# Temat 11 – PL/SQL, wyzwalacze

1. Utwórz wyzwalacz, który nie pozwoli usunąć rekordu z tabeli *Emp*.

CREATE OR REPLACE TRIGGER DeleteFromEmp

BEFORE DELETE

ON EMP

BEGIN

RAISE\_APPLICATION\_ERROR(-20100, 'Nie mozna usuwac rekordow z tabeli EMP.');

END;

DELETE FROM EMP

WHERE SAL > 1000;

1. Utwórz wyzwalacz, który przy wstawianiu lub modyfikowaniu danych w tabeli *Emp* sprawdzi czy nowe zarobki (wstawiane lub modyfikowane) są większe niż 1000. W przeciwnym przypadku wyzwalacz powinien zgłosić błąd i nie dopuścić do wstawienia rekordu. Uwaga: Ten sam efekt można uzyskać łatwiej przy pomocy więzów spójności typu CHECK. Użyjmy wyzwalacza w celach treningowych.

CREATE OR REPLACE TRIGGER ModyfikujZarobki

BEFORE INSERT OR UPDATE

ON EMP

FOR EACH ROW

BEGIN

IF :NEW.sal > 1000 THEN

RAISE\_APPLICATION\_ERROR(-20100, 'Zarobki nie moga byc wieksze niz 1000.');

END IF;

END;

UPDATE emp

SET sal = 2000

WHERE sal < 1000;

UPDATE emp

SET sal = 900

WHERE sal < 1000;

1. Utwórz tabelę *budzet*:  
     
   CREATE TABLE budzet (wartosc INT NOT NULL)  
     
   W tabeli tej będzie przechowywana łączna wartość wynagrodzenia wszystkich pracowników. Tabela będzie zawsze zawierała jeden wiersz. Należy najpierw obliczyć początkową wartość zarobków:  
     
   INSERT INTO budzet (wartosc)  
   SELECT SUM(sal) FROM emp  
     
   Utwórz wyzwalacz, który będzie pilnował, aby wartość w tabeli *budzet* była zawsze aktualna, a więc przy wszystkich operacjach aktualizujących tabelę *emp* (INSERT, UPDATE, DELETE), wyzwalacz będzie aktualizował wpis w tabeli *budżet*

CREATE OR REPLACE TRIGGER SprawdzSal

AFTER INSERT OR UPDATE OR DELETE

ON EMP

BEGIN

DELETE FROM budzet

WHERE 1 = 1;

INSERT INTO budzet (wartosc)

SELECT SUM(sal) FROM emp;

END;

SELECT \* FROM BUDZET;

UPDATE EMP

SET SAL = 900

WHERE SAL > 4000;

SELECT \* FROM BUDZET;

SELECT \* FROM EMP;

1. Napisz jeden wyzwalacz, który:
   * Nie pozwoli usunąć pracownika, którego pensja jest większa od 0.
   * Nie pozwoli zmienić nazwiska pracownika.
   * Nie pozwoli wstawić pracownika, który już istnieje (sprawdzając po nazwisku).

CREATE OR REPLACE TRIGGER SprawdzPracownika

BEFORE INSERT OR UPDATE OR DELETE

ON EMP

FOR EACH ROW

DECLARE

v\_exists INTEGER := 1;

BEGIN

IF DELETING AND :OLD.SAL > 0 THEN

RAISE\_APPLICATION\_ERROR(-20100, 'Nie mozna usunac pracownika z pensja wieksza od 0.');

END IF;

IF UPDATING AND :OLD.ename <> :NEW.ename THEN

RAISE\_APPLICATION\_ERROR(-20100, 'Nie mozna zmieniac nazwiska pracownika.');

END IF;

IF INSERTING THEN

SELECT COUNT(\*) INTO v\_exists

FROM EMP

WHERE ename = :NEW.ename;

IF v\_exists > 0 THEN

RAISE\_APPLICATION\_ERROR(-20003, 'Pracownik o podanym nazwisku już istnieje.');

END IF;

END IF;

END;

INSERT INTO EMP (EMPNO, ENAME, SAL)

VALUES (1111, 'BLAKE', 12);

1. Napisz wyzwalacz, który:
   * Nie pozwoli zmniejszać pensji.
   * Nie pozwoli usuwać pracowników.

CREATE OR REPLACE TRIGGER SprawdzPracownika2

BEFORE UPDATE OR DELETE

ON EMP

FOR EACH ROW

BEGIN

IF UPDATING AND :NEW.sal < :OLD.sal THEN

RAISE\_APPLICATION\_ERROR(-20100, 'Nie mozna zmieniejszac pensji pracownikowi');

END IF;

IF DELETING THEN

RAISE\_APPLICATION\_ERROR(-20100, 'Nie mozna usuwac pracownikow');

END IF;

END;