

МОБА SQL.

ВВЕДЕНИЯ.

ЗМІСТ

SQL

1. Характеристики
2. Реалізації SQL
3. Типи команд SQL
4. Переваги і недоліки мови SQL
 - 4.1. Переваги
 - 4.2. Недоліки

Характеристики

На початку 1970-х років в одній з дослідницьких лабораторій компанії IBM була розроблена експериментальна реляційна СУБД IBM System R, для якої потім була створена спеціальна мова SEQUEL, що дозволяла відносно просто управляти даними в цій СУБД. Аббревіатура SEQUEL розшифровувалась як Structured English QUERy Language - «структурований англійська мова запитів». Пізніше з юридичних міркувань мова SEQUEL була перейменована в SQL.

Метою розробки було створення простої не процедурної мови, якою міг скористатися будь-який користувач, який навіть не має навичок програмування. Власне розробкою мови запитів займалися

Дональд Чемберлін (Donald D. Chamberlin)
і Рей Бойс (Ray Boyce).



SQL (англ. Structured Query Language - «мова структурованих запитів») - універсальний комп'ютерний інформаційно-логічна мова, що з'явилася в результаті розробки реляційної моделі даних, що застосовувалася для створення, модифікації та управління даними в реляційних базах даних.

Спочатку, SQL була основним способом роботи користувача з базою даних і представляв собою невелику сукупність команд (операторів), що допускала створення таблиць, додавання в таблиці нових записів, витяг записів з таблиць (відповідно до заданого умовою), видалення записів і зміна структур таблиць. У зв'язку з ускладненням, мова SQL стала більш прикладною мовою програмування, а користувачі отримали можливість використовувати візуальні побудовники запитів.

SQL принципово відрізняється від традиційних алгоритмічних мов програмування насамперед тим, що вона відноситься до не процедурних мов. Мовою типу Сі можна записати крок за кроком всі інструкції, необхідні для виконання програми. Мова SQL дозволяє задати тільки те, "що потрібно робити", а саме виконання окремих операцій ("як робити") покладається безпосередньо на СУБД. Такий підхід значною мірою визначається самою філософією реляційних баз даних. СУБД в даному випадку розглядається як "чорний ящик", і що відбувається всередині нього, користувача не повинно стосуватися. Його повинно цікавити тільки внесення в базу даних необхідних змін і отримання правильної відповіді на запит.

Іншою особливістю SQL є так звана тризначна логіка. У більшості мов бульовий вираз може приймати тільки два значення: істина і хибність. Мова SQL дозволяє записувати в базу даних значення NULL (порожнє значення). NULL - це спеціальний код, який поміщається в стовпчик таблиці, якщо з якоїсь причини в ньому відсутні дані. Коли значення NULL бере участь в операціях порівняння, логічний результат буде ні істина і ні хибно, а невідомо.

Всі мови маніпулювання даними, створені для багатьох СУБД до появи реляційних баз даних, були орієнтовані на операції з даними, представленими у вигляді логічних записів файлів. Зрозуміло, це вимагало від користувача детального знання організації зберігання даних і серйозних зусиль для вказівки того, які дані необхідні, де вони розміщуються і як їх отримати.

Завдяки працюючим з файловими серверами СУБД, безліч користувачів отримують доступ до одних і тих же баз даних. Спрощується розробка різних автоматизованих систем управління організаціями. Однак при такому підході вся обробка запитів з програм або з терміналів користувача на них і виконується, тому для реалізації навіть простого запиту необхідно зчитувати з файлового сервера або записувати на нього цілі файли, а це веде до конфліктних ситуацій і перевантаження мережі. Для виключення зазначених недоліків була запропонована технологія клієнт-сервер, але при цьому знадобилася єдина мова спілкування з сервером - вибір припав на SQL.



Розглянутий мову SQL орієнтований на операції з даними, представленими у вигляді логічно взаємопов'язаних сукупностей таблиць-відносин.

Найважливіша особливість його структур - орієнтація на кінцевий результат обробки даних, а не на процедуру цієї обробки. Мова SQL сама визначає, де знаходяться дані, індекси і навіть які найбільш ефективні послідовності операцій слід використовувати для отримання результату, а тому вказувати ці деталі в запиті до бази даних не потрібно.

SQL в даний час отримала дуже широке поширення і фактично перетворився на стандартну мову реляційних баз даних. Стандарт на мову SQL був випущений Американським національним інститутом стандартів (ANSI) в 1986 р, а в 1987 р Міжнародна організація стандартів (ISO) прийняла його в якості міжнародного. Подальший розвиток мови постачальниками СУБД зажадав ухвалення в 1992 році нового розширеного стандарту (ANSI SQL-92 або просто SQL2). Наступним стандартом став SQL: 1999 (SQL3). В даний час діє стандарт, прийнятий в 2003 році (SQL: 2003) з невеликими модифікаціями, внесеними пізніше.

Реалізації SQL

Строго кажучи, SQL не є мовою програмування згідно з визначенням цього терміна. SQL є субмова даних, призначений для використання тільки в якості інтерфейсу з БД. Сам по собі SQL не містить тих засобів, які необхідні для розробки закінчених програм. Подібні завдання мають вирішуватися або за допомогою мов програмування або управління завданнями, або інтерактивно, в результаті дій, які виконуються самим користувачем. Унаслідок подібної незавершеності в плані організації обчислювального процесу мову SQL може застосовуватися у формі однієї з трьох прикладних реалізацій:

- 1. **Інтерактивний** (або автономний) SQL дає можливість користувачам безпосередньо отримувати інформацію з бази або записувати в неї дані. Інформація, що отримується за запитом SQL, може бути видана на екран, переадресовано в файл або на принтер.*
- 2. **Статичний** SQL дозволяє записати фіксований виконуваний код SQL, він зазвичай використовується в додатках на процедурних мовах.*

Є два різновиди статичного SQL: вбудований і модульний.

Вбудований SQL визначено як код SQL, який входить у вихідний текст програми, написаної на іншій мові програмування. У реальному житті більшість додатків розробляються на алгоритмічних мови, однак для звернення до БД використовується вбудований SQL.

У модульному варіанті оператори SQL записані в окремих модулях, які компонуються з модулями основного мови. У перший стандарт - SQL86 - був включений тільки модульний SQL, а вбудований SQL не знайшов собі місця в офіційній частині стандарту, хоча і був приведений в додатку. Сьогодні вбудований SQL користується набагато більшою популярністю серед розробників, ніж модульний, і стандарт SQL92 підтримує обидві версії статичного SQL.

*3. **Динамічний SQL** дає можливість генерувати код SQL під час виконання програми та використовується замість статичного SQL в тих випадках, коли при розробці програми необхідний код SQL ще не може бути визначений або залежить від того, який вибір зробить користувач.*

Оператори динамічного SQL зазвичай застосовуються в діалогових середовищах для побудови запитів і в графічних засобах розробки додатків БД.

Типи команд SQL

Реалізація в SQL концепції операцій, орієнтованих на табличне представлення даних, дозволила створити компактний мову з невеликим набором пропозицій. Мова SQL може використовуватися як для виконання запитів до даних, так і для побудови прикладних програм.

Основні категорії команд мови SQL призначені для виконання різних функцій, включаючи побудову об'єктів бази даних і маніпулювання ними, початкове завантаження даних в таблиці, оновлення та видалення існуючої інформації, виконання запитів до бази даних, управління доступом до неї і її загальне адміністрування.

Основні категорії команд мови SQL:

DDL - мова визначення даних;

DML - мова маніпулювання даними;

DQL - мова запитів;

DCL - мова керування даними;

команди адміністрування даних;

команди управління транзакціями

Визначення структур бази даних (DDL)

Мова визначення даних (Data Definition Language, DDL) дозволяє створювати і змінювати структуру об'єктів бази даних, наприклад, створювати і видаляти таблиці. Основними командами мови DDL є наступні: CREATE TABLE, ALTER TABLE, DROP TABLE, CREATE INDEX, ALTER INDEX, DROP INDEX.

Маніпулювання даними (DML)

Мова маніпулювання даними (Data Manipulation Language, DML) використовується для маніпулювання інформацією всередині об'єктів реляційної бази даних за допомогою трьох основних команд: INSERT, UPDATE, DELETE.

Вибірка даних (DQL)

Мова запитів DQL найбільш відомий користувачам реляційної бази даних, незважаючи на те, що він включає всього одну команду SELECT. Ця команда разом зі своїми численними опціями і пропозиціями використовується для формування запитів до реляційної бази даних.

Мова керування даними (DCL - Data Control Language)

Команди управління даними дозволяють управляти доступом до інформації, що знаходиться всередині бази даних. Як правило, вони використовуються для створення об'єктів, пов'язаних з доступом до даних, а також служать для контролю над розподілом привілеїв між користувачами. Команди управління даними наступні: GRANT, REVOKE.

Команди адміністрування даних

За допомогою команд адміністрування даних користувач здійснює контроль за виконуваними діями і аналізує операції бази даних; вони також можуть виявитися корисними при аналізі продуктивності системи. Не слід плутати адміністрування даних з адмініструванням бази даних, яке представляє собою загальне керівництво базою даних і має на увазі використання команд всіх рівнів.

Переваги і недоліки мови SQL

Переваги

Мова SQL є основою багатьох СУБД, тому що відповідає за фізичне структурування та запис даних на диск, а також за читання даних з диска, дозволяє приймати SQL-запити від інших компонентів СУБД і призначених для користувача додатків. Таким чином, SQL - потужний інструмент, який забезпечує користувачам, програмам і обчислювальних систем доступ до інформації, що міститься в реляційних базах даних.

Основні переваги мови SQL полягають в наступному:

- ✓ *стандартність* - використання мови SQL в програмах стандартизовано міжнародними організаціями;
- ✓ *незалежність від конкретних СУБД* - всі поширені СУБД використовують SQL, тому що реляційну базу даних можна перенести з однієї СУБД на іншу з мінімальними доробками;
- ✓ *можливість перенесення* з однієї обчислювальної системи на іншу - СУБД може бути орієнтована на різні обчислювальні системи, однак додатки, створені за допомогою SQL, допускають використання як для локальних БД, так і для великих багатокористувацьких систем;

- ✓ **реляційна основа мови** - SQL є мовою реляційних БД, тому вона стала популярним тоді, коли набула широкого поширення реляційна модель представлення даних. Таблична структура реляційної БД добре зрозуміла, а тому мова SQL проста для вивчення;
- ✓ **можливість створення інтерактивних запитів** - SQL забезпечує користувачам негайний доступ до даних, при цьому в інтерактивному режимі можна отримати результат запиту за дуже короткий час без написання складної програми;
можливість програмного доступу до БД - мова SQL легко використовувати в додатках, яким необхідно звертатися до баз даних. Одні і ті ж оператори SQL вживаються як для інтерактивного, так і програмного доступу, тому частини програм, що містять звернення до БД, можна спочатку перевірити в інтерактивному режимі, а потім вбудовувати в програму;
- ✓ **забезпечення різного представлення даних** - за допомогою SQL можна уявити таку структуру даних, що той чи інший користувач бачитиме різні її уявлення. Крім того, дані з різних частин БД можуть бути скомбіновані і представлені у вигляді однієї простої таблиці, а значить, уявлення придатні для посилення захисту БД і її налаштування під конкретні вимоги окремих користувачів;

- ✓ ***можливість динамічної зміни і розширення структури БД*** - мова SQL дозволяє маніпулювати структурою БД, тим самим забезпечуючи гнучкість з точки зору пристосованості БД до мінливих вимог предметної області;
- ✓ ***підтримка архітектури клієнт-сервер*** - SQL - один з кращих засобів для реалізації програм на платформі клієнт-сервер. SQL служить сполучною ланкою між взаємодією з користувачем клієнтської системою і серверної системою, що управляє БД, дозволяючи кожній з них зосередитися на виконанні своїх функцій.

Мова SQL може використовуватися широким колом фахівців, включаючи адміністраторів баз даних, прикладних програмістів і безліч інших кінцевих користувачів.

Мова SQL - перший і поки єдиний стандартний мова для роботи з базами даних, який отримав досить широке поширення. Практично всі найбільші розробники СУБД в даний час створюють свої продукти з використанням мови SQL або з SQL-інтерфейсом.

Недоліки

Невідповідність реляційної моделі даних

Творці реляційної моделі даних Едгар Кодд, Крістофер Дейт і їх прихильники вказують на те, що SQL не є істинно реляційною мовою. В опублікованому Крістофером Дейтом і Х'ю Дарвені Третньому Маніфесті викладають принципи СУБД наступного покоління і пропонують мову Tutorial D, яка є справді реляційною.

Складність

Хоча SQL і замислювалася як засіб роботи кінцевого користувача, врешті-решт вона стала настільки складною, що перетворилася в інструмент програміста.

Жодних відступів від стандартів

Незважаючи на наявність міжнародного стандарту ANSI SQL-92, багато компаній, що займаються розробкою СУБД (наприклад, Oracle, Sybase, Microsoft, MySQL AB), вносять зміни в мову SQL, тим самим відступаючи від стандарту. Таким чином, з'являються специфічні для кожної конкретної СУБД діалекти мови SQL.

Складність роботи з ієрархічними структурами