

13. LABORATORIJSKE VJEŽBE

Tema: Okidači

Zadatak 1.

Zadana je šema nove relacije:

```
CREATE TABLE uspjeh (  
    mbrStud CHAR(8) NOT NULL  
    , bodovi INTEGER NOT NULL  
    , PRIMARY KEY (mbrStud)  
);
```

U relaciji **uspjeh** za svakog studenta koji se pojavljuje u relaciji **ispit** trebao bi pisati broj bodova koji se izračunava na sljedeći način:

```
0 položenih ispita 0 bodova  
1 položeni ispit 5 bodova  
2 položena ispita 15 bodova  
više od dva položena ispita 50 bodova
```

Podaci u relaciji **uspjeh** uvijek moraju biti u skladu s podacima u relaciji **ispit** (osim na početku, nakon kreiranja relacije **uspjeh**).

Sve korisnike (čak i administratora) treba spriječiti u brisanju zapisa ili ažuriranju (mijenjaju) matičnih brojeva u relacijama **ispit** i **uspjeh**. U slučaju pokušaja obavljanja takvih operacija dojaviti pogreške: "Nedozvoljeno brisanje u relaciji ispit"; "Nedozvoljeno brisanje u relaciji uspjeh"; "Nedozvoljeno ažuriranje matičnih brojeva u relaciji ispit"; "Nedozvoljeno ažuriranje matičnih brojeva u relaciji uspjeh".

Svaki unos ili izmjenu zapisa u relaciji uspjeh, koji bi narušio konzistentnost podataka, treba spriječiti i dojaviti pogrešku "Pokušano obavljanje nedozvoljene operacije u relaciji uspjeh".

Unos i izmjena (**samo nekih - kojih?**) podataka u relaciji ispit moraju uzrokovati odgovarajući unos ili izmjenu podataka u relaciji uspjeh.

Napišite naredbe za kreiranje okidača i procedura kojima će se, u skladu s navedenim specifikacijama, spriječiti narušavanje konzistentnosti podataka.

Zadatak 2.

Napišite primjere SQL naredbi kojima bi se (kad ne bi bilo okidača) mogla narušiti konzistentnost podataka relacije **uspjeh** iz drugog zadatka:

- UPDATE naredbu koja ažurira jednu ocjenu u relaciji *ispit* zbog koje će se promijeniti uspjeh nekog studenta
- UPDATE naredbu koja ažurira više ocjena u relaciji *ispit*
- INSERT naredbu koja ubacuje jedan zapis u relaciju *ispit*
- INSERT naredbu koja "istovremeno" ubacuje više zapisa u relaciju *ispit*. Koristiti pomoćnu relaciju (TEMP TABLE) i naredbu INSERT INTO ... SELECT ... FROM ...
- UPDATE naredbu koja ažurira matični broj u relaciji *ispit*
- UPDATE naredbu koja ažurira matični broj u relaciji *uspjeh*
- DELETE naredbu koja briše više zapisa u relaciji *ispit*
- DELETE naredbu koja briše jedan zapis u relaciji *uspjeh*

Na laboratorijskim vježbama kreirajte relaciju **uspjeh**, sve pripremljene okidače i procedure, te relaciju **uspjeh** dovedite u konzistentno stanje obavljanjem operacije

```
UPDATE ispit SET ocjena = ocjena WHERE 1=1;
```

(ova naredba bi trebala aktivirati okidač koji će za svaku ažuriranu n-torku ubaciti ili ažurirati zapis u relaciji **uspjeh**)

Testirajte djelovanje okidača pri izvođenju SQL naredbi iz zadatka 3 (izvesti jednu po jednu svaku pripremljenu naredbu te provjeriti rezultat djelovanja okidača).