

Neka su date relacije koje čuvaju podatke o bendovima i koncertima:

tip_instrumenta (idTipa, naziv) - Spisak svih tipova instrumenata koje muzičari koriste.

muzicari (idMuzicara, ime, prezime, idTipaInstrumenta) - Spisak svih muzičara sa instrumentima koje sviraju. Jedan muzičar može svirati samo jedan instrument.

bendovi (idBenda, naziv) - Spisak svih bendova.

sastav_benda (idMuzicara, datum1, datum2, idBenda) - Spisak muzičara koji su svirali u bendovima u periodu (**datum1** - **datum2**). Atribut **datum2** ima vrednost NULL ako muzičar i dalje svira u tom bendu.

festivali (idFestivala, naziv) - Spisak svih festivala.

lineup_festivala (idFestivala, godina, idBenda, dan) - Spisak bendova koji su učestvovali na festivalima, po godinama i danima (redni broj dana festivala).

posetioци (idPosetioca, ime) - Spisak svih posetioca.

karte (idFestivala, godina, dan, idPosetioca, cena) - Spisak svih kupljenih karata za određene dane i godine festivala.

1. (3 poena) Kreirati tabelu **VecProdateKarte (id, idFestivala, godina, dan, idPosetioca, cena)** gde kolona **id** ima ograničenje primarnog ključa i potrebno je obezbediti mehanizam u kome nije potrebno navoditi vrednost za kolonu **id** prilikom upisivanja vrednosti u tabelu. Kolonama **idFestivala** i **idPosetioca** postaviti odgovarajuće ograničenje stranog ključa (posebno).

Postaviti triger na unos podataka u tabelu **karte** čiji je zadatak da prilikom unosa u istu, upiše samo one karte koje trenutno ne postoje u toj tabeli. Ukoliko karte već postoje, njih je potrebno upisati u tabelu **VecProdateKarte**.

Primer: Ako se u tabeli karte nalazi sledeće stanje:

	idFestivala	godina	dan	idPosetioca	cena
1	1	1988	1	2	862
2	1	1988	1	6	862

prilikom upisa insert **into karte values (1, 1988, 1, 2, 1000), (1, 1988, 1, 3, 920)** vrednosti **(1, 1988, 1, 2, 1000)** će biti upisane samo u tabelu **VecProdateKarte**, a vrednosti **(1, 1988, 1, 3, 920)** će biti upisane u tabelu **karte**.

2. (3 poena) Kreirati funkciju **dajSastavBenda** koja za prosleđene parametre **idBenda** i **godinu** vraća tabelu koja sadrži idMuzičara, ime i prezime onih muzičara koji su te godine bili u sastavu benda.
3. (3 poena) Kreirati funkciju **dajZaradu** koja za prosleđene parametre **idFestivala** i **godinu** vraća informaciju o količini novca koji je svaki bend zaradio te godine na

festivalu. Bendovi dele ukupnu zaradu od karata te godine datog festivala na “*ravne časti*”.

4. (3 poena) Kreirati funkciju **dajZaraduBendovaPoFestivalima** gde se za svaki bend, svaki festival i godinu njegovog održavanja određuje koliko je taj bend zaradio. Funkcija vraća tabelu koja sadrži sledeće kolone:

idFestivala, godina, idBenda, zaradaBenda

5. (3 poena) Kreirati proceduru **prikaziZaraduPoClanuBenda** koja za prosleđene parametre *idFestivala, godina, idBenda, zaradaBenda* ispisuje koliko je svaki član benda zaradio na tom festivalu u formatu ime, prezime, idBenda, zaradaČlana (koristiti naredbu *print*). Muzičari dele zaradu benda na ravne časti, a tražene podatke je potrebno ispisati ako je muzičar zaradio više od 1000 din (koristiti kursore po potrebi).
6. (4 poena) Kreirati proceduru **prikaziZaradu** koja za svaki festival, godinu njegovog održavanja i bend ispisuje koliko je svaki član zaradio novca (koristiti naredbu *print* i kursore po potrebi).

Napomena: Za svaku kreiranu rutinu je potrebno ispisati primer poziva.