organisasi dalam melakukan migrasi data dari sumber database yang ada ke platform cloud mereka. Database Migration Service menawarkan alat dan layanan untuk menangani proses migrasi dengan cara yang aman, efisien, dan minimal mengganggu operasi bisnis yang sedang berjalan.

Beberapa fitur dan kemampuan umum dari Database Migration Service meliputi:

1. **Replication:** Menyalin data dari sumber database ke basis data target di cloud secara berkala untuk menjaga konsistensi data selama proses migrasi.
2. **Schema Conversion:** Mengonversi skema basis data dari sumber ke format yang sesuai dengan basis data target di cloud.
3. **Data Validation:** Memvalidasi data yang telah dimigrasi untuk memastikan integritas dan konsistensi data.
4. **Continuous Data Replication:** Mendukung replikasi data terus menerus untuk meminimalkan waktu tidak aktif selama migrasi.
5. **Monitoring and Troubleshooting Tools:** Menyediakan alat pemantauan dan pemecahan masalah untuk memonitor dan memastikan kelancaran proses migrasi.

Database Migration Service adalah solusi yang umum digunakan untuk memudahkan perusahaan dalam mengadopsi teknologi cloud dengan memfasilitasi migrasi data dari infrastruktur lokal atau data center ke cloud dengan sedikit atau tanpa gangguan operasional.