

1. **DBT (Data Build Tool)** adalah alat sumber terbuka yang digunakan untuk mentransformasi dan mengelola data di dalam warehouse data. DBT dirancang khusus untuk bekerja dengan warehouse data modern seperti Snowflake, BigQuery, dan Redshift. Ini memungkinkan analis data dan insinyur data untuk menulis SQL untuk mendefinisikan transformasi data, mengelola dependensi antara transformasi, dan menjalankan transformasi secara otomatis.
2. **Keuntungan menggunakan DBT:**
 - **Keterbacaan:** DBT menggunakan SQL sebagai bahasa utamanya, yang sudah umum dikenal oleh sebagian besar analis data dan insinyur data.
 - **Pemodelan yang Mudah:** DBT memisahkan antara logika transformasi dan struktur data, yang memungkinkan pengguna untuk memodelkan data dengan lebih mudah dan intuitif.
 - **Pengelolaan Dependensi:** DBT secara otomatis mengelola dependensi antara transformasi, memastikan bahwa transformasi dieksekusi dalam urutan yang benar.
 - **Pelacakan Versi:** DBT menyediakan alat untuk melacak versi transformasi data, memungkinkan pengguna untuk mengetahui dan mengelola perubahan yang dilakukan pada transformasi data dari waktu ke waktu.
 - **Kemudahan Kolaborasi:** DBT mendukung kolaborasi antar tim dengan menyediakan alat untuk mengelola dan berbagi kode transformasi data.
3. **Dependency Tree:** Di DBT, setiap transformasi data disusun dalam bentuk graph dependensi, yang dikenal sebagai "dependency tree". Dependency tree ini menunjukkan hubungan antara setiap tabel hasil transformasi dengan tabel-tabel yang menjadi inputnya. Hal ini memungkinkan DBT untuk menjalankan transformasi data secara otomatis dalam urutan yang benar, yaitu memastikan bahwa tabel-tabel yang diperlukan sebagai input untuk suatu transformasi telah tersedia sebelum transformasi itu dieksekusi.
Versioning: DBT menyediakan fitur untuk melacak versi dari setiap transformasi data. Setiap kali transformasi data diperbarui atau diubah, pengguna dapat membuat versi baru dari transformasi tersebut. Ini memungkinkan pengguna untuk melacak perubahan yang dilakukan pada transformasi data dari waktu ke waktu, serta memungkinkan untuk kembali ke versi sebelumnya jika diperlukan. Versioning ini juga memudahkan pengembangan dan pengujian transformasi data dalam lingkungan pengembangan sebelum diterapkan ke lingkungan produksi.