

Oracle PL/SQL

Paweł Rajba

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

Zawartość modułu 7

- Dynamiczny SQL i PL/SQL
 - Pierwotny dynamiczny SQL
 - Pierwotny dynamiczny DDL
 - Pierwotny dynamiczny DML i SELECT
 - Pakiet DBMS_SQL
- Transakcje
 - Punkty bezpieczeństwa
 - Poziomy izolacji i blokady
 - Transakcje autonomiczne

Dynamiczny SQL i PL/SQL

- W ramach dynamicznego kodu możemy wykonywać
 - dynamiczne operacje DDL,
 - dynamiczne instrukcje SELECT,
 - dynamiczne instrukcje INSERT, UPDATE i DELETE
 - dynamiczny kod PL/SQL
 - obsługę wyjątków
 - operacje DDL i DML związane z obiektami
 - to omówimy przy okazji obiektów

Dynamiczny SQL i PL/SQL

- Dynamiczny kod możemy uruchamiać
 - za pomocą pakietu DBMS_SQL
 - za pomocą pierwotnego dynamicznego SQL

Pierwotny dynamiczny SQL

- Zalety tego rozwiązania
 - Łatwiejszy w użyciu
 - Szybszy niż DBMS_SQL
 - Obsługuje typy tworzone przez użytkownika
 - np. obiekty, kolekcje
 - Pozwala na przekierowanie wyników do rekordów

Pierwotny dynamiczny DDL

- Do uruchamiania takiego kodu wykorzystamy
 - EXECUTE IMMEDIATE string
- Przykład: dynamiczny-ddl.txt

Pierwotny dynamiczny DML i SELECT

- Do uruchamiania takiego kodu wykorzystamy
 - EXECUTE IMMEDIATE string INTO zmienne
USING zmienne
- Klauzula USING
 - pozwala na zmienny wiązanych do polecenia
 - użycie zmiennych wiązanych zwiększa wydajność
- Klauzula INTO
 - pozwala na przechwycenie wyników polecenia
- Przykład: dynamiczny-dml-select.txt

Pakiet DBMS_SQL

- Jest to starsze rozwiązanie
- Używanie go jest bardziej skomplikowane
- Kiedy przydaje się pakiet DBMS_SQL
 - w aplikacjach klienckich takich jak Oracle Developer
 - kiedy kod jest większy niż 32KB
 - kiedy w ramach sesji ta sama instrukcja jest powtarzana wielokrotnie, jest tylko raz parsowana
- Przykład: dbms-sql.txt

Transakcje

- Wprowadzenie
 - Po co są transakcje?
 - Własność ACID:
 - niepodzielność
 - spójność
 - odseparowanie
 - trwałość
 - W Oracle nie ma zagnieżdżania transakcji

Transakcje

- Rozpoczęcie transakcji
 - Niejawne przy pierwszym poleceniu DML
 - Jawne za pomocą polecenia SET TRANSACTION
- Transakcja kończy się, gdy
 - użytkownik wyda polecenie COMMIT lub ROLLBACK
 - użytkownik wyda polecenie DDL
 - najpierw jest COMMIT, potem polecenie DDL
 - użytkownik się odłączy (jest COMMIT)
 - użytkownik zerwie połączenie (jest ROLLBACK)

Punkty bezpieczeństwa

- Pozwalają na wycofanie fragmentu transakcji
- Przed wykonaniem operacji DML są tworzone takie punkty niejawnie i w razie niepowodzenia operacji, wycofanie jest do tego punktu
- Tworzenie punktu bezpieczeństwa
 - SAVEPOINT nazwa;
- Wykorzystanie
 - ROLLBACK TO nazwa;
- Przykład: transakcje-punktybezpieczenia.txt

Dobre zwyczaje

- Czas wykonania transakcji powinien być jak najkrótszy, co wpływa na czas blokad
 - Należy przemyśleć użycie pętli while
 - Nie należy w ramach transakcji czekać na dane od użytkownika
 - Wszystkie analizy należy wykonać przed rozpoczęciem transakcji
- Transakcja powinna otrzymywać dostęp do minimalnego zbioru wierszy

Transakcje i blokady

- Niepożądanym zjawiskom (np. brudne odczyty) zabiega ustawienie poziomu izolacji
 - jest on realizowany poprzez odpowiednio nakładane blokady
- Standard ANSI SQL definiuje następujące poziomy izolacji
 - Read uncommitted
 - Read committed
 - Repeatable read
 - Serializable

Transakcje i poziomy izolacji

- Oracle obsługuje dwa poziomy izolacji
 - Read committed
 - Serializable
- Ustawienie poziomu izolacji (i start transakcji)
 - SET TRANSACTION ISOLATION LEVEL
{SERIALIZABLE|READ COMMITTED}
[NAME 'tekst']
- Zmiana poziomu izolacji dla sesji
 - ALTER SESSION SET ISOLATION_LEVEL=
{SERIALIZABLE|READ COMMITTED}

Transakcje i poziom izolacji

- Przykład: transakcje-izolacja.txt

Transakcje autonomiczne

- Zasada działania
 - Główna transakcja jest zawieszana
 - Wykonywana jest transakcja autonomiczna
 - Główna transakcja jest kontynuowana
- Zalety
 - modularyzacja i uproszczenie kodu
 - np. osobne procedury mogą zajmować się audytem
 - pozwalają na COMMIT w wyzwalaczach
 - wywoływanie z SQL funkcji, które wykonują operacje DML

Transakcje autonomiczne

- Deklaracja transakcji autonomicznych
 - PRAGMA AUTONOMOUS_TRANSACTION
- Kiedy blok PL/SQL może być w ramach transakcji autonomicznej?
 - Jeśli blok jest anonimowy, musi być na najwyższym poziomie, czyli nie może być blokiem zagnieżdżonym
 - Jest procedurą lub funkcją, bezpośrednio w schemacie lub w ramach pakietu
 - Jest metodą typu obiektowego lub wyzwalaczem
- Przykład: transakcje-autonomiczne.txt