

Oracle PL/SQL

Paweł Rajba

pawel@ii.uni.wroc.pl

<http://www.kursy24.eu/>

Zawartość modułu 9

- Kolekcje
- Operacje na kolekcjach
- Testowanie kolekcji
- Kolekcje w bazie danych
- Funkcje tabelaryczne
- Kolekcje wielopoziomowe

Kolekcje

- Do kolekcji zaliczamy
 - Tabele o zmiennym rozmiarze
 - Tabele zagnieżdżone
- Charakterystyka kolekcji
 - Tablice jednowymiarowe
 - Numeracja elementów tablicy od 1
- Rozmiar jest
 - określony w przypadku tabel o zmiennym rozmiarze (ale rozmiar można zmienić)
 - nieograniczony w przypadku tabel zagnieżdżonych

Kolekcje

- Deklaracja tablic o zmiennym rozmiarze
 - TYPE nazwa_typu IS {VARRAY|VARYING ARRAY} (rozmiar) OF typ_danych [NOT NULL];
 - zmienna nazwa_typu;
- Deklaracja tablic zanieżdżonych
 - TYPE nazwa IS TABLE OF typ [NOT NULL]
 - zmienna nazwa_typu;

Kolekcje

- Komentarz
 - W definicji możemy korzystać z %type i %rowtype
 - Typem danych może być inna tablica o zmiennym rozmiarze jak również tablica zagnieżdżona
 - Kolejne elementy mogą być dodawane na końcu (poprzez odpowiednią metodę)
 - Z tabeli zagnieżdżonej można usuwać elementy (poprzez odpowiednią metodę)

Kolekcje

- Inicjalizowanie kolekcji
 - `tablica := nazwa_typu(el1, el2, el3, ...)`
- Dostęp do elementów
 - `t(i)`
- Klauzula `not null`
 - Nie można w konstruktorze podawać wartości `NULL`

Kolekcje

- Metody kolekcji
 - EXISTS
 - collection.EXISTS(index)
 - COUNT
 - collection.COUNT
 - DELETE
 - nested.DELETE
 - nested.DELETE(index)
 - nested.DELETE(start_index, end_index)

Kolekcje

- Metody cd.
 - FIRST
 - collection.FIRST
 - LAST
 - collection.LAST
 - NEXT
 - collection.NEXT(index)
 - PRIOR
 - collection.PRIOR(index)

Kolekcje

- Metody cd.
 - LIMIT
 - collection.LIMIT
 - EXTEND
 - collection.EXTEND
 - collection.EXTEND(n)
 - collection.EXTEND(n, index)
 - TRIM
 - collection.TRIM
 - collection.TRIM(n)

Kolekcje

- Przykład: kolekcje-start.txt

Operacje na kolekcjach

- Możemy wykonywać następujące operacje
 - SET – zwraca zbiór
 - MULTISSET UNION – suma
 - MULTISSET INTERSECT – przecięcie
 - MULTISSET EXCEPT – różnica
- Możemy użyć dodatkowo słowa DISTINCT
- Operacje dotyczące tabel zagnieżdżonych
- Przykład: kolekcje-operacje.txt

Testowanie kolekcji

- Do testowania mamy operatory
 - CARDINALITY – ilość elementów
 - MEMBER OF – czy jest w kolekcji
 - IS A SET – czy jest zbiorem
 - IS EMPTY – czy pusty
 - SUBMULTISET – czy jest podzbiorem
- Operacje dotyczące tabel zagnieżdżonych
- Przykład: kolekcje-testowanie.txt

Kolekcje w bazie danych

- W bazie danych możemy utworzyć
 - typ reprezentujący
 - tabele o zmiennej długości (VARRAY)
 - tabelę zagnieżdżoną
 - tabelę o kolumnie typu kolekcji
- Definicja typu
 - CREATE OR REPLACE TYPE nazwa IS VARRAY(rozmiar) OF typ_danych
 - CREATE OR REPLACE TYPE nazwa IS TABLE OF typ_danych

Kolekcje w bazie danych

- Składowanie tablic o zmiennym rozmiarze
 - Tworzymy typ
 - CREATE OR REPLACE TYPE num_list AS VARRAY(10) OF VARCHAR2(15);
 - Tworzymy tabelę z kolumną o utworzonym typie
 - CREATE TABLE ip_addr_list(
List_ID INT PRIMARY KEY, ip_addresses num_list);
- Dane tablic są składowane wewnątrz tabeli
- Operacje DML obsługujemy standardowo

Kolekcje w bazie danych

- Składowanie tablic o zmiennym rozmiarze
 - Tworzymy typ
 - `CREATE OR REPLACE TYPE num_list AS TABLE OF VARCHAR2(15);`
 - Tworzymy tabelę z kolumną o utworzonym typie
 - `CREATE TABLE ip_addr_list(List_ID INT PRIMARY KEY, ip_addresses num_list) NESTED TABLE ip_addresses STORE AS ip_addresses_tab`
- Dane tablic są składowane w dodatkowo utworzonej tabeli
- Operacje DML obsługujemy standardowo

Kolekcje w bazie danych

- Przykład: kolekcje-skladowanie.txt

Funkcje tabelaryczne

- Są to funkcje zwracające w wyniku tabele
- Powyższą funkcjonalność można zrealizować przy pomocy omawianych tablic, czyli
 - tablic zmiennej długości (VARRAY)
 - tablic zagnieżdżonych
- Najczęstsze zastosowanie to pobranie wyniku tabeli w instrukcji SELECT

Funkcje tabelaryczne

- Etapy prowadzące do korzystania z wyniku funkcji tabelarycznej:
 - Utworzenie typu (CREATE TYPE)
 - Utworzenie odpowiedniej funkcji
 - Wywołanie SELECT wraz z operatorem TABLE
- Przykład: kolekcje-funkcje.txt

Kolekcje wielopoziomowe

- Można tworzyć kolekcje wielowymiarowe
 - tworzymy po prostu kolekcję kolekcji
- Dopuszczalne są dowolne kombinacje
- Przykład: kolekcje-wielopoziomowe.txt