

Ćwiczenie 10.

PL/SQL. Kursory i wyjątki – zadania

1. Zdefiniuj kursor zawierający nazwiska i daty zatrudnienia wszystkich asystentów. Posłuż się tym kusem do wyświetlenia rekordów w formie zdań „Asystent <nazwisko> pracuje od <data_zatrudnienia>”. Posłuż się poleceniem FETCH.

```
Asystent Opolski pracuje od 01.10.1992
Asystent Kotarski pracuje od 01.09.1993
Asystent Siekierski pracuje od 01.10.1993
Asystent Dolny pracuje od 01.09.1992
```

2. Zdefiniuj kursor, dzięki któremu będzie można wyświetlić trzech najlepiej zarabiających pracowników. Posłuż się atrybutem kursora %ROWCOUNT.

```
1. Marecki
2. Nowak
3. Janicki
```

3. Zdefiniuj kursor, który posłuży do dokonania następującej modyfikacji: pracownikom zespołu „ALGORYTMY” podnieś płacę dodatkową o 100 złotych, pracownikom zespołu „ADMINISTRACJA” podnieś płacę dodatkową o 150 złotych, w pozostałych zespołach usuń doktorantów.

```
SQL> select placa_dod from pracownicy natural join zespoly
2  where nazwa = 'ALGORYTMY';
```

```
PLACA_DOD
-----
610
```

```
SQL> select nvl(placa_dod,0) from pracownicy natural join zespoly
2  where nazwa = 'ADMINISTRACJA';
```

```
NVL(PLACA_DOD,0)
-----
980,5
0
```

```
SQL> select count(*) from pracownicy natural join zespoly
2  where nazwa not in ('ADMINISTRACJA','ALGORYTMY')
3  and etat = 'DOKTORANT';
```

```
COUNT(*)
-----
2
```

```
SQL> declare ...
```

Procedura PL/SQL została zakończona pomyślnie.

```
SQL> select placa_dod from pracownicy natural join zespoly
2  where nazwa = 'ALGORYTMY';
```

```
PLACA_DOD
-----
710
```

```
SQL> select nvl(placa_dod,0) from pracownicy natural join zespoly
2  where nazwa = 'ADMINISTRACJA';
```

```
NVL(PLACA_DOD,0)
-----
          1130,5
           150
```

```
SQL> select count(*) from pracownicy natural join zespoly
2  where nazwa not in ('ADMINISTRACJA','ALGORYTMY')
3  and etat = 'DOKTORANT';
```

```
COUNT(*)
-----
          0
```

4. Napisz program, który zapyta się użytkownika o etat a następnie wyświetli nazwiska wszystkich pracowników posiadających dany etat. Zastosuj pętlę FOR z kursorem sparametryzowanym.

```
SQL> declare ...
```

```
Podaj wartość dla nazwa_etatu: ASYSTENT
stare  4: v_etat pracownicy.etat%TYPE := '&nazwa_etatu';
nowe   4: v_etat pracownicy.etat%TYPE := 'ASYSTENT';
```

```
Opolski
Kotarski
Siekierski
Dolny
```

Procedura PL/SQL została zakończona pomyślnie.

5. Rozszerz program z zadania 4. z kursorów o obsługę sytuacji podania niepoprawnego etatu (etatu, na którym nie pracuje żaden pracownik).

```
SQL> declare ...
```

```
Podaj wartość dla nazwa_etatu: PREZES
stare  3: v_etat pracownicy.etat%TYPE := '&nazwa_etatu';
nowe   3: v_etat pracownicy.etat%TYPE := 'PREZES';
Brak pracownika na etacie PREZES
```

Procedura PL/SQL została zakończona pomyślnie.

6. Napisz program z użyciem kursora, który odczyta informacje o wszystkich profesorach i przyzna im podwyżkę w wysokości 10% sumy płac podstawowych ich podwładnych. Jeśli po podwyżce pensja któregoś z profesorów przekroczyłaby 4000 złotych, program powinien zgłosić błąd (skorzystaj z procedury RAISE APPLICATION ERROR).

```
SQL> declare ...
```

```
DECLARE
*
```

```
BŁĄD w linii 1:
ORA-20010: Zbyt wysoka płaca!
ORA-06512: przy linii 11
```