Administracja i programowanie pod Microsoft SQL Server 2000

Paweł Rajba

pawel@ii.uni.wroc.pl
http://www.kursy24.eu/

Zawartość modułu 5

- Kursory w SQL Serverze
  - Wprowadzenie
  - Modele kursorów
  - Używanie kursorów
  - Rodzaje kursorów
  - Praca z kursorami

Wprowadzenie do kursorów

- Co to jest kursor?
- Operacje wykonywane na kursorze:
  - utworzenie
  - pobieranie wartości
    - ewentualnie update lub delete
  - zamknięcie
  - zwolnienie zasobów

Wprowadzenie do kursorów

- Jak to się robi SQLServerze?
  - utworzenie
    - DECLARE nazwa CURSOR FOR zapytanie
    - OPEN nazwa
  - pobieranie wartości
    - FETCH NEXT FROM nazwa
  - zamykanie
    - CLOSE nazwa
  - zwolnienie zasobów
    - DEALLOCATE nazwa

Wprowadzenie do kursorów

- Przykład
  - sqlserver-05-01.sql

Modele kursorów

- Dalej omówimy dwa modele
  - Kursory języka Transact-SQL
  - Kursory serwerowe API

Modele kursorów

- Kursory języka Transact-SQL
  - Używamy zwykle we wsadach i procedurach
  - Pozwalają na modyfikację danych związanych z kursorem (standard ANSI wymaga trybu read-only)
  - Kursory można przewijać do przodu i wstecz, jak również pozycjonować względnie i bezwzględnie
  - Możliwości w T-SQL są bardzo spore (w porównaniu z innymi systemami)

Modele kursorów

- Kursory serwerowe API
  - stosowane głównie w aplikacjach klienckich
  - są zoptymalizowane pod kątem komunikacji klient-serwer
  - zwykle pobierany jest nie jeden wiersz, a np. 20 wierszy za jednym razem
  - nagłówki przesyłane są raz na początku, a przy każdym żądaniu wiersza
  - udostępniane przez różne biblioteki, np. DB-Library, ODBC, ADO, RDO, itp.

Używanie kursorów

- Kiedy używamy kursorów?
  - Najbardziej właściwa odpowiedź brzmi: jeśli już koniecznie musimy
  - Jeśli wystarczy, korzystamy z odpowiednich zapytań i złączeń
  - Czasami jednak kursory znacznie ułatwiają wykonanie zadania

Używanie kursorów

- Przykład 1, użycie kursorów T-SQLa
  - mamy tabelkę
    - Pracownik(pesel, imie, nazwisko, adres, pensja)
  - chcemy zrobić rzecz następującą: w zależności od pensji wykonać odpowiednią procedurę:
    - pensja w przedziale [0,2000] – proc. socjalne
    - pensja w przedziale [2000-5000] – proc. podatek
    - pensja powyżej 5000 – proc. superpodatek

Używanie kursorów

- Przykład 2, użycie kursorów API
  - użytkownik przegląda listę książek zawierającą w sumie 1000000 pozycji
  - zwykle nie ogląda wszystkich, tylko pewne wybrane fragmenty
  - i tutaj idealne są kursowy serwerowe API – wysyłamy tylko te fragmenty tabeli, które są żądane
  - pamiętamy, że za jednym razem wysyłamy tyle danych, ile mieści się w okienku aplikacji (albo 2, 3 razy więcej)

Rodzaje kursorów

- Kursory statyczne
  - dla danych wynikowych kursora tworzona jest tymczasowa tabela w bazie tempdb i na tej tabeli wykonywane są operacje
- Kursory kluczowe
  - tylko wartości kluczy są kopiowane do tymczasowej tabeli w tempdb
- Kursory dynamiczne
  - pracujemy na żywej tabeli
  - nie działa fetch absolute, ale działa fetch relative

Rodzaje kursorów

- Kursory jednokierunkowe
  - kursory dynamiczne, w których działa tylko fetch next
  - są szybkie
  - przetwarzanie takiego kursora polega zwykle na przeglądzie rekordów od początku do końca
  - instrukcja SELECT jest nadal szybsza od takiego kursora

Praca z kursorami

- Deklaracja kursora
  - Składnia ANSI-92
    - DECLARE *cursor_name* [INSENSITIVE] [SCROLL] CURSOR
      FOR *select_statement*
      [FOR { READ ONLY | UPDATE [OF *column_name* [,...*n*]] }]
  - Składnia rozszerzona
    - DECLARE *cursor_name* CURSOR
      [LOCAL | GLOBAL]
      [FORWARD_ONLY | SCROLL]
      [STATIC | KEYSET | DYNAMIC | FAST_FORWARD]
      [READ_ONLY | SCROLL_LOCKS | OPTIMISTIC]
      [TYPE_WARNING]
      FOR *select_statement*
      [FOR UPDATE [OF *column_name* [,...*n*]]]

Praca z kursorami

- Zasięg kursorów
  - kursory nie są widoczne poza danym połączeniem
  - kursory mogą być globalne lub lokalne
    - globalne są widoczne dla wszystkich wsadów
    - lokalne tylko dla bieżącego wsadu
  - domyślnie kursory są globalne
    - można to zmienić poprzez polecenie ALTER DATABASE
      - alter database nazwa_bazy set cursor_default local

Praca z kursorami

- Istotna uwaga
  - jeśli w deklaracji kursora jest użyta zmienna, to wartość jest pobierana już w momencie deklaracji a nie w momencie otwierania
- Przykład
  - sqlserver-05-02.sql

Praca z kursorami

- Otwieranie kursora
  - służy do tego polecenie OPEN
    - open nazwa_kursora
  - jego zadaniem jest udostępnienie danych i w razie potrzeby wypełnienie tabel tymczasowych
- Po wykonaniu OPEN można wykonać funkcję @@CURSOR_ROWS, która zwykle zwraca
  - -1 – kursor dynamiczny, nie można użyć tej funkcji
  - n – liczba wierszy w kursorze
  - 0 – kursor nie jest otwary lub nic nie zawiera

Praca z kursorami

- Pobieranie wiersza z kursora
  - służy do tego polecenie FETCH
    - FETCH selektor FROM nazwa [INTO @z1, @z2, ...]
  - dostępne selektory: NEXT, PRIOR, FIRST, LAST, ABSOLUTE n, RELATIVE n
  - w przypadku kurora dynamicznego nie jest dostępne ABSOLUTE n
  - w ABSOLUTE i RELATIVE n może być ujemne
    - w ABSOLUTE liczymy od końca
    - w RELATIVE liczymy wstecz od bieżącej pozycji
  - RELATIVE 0 – oznacza pobranie bieżącego wiersza

Praca z kursorami

- Pobieranie wiersza z kursora
  - Znaczenie opcji FETCH zaraz po otwarciu kursora
    - FETCH NEXT – to samo co FETCH FIRST
    - FETCH PRIOR – nic nie zwróci
    - FETCH RELATIVE n
      - jeśli n>0, to samo co FETCH ABSOLUTE n,
      - jeśli n<=0, nic nie zwróci
  - Zmienna @@FETCH_STATUS
    - jeśli 0, wiersz został pobrany
    - jeśli -1, przejście poza zakres
    - jesli -2, brak wiersza w kursorze kluczowym

Praca z kursorami

- Pobieranie wiersza z kursora
  - Klauzula INTO
    - jeśli ją podamy, wartości z wiersza zostaną przekazane pod wskazane zmienne
    - jeśli jej nie podamy, dla każdego wiersza będę generowane osobne zestawy wynikowe
  - Ta druga opcja jest oczywiście bardzo nieefektywna
  - Liczba zmiennych, typy i rozmiary muszą się zgadzać z pobranym wynikiem, inaczej będzie błąd

Praca z kursorami

- Modyfikacja kursora
  - Służy do tego polecenia UPDATE
    - UPDATE tabela SET przypisania
      WHERE CURRENT OF nazwa_kursora
  - Tak naprawdę wykonywane są dwie operacje: usunięcia i wstawienia wiersza
- Usunięcie wiersza z kursora
  - Służy do tego polecenie DELETE
    - DELETE FROM tabela WHERE CURRENT OF kursor

Praca z kursorami

- Zamykanie kursora
  - Służy do tego polecenie CLOSE
    - CLOSE nazwa_kursora
  - Zamknięcie kursora nie usuwa deklaracji – można bez problemu kursor ponownie otworzyć
  - Natomiast przy zamykaniu usuwaną są tabele tymczasowe, czyli przy ponownym otworzeniu są odtwarzane

Praca z kursorami

- Niszczenie kursora
  - Służy do tego polecenie DEALLOCATE
    - DEALLOCATE nazwa_kursora
  - Po tym poleceniu konieczna jest ponowna deklaracja
  - Polecenia tego nie ma w standardzie, ale dzięki niemu możemy lepiej zarządzać zasobami

Praca z kursorami

- Przykłady
  - sqlserver-05-03.sql
  - sqlserver-05-04.sql
  - sqlserver-05-05.sql
  - sqlserver-05-06.sql
  - sqlserver-05-07.sql

