

Baze podataka - kolokvijum

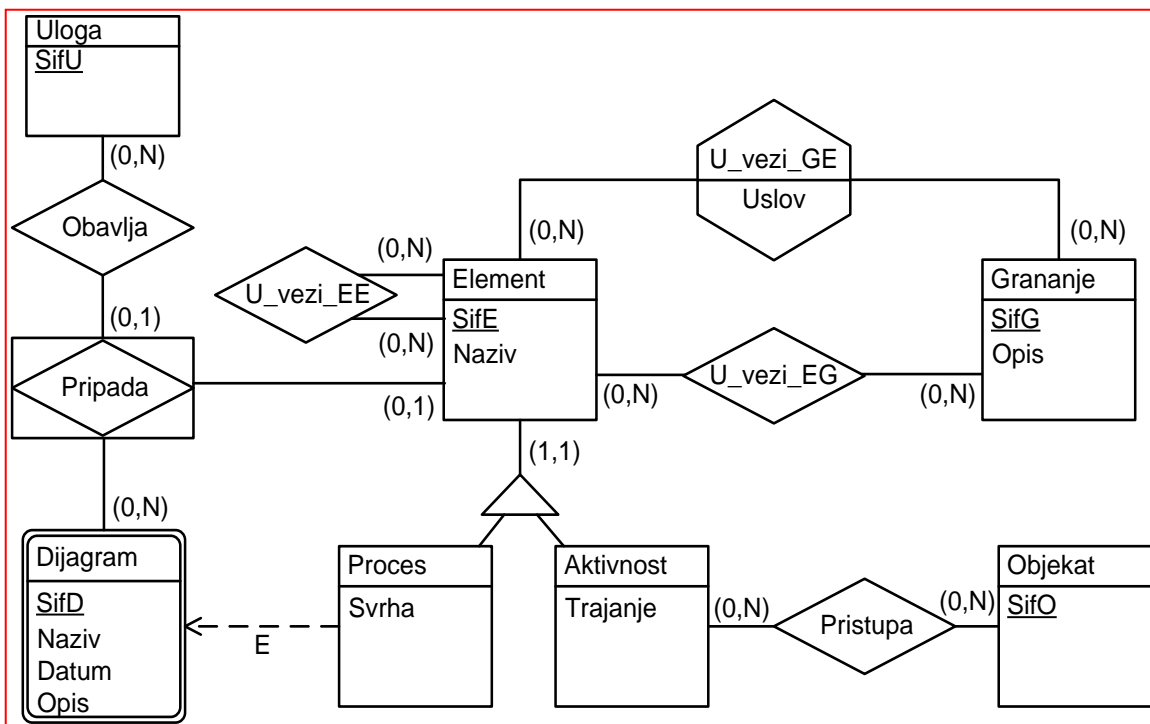
Kolokvijum traje **150** minuta

| | | |
|-------------------------------------|-----------------|------------|
| Ime prezime i broj indeksa studenta | Potpis dežurnog | Broj poena |
| | | |

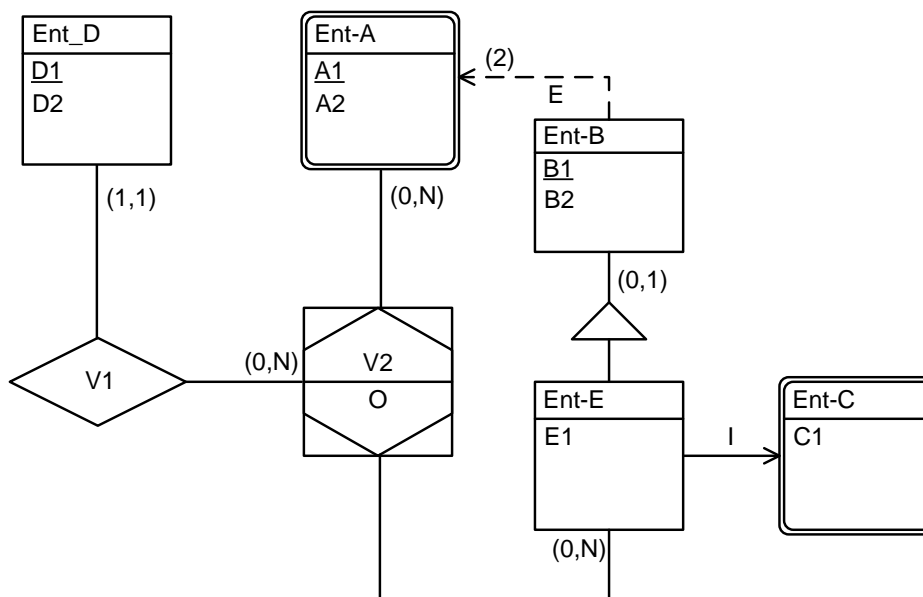
Napomena: Nije dozvoljena upotreba literature.

1. (7) Posmatra se baza podataka za potrebe čuvanja informacija o dijagramima poslovnih procesa. Dijagram ima naziv, datum kreiranja i opis. Jedan dijagram se odnosi na tačno jedan proces. Za procese se prati svrha. U bazi se evidentiraju i uloge. Uloge mogu, na određenom dijagramu, obavljati aktivnosti i procese (podprocese). Aktivnosti i procesi mogu biti međusobno povezani. Aktivnosti i procesi imaju naziv i informaciju o tome sa kojim grananjima su u vezi. Grananja sadrže opis i uslove pod kojim su povezani sa određenim aktivnostima, i procesima. Grananja međusobno ne mogu biti povezana. Aktivnosti sadrže i informaciju o trajanju kao i tome kojim objektima pristupaju. Više različitih aktivnosti može pristupati istom objektu.

Za model prikazan na slici potrebno je ukloniti suvišne (odnosno dodati potrebne) attribute/entitete, a potom dodati potrebne neposredne i posredne odnose (slabe entitete nije potrebno duplo zaokruživati).



2. (3) Model entiteta i odnosa, prikazan na slici, prevesti u šemu relacije baze podataka, uz naznaku svih stranih ključeva zaokruživanjem. (Napomena: idenstifikaciona zavisnost je prikazana punom linijom, a egzistencijalna isprekidanom)



Odgovor:

Ent-A: (A1, A2, B11, B12)

Ent-B: (B1, B2)

Ent-C: (B1, C1)

Ent-D: (D1, D2, A1, B1)

Ent-E: (B1, E1)

V1: -

V2: (A1, B1, O)

3. Dat je deo šeme relacije baze podataka fakulteta, kod koga postoje odseci koji se granaju na veći broj pododseka.

STUDENT (SifS, Ime, BrIndeksa);

NASTAVNIK (SifN, Ime);

ODSEK (SifO, Naziv, BrStudenata, SifONad);

PREDMET (SifP, Naziv, SifO);

ODSLUSAO (SifS, SifP);

ROK(SifR, Naziv, DatumPocetka, DatumKraja);

ISPIT(SifI, SifP, Datum, SifR);

POLAGANJE (SifL, SifI, SifS, Ocena, Datum, SifN, Komentar);

a.(5) Napisati SQL skript kojim se formira tabela POLAGANJE ukoliko je poznato da je komentar niz od 25 karaktera sa podrazumevanom vrednošću "Prazno". Ocena mora biti u opsegu od 3 do 10 (3–udaljen, 4–nije izašao). Komentar ne sme imati podrazumevanu vrednost ukoliko je ocena jednaka 3, u ostalim slučajevima vrednost komentara nije bitna. Datum polaganja mora biti nakon datuma održavanja ispita, a pre kraja ispitnog roka. Datumi se evidentiraju kao celobrojne vrednosti.

Odgovor:

```
CREATE TABLE Polaganje
( SifI INT PRIMARY KEY,
  SifI INT NOT NULL REFERENCES Ispit ON UPDATE CASCADE,
  SifS INT NOT NULL REFERENCES Student ON UPDATE CASCADE,
  Ocena INT NOT NULL CHECK (Ocena BETWEEN 3 AND 10),
  Datum INT NOT NULL CHECK (Datum BETWEEN 1 AND 365),
  SifN INT NOT NULL REFERENCES Nastavnik ON UPDATE CASCADE,
  Komentar CHAR(25) NOT NULL DEFAULT = 'Prazno',
  CHECK (Ocena > 3 OR Komentar <> 'Prazno')
)

CREATE ASSERTION DatumIspita
CHECK (NOT EXISTS ( SELECT *
                    FROM Polaganje P, Ispit I, Rok R
                    WHERE P.SifI = I.SifI AND I.Rok = R.Rok
                    AND P.Datum >= I.Datum AND P.Datum <= R.DatumKraja
                  )
)
-- provera da li je isti predmet položen više puta, ili isti ispit više puta, nije ocenjivan
```

b. (3) Sastaviti SQL skript koji vraća ukupan broj studenata na odseku sa šifrom RZ. Odsekom se smatra konkretan odsek i svi njegovi pododseci (i pododseci tih pododseka itd.).

Odgovor:

```
WITH RECURSIVE SviRZOdseci(SifO, BrStudenata, SifONad)
AS SELECT SifO, BrStudenata, SifONad
   FROM Odsek
   WHERE SifO='RZ'
   UNION
   SELECT O.SifO, O.BrStudenata, O.SifONad
   FROM Odsek O, SviRZOdseci R
   WHERE O.SifONad=R.SifO

SELECT SUM(BrStudenata)
FROM SviRZOdseci;
```

c. (5) Sastaviti SQL skript koji vraća šifre i imena studenata koji mogu da prijave ispit za predmet sa šifrom MAT1. Neki od preduslova su da student mora da odsluša predmet, i da sme da ima najviše tri pokušaja polaganja. Ukoliko je student prijavio, a nije izašao na ispit to se evidentira ocenom 4, odnosno ocenom 3 ukoliko je udaljen sa ispita, i u oba slučaja to se računa kao pokušaj.

Odgovor:

```
CREATE VIEW StudentiNotOK(SifS)
AS  SELECT P.SifS
    FROM Polaganje P, Ispit I
    WHERE P.SifI=I.SifI AND I.SifP='MAT1'
    GROUP BY P.SifS, I.SifP
    HAVING (COUNT(*) > 2) OR (MAX(P.Ocena) > 5);

SELECT SifS, Ime
FROM Student
WHERE SifS IN (SELECT SifS FROM Odslusao WHERE SifP='MAT1')
AND SifS NOT IN (SELECT SifS FROM StudentiNotOK);
```