1. Penggunaan database relational atau NoSQL umumnya ditentukan berdasarkan beberapa faktor, beberapa faktor tersebut antara lain faktor struktur dari data, kinerja, skalabilitas, kemudahan pengembangan, distribusi data, perkembangan data, biaya, kompleksitas query, konsistensi data, dan kemampuan analitik. Tidak semua solusi cocok untuk semua masalah, dalam beberapa kasus digunakan campuran dari kedua database. Database relational menggunakan relasi antar tabel untuk menyimpan dan mencocokan data dengan karakteristik umum dari setiap dataset. Sedangkan NoSQL umumnya tidak membutuhkan skema dan tidak memiliki relasi antar tabel dengan bentuk yang lebih mudah dibca dan dimengerti. Umumnya relational database digunakan untuk data yang membutuhkan integritas referensial dan konsistensi transaksi, sedangkan NoSQL digunakan pada data yang fleksibel, cepat dan memerlukan skalabilitas yang tinggi.
2. – Database adalah koleksi data terstruktur yang disimpan dalam bentuk tabel yang terorganisir dan skema yang jelas. Umumnya digunakan untuk menyimpan data transaksional dengan konsistensi dan integritas data. Umumnya database digunakan untuk aplikasi yang membutuhkan modifikasi seperti penambahan, pembaruan, dan pengambilan data secara realtime.

– Data Lake merupakan penyimpanan data yang fleksibel dan mampu menampung data dengan jenis yang berbeda, terstruktur, semi-terstruktur, dan tak terstruktur. Data lake biasa digunakan untuk mengumpulkan, menyimpan, dan menganalisis data mentah dalam bentuk asli. Umumnya cocok untuk menganalisis data lanjutan dan melakukan proses terhadap big data.

– Data Warehouse merupakan tempat penyimpanan data yang optimal untuk laporan dan analisa bisnis yang kompleks dengan menggunakan skema yang telah terstruktur untuk mengkonsolidasikan data dari berbagai sumber dan membentuk data yang siap untuk digunakan.

– Data Mart adalah turunan dari data warehouse yang merupakan penyimpanan yang dioptimalkan untuk kebutuhan spesisifik dengan subset data yang sesuai dan berfokus untuk kelompok atau departemen bisnis tertentu. Data mart cenderung lebih kecil, lebih cepat untuk diakses, dan lebih mudah dikelola.

1. Normalisasi database adalah proses merancang struktur database dengan tujuan mengurangi redudansi data, meningkatkan integritas data, dan mengurangi pekuang terjadinya anomali data. Pada suatu tabel yang memiliki banyak data yang redudan dan berulang akan dipecah menjadi beberapa tabel yang lebih kecil dan terfokus. Normalisasi data dapat menyebabkan peningkatan kompleksitas query pengambilan data yang disebabkan penyebaran data pada banyak tabel terpisah.

Tabel employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **employee\_id** | **employee\_name** | **job\_code** | **city\_code** | **province\_code** |
| 1 | John Smith | 101 | 201 | 301 |
| 2 | Alice Johnson | 102 | 202 | 302 |
| 3 | Bob Davis | 103 | 203 | 303 |
| 4 | Emily Wilson | 101 | 204 | 304 |
| 5 | Michael Lee | 102 | 205 | 305 |
| 6 | Sarah Brown | 103 | 206 | 306 |
| 7 | James Clark | 101 | 207 | 307 |
| 8 | Laura Taylor | 102 | 208 | 308 |
| 9 | Daniel White | 103 | 209 | 309 |
| 10 | Olivia Martin | 101 | 210 | 310 |

Tabel job

|  |  |
| --- | --- |
| **job\_code** | **job** |
| 101 | Software Engineer |
| 102 | Data Analyst |
| 103 | Data Engineer |

Tabel city

|  |  |
| --- | --- |
| **city\_code** | **city\_name** |
| 201 | New York |
| 202 | Los Angeles |
| 203 | Chicago |
| 204 | Houston |
| 205 | Miami |
| 206 | Boston |
| 207 | San Francisco |
| 208 | Seattle |
| 209 | Denver |
| 210 | Atlanta |

Tabel province

|  |  |
| --- | --- |
| **province\_code** | **Province\_name** |
| 301 | New York |
| 302 | California |
| 303 | Illinois |
| 304 | Texas |
| 305 | Florida |
| 306 | Massachusetts |
| 307 | California |
| 308 | Washington |
| 309 | Colorado |
| 310 | Georgia |