

Wyzwalacze cz.2

Zadanie.1 Stworzyć tabele według wzorca:

- a)

```
create table Samoloty(  
    id int PRIMARY KEY,  
    nazwa varchar(256) NOT NULL,  
    iloscMiejsc int NOT NULL,  
    pojemnoscBagazu int NOT NULL  
)
```
- b)

```
create table helikoptery(  
    id int PRIMARY KEY,  
    nazwa varchar(256) NOT NULL,  
    iloscMiejsc int NOT NULL  
)
```
- c)

```
create table Loty(  
    id int PRIMARY KEY,  
    aktualnaCena decimal(10,2) NOT NULL,  
    cel varchar(256) NOT NULL,  
    idSamolotu int NOT NULL,  
    dataWylotu date NOT NULL  
)
```
- d)

```
create table Bilety(  
    id int PRIMARY KEY,  
    idLotu int NOT NULL REFERENCES Loty(id),  
    cena decimal(10,2) NOT NULL,  
    klient varchar(256) NOT NULL  
)
```

Zadanie.2 Uzupełnić tabele danymi (tabelę helikoptery samodzielnie)

- a)

```
INSERT INTO Samoloty VALUES  
(1, 'Boeing1', 5, 30),  
(2, 'Boeing2', 4, 40),  
(3, 'Boeing3', 3, 30),  
(4, 'AirBus1', 7, 50),  
(5, 'AirBus2', 3, 50);
```
- b)

```
INSERT INTO Loty VALUES  
(4, 1500, 'Warszawa', 1, '2016-10-18'),  
(5, 1200, 'Poznań', 2, '2016-10-18'),  
(6, 1800, 'Gdańsk', 3, '2016-10-19'),  
(7, 2100, 'Lublin', 2, '2016-10-19'),  
(8, 1700, 'Warszawa', 1, '2016-10-19'),  
(9, 2300, 'Kraków', 1, '2016-10-20'),  
(10, 1300, 'Poznań', 3, '2016-10-20'),  
(11, 1800, 'Gdańsk', 2, '2016-10-20'),  
(12, 1500, 'Warszawa', 2, '2016-10-21');
```
- c)

```
INSERT INTO Bilety VALUES  
(1, 4, 400, 'Abacki J.'),  
(2, 4, 400, 'Abacka A.'),  
(3, 4, 400, 'Abacki M.'),  
(4, 4, 400, 'Babacki L.'),  
(5, 4, 400, 'Babacka A.'),  
(6, 5, 500, 'Cacki M.'),  
(7, 5, 500, 'Cacka Z.'),  
(8, 5, 500, 'Dacka I.'),  
(9, 5, 500, 'Dacki P.'),  
(10, 6, 400, 'Abacki J.'),  
(11, 6, 400, 'Abacka A.'),  
(12, 6, 400, 'Abacki M.'),  
(13, 7, 600, 'Babacki L.'),  
(14, 7, 600, 'Babacka A.'),  
(15, 8, 500, 'Cacki M.'),  
(16, 8, 500, 'Cacka Z.');
```

Zadania

1. Stwórz trigger, który po wstawieniu wierszy do tabeli bilety zmieni ich cenę na aktualną.
2. Stwórz trigger, który podczas wstawienia biletu sprawdzi czy w samolocie jest jeszcze wolne miejsce. Załóż, że wszystkie bilety są na ten sam lot.
3. Stwórz trigger na tabeli loty, które umożliwi wstawianie tylko takich lotów, których id samolotu jest w tabeli Samoloty lub Helikoptery.
4. Stwórz dwa trigery na tabelach samoloty i helikoptery, które sprawdzą czy dodawany pojazd ma poprawne id. Jedno id nie może powtarzać się w dwóch tabelach.
5. Stwórz trigger, który po usunięciu wierszy z tabeli bilety zmieni aktualną cenę na niższą o 10% jeśli ilość biletów na dany lot spadnie poniżej 3. Usuwamy bilety tylko na jeden lot w jednym zapytaniu DELETE.
6. Sprawdź co się stanie, jeśli wykonamy operację DELETE * FROM bilety. Czy trigger się wykonał?
7. Stwórz trigger, który podczas wstawienia biletu sprawdzi czy w samolocie jest jeszcze wolne miejsce, jeśli tak to doda bilet, powiększy cenę bazową lotu o 5%, oraz policzy obniżkę tym klientom, który kupili już kiedykolwiek więcej niż 2 biletów.