

Ćwiczenie 11.

PL/SQL. Procedury, funkcje, pakiety, wyzwalacze – zadania

1. Napisz procedurę Podwyzka, która wszystkim pracownikom zespołu o danym numerze (parametr) podniesie płacę podstawową o podany procent (parametr). Domyślnie podwyzka powinna wynosić 15%.

```
SQL> exec podwyzka(10, 25);
```

Procedura PL/SQL została zakończona pomyślnie.

2. Dodaj do powyższej procedury obsługę błędów – jeśli podano numer nieistniejącego zespołu, procedura powinna wygenerować błąd o numerze -20010 i odpowiednim komunikacie.

```
SQL> exec podwyzka(100, 25);  
BEGIN podwyzka(100, 25); END;
```

*

```
BŁĄD w linii 1:  
ORA-20010: Zły identyfikator zespołu!  
ORA-06512: przy "WSNHID.PODWYZKA", linia 10  
ORA-06512: przy linia 1
```

3. Napisz funkcję PlacaNetto, która dla podanej płacy brutto (parametr) i podanej stawki podatku (parametr o wartości domyślnej 20%) wyliczy płacę netto.

```
SQL> begin  
2   dbms_output.put_line(PlacaNetto(1000,20));  
3 end;  
4 /  
800
```

Procedura PL/SQL została zakończona pomyślnie.

4. Napisz funkcję Staz, która dla daty zatrudnienia pracownika (parametr) wyliczy staż pracy w latach.

```
SQL> begin  
2   dbms_output.put_line(staz(date '1990-03-30'));  
3 end;  
21
```

Procedura PL/SQL została zakończona pomyślnie.

5. Napisz funkcję Silnia, która dla danego n obliczy $n! = 1 * 2 * \dots * n$ (zastosuj iterację).

```
SQL> begin  
2   dbms_output.put_line(silnia(10));  
3 end;  
4 /  
3628800
```

Procedura PL/SQL została zakończona pomyślnie.

6. Napisz pakiet Konwersja, zawierający funkcje: CelsToFahr (konwertującą skalę Celsjusza na skalę Fahrenheita) i FahrToCels (konwertującą skalę Fahrenheita na skalę Celsjusza). Wskazówka: $TC = 5/9 * (TF - 32)$, $TF = 9/5 * TC + 32$.

```
SQL> begin
  2   dbms_output.put_line(konwersja.celstofahr(100));
  3   dbms_output.put_line(konwersja.fahrtocels(212));
  4 end;
  5 /
212
100
```

Procedura PL/SQL została zakończona pomyślnie.

7. Napisz wyzwalacz, który będzie automatycznie przyznawał kolejne identyfikatory nowym zespołom (utwórz i wykorzystaj sekwencję). Przetestuj działanie wyzwalacza.

```
SQL> insert into zespoly(nazwa) values('Zespół testowy');
```

1 wiersz został utworzony.

```
SQL> select id_zesp
  2   from zespoly
  3  where nazwa = 'Zespół testowy';
```

```
   ID_ZESP
-----
        70
```

8. Stwórz relację HISTORIA o schemacie (*id_prac*, *placa_pod*, *etat*, *zespól*, *data*). Napisz wyzwalacz, który po każdej modyfikacji płacy podstawowej, etatu lub zespołu w relacji PRACOWNICY będzie wpisywał wartości historyczne do relacji HISTORIA. Atrybut *data* będzie przechowywała datę modyfikacji.

```
SQL> update pracownicy
  2   set placa_pod = placa_pod + 100
  3  where nazwisko = 'Kowalski';
```

1 wiersz został zmodyfikowany.

```
SQL> update pracownicy
  2   set etat = 'ASYSTENT'
  3  where nazwisko = 'Siekierski';
```

1 wiersz został zmodyfikowany.

```
SQL> select * from historia;
```

ID_PRAC	PLACA_POD	ETAT	ZESPOL	DATA
140	3230	PROFESOR	20	11/09/22
220	1889	ASYSTENT	20	11/09/22