

Domaći zadatak

Opis sistema

Potrebno je napraviti sistem za simulaciju online prodaje artikala. Firma koja poseduje ovaj sistem posreduje između kupaca i prodavnica koje oglašavaju svoje artikale kroz sistem. Sistem brine o oglašavanju artikala, pravljenju porudžbina, dostavi porudžbina od prodavnica do kupca, naplatama.

Prodavnica se nalazi u određenom gradu i oglašava artikale za koje je definisana cena i količina na stanju. Prodavnica može za određeni period definisati popust koji važi za sve njene artikale.

Kupac kreira porudžbinu tako što selektuje artikale iz nekoliko prodavnica i količinu u kojoj ih poručuje. Nakon slanja zahteva sistem proverava da li porudžbina može biti ostvarena, nakon čega vrši naplatu tako što kupcu skida određenu sumu sa virtuelnog računa (račun u okviru ovog sistema). Svaka naplata za porudžbinu se beleži kao posebna transakcija. Nakon naplate porudžbina se automatski šalje kupcu. Porudžbina ima tri stanja: „created“ (kreirana ali još nije kompletirana), „sent“ (kompletna i poslata kupcu), „arrived“ (porudžbina je stigla u grad kupca).

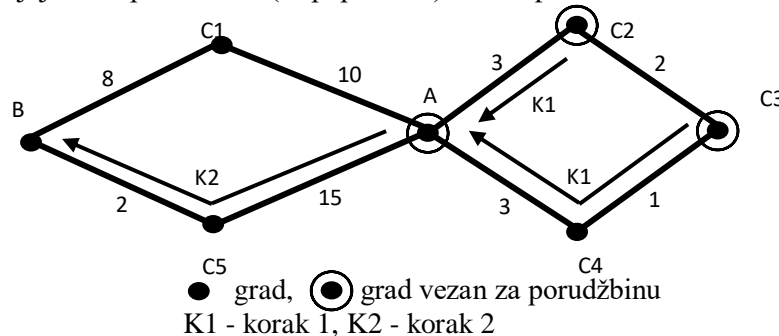
Gradovi su međusobno povezani linijama u kojima saobraćaj teče u oba smera. Smatramo da postoji najviše jedna linija između dva grada. Razdaljina između gradova preko linija izraženima je u danima transporta. S obzirom da jedna porudžbina može biti vezana za više prodavnica transport se obavlja u dva koraka. Prvi korak je sastavljanje porudžbine, kada se artikli iz svih prodavnica transportuju najkraćim putevima u grad prodavnice (grad A) koja je najbliža gradu kupca (grad B). Drugi korak je transport porudžbine od tog grada A do grada B najkraćim putem.

Kada porudžbina stigne u grad kupca sistem automatski prenosi novac prodavnicama koje su vezane za porudžbinu, ali tako da 5% sume zadržava za sebe, a svaku transakciju prema prodavnicama posebno evidentira.

Sistem odobrava dodatni popust od 2% na narudžbinu ukoliko je kupac u proteklih 30 dana ostvario kupovinu u iznosu koji je veći od 10000. Tada se profit sistema umanjuje sa 5% na 3%.

Isplate prodavnicama za pristiglu porudžbinu uraditi okidačem `TR_TRANSFER_MONEY_TO_SHOPS`.

Računanje krajnje cene porudžbine (sa popustima) uraditi procedurom `SP_FINAL_PRICE`.



Poželjno je da onde gde je to moguće referencijalni integriteti budu: ON UPDATE CASCADE, ON DELETE NO ACTION. Iz tog razloga, metoda koja briše određeni red iz tabele ne treba da briše i ostale redove drugih tabela koje referenciraju red za brisanje. Ostale redove treba prvo eksplicitno obrisati drugim metodama (ukoliko ta metoda ne postoji koristiti referencijalni integritet ON DELETE CASCADE).

Svaka kolona Id koja je primarni ključ i nije strani ključ treba da bude IDENTITY kolona.

Koristiti tip `DECIMAL(10,3)` za brojeve sa zarezom. Podrazumevana maksimalna dužina svih tekstualnih kolona u tabelama je 100 karaktera, osim ukoliko za neku kolonu nije drugačije navedeno.

Zahtevi i testiranje

Potrebno je napraviti bazu podataka koja je definisana unutar sekcije “Opis sistema”.

Potrebno je napisati klase sa prefiksom **piggbbbb_** koje implementiraju sve interfejsa i unutar njih implementirati sve metode interfejsa. Klasu možete nadograditi novim metodama, poljima i drugim elementima, ali je neophodno implementirati metode interfejsa jer će one biti pozivane u testovima.

Postoje par javnih testova i nekoliko tajnih testova. Broj ostvarenih poena na domaćem zadatku formira se po sledećoj formuli:

$$OsojeniPoeni=(PoeniSaJavnogTesta+PoeniSaTajnogTesta)*faktorModifikacije$$

Domaći zadatak vredi 20 poena (javni testovi – 10 poena, tajni testovi – 10 poena, faktor modifikacije - vrednost izmedju 0 i 1).

Struktura rešenja

Potrebno je poslati zip arhivu pod nazivom **piggbbbb.zip** na stranici https://rti.etf.bg.ac.rs/domaci/index.php?servis=SAB_1819 unutar koje se nalaze:

1. klase sa prefiksom **piggbbbb_** koje implementiraju interface (treba da se nalaze unutar paketa *student*),
2. modikovanu klasu **studentMain.java**
3. ostale java klase i pakete ukoliko postoje (treba da se nalaze unutar paketa *student*),
4. **piggbbbb.erwin** i **piggbbbb.png**– model baze napravljen u alatu ERwin Data Modeler r9.8,
5. **piggbbbb.sql**– SQL fajl dobijen alatom ERwin Data Modeler + sav dodatni SQL kod (okidači, procedure, funkcije i ostali SQL kod) uključujući i okidače iz **piggbbbb-triggers.sql** i store procedure iz **piggbbbb-stored_procedure.sql**,
6. **piggbbbb-triggers.sql**– SQL fajl sa traženim i ostalim okidačima,
7. **piggbbbb-stored_procedure.sql** –SQL fajl sa traženim i ostalim store procedurama,
8. **piggbbbb.bak**– backup baze podataka.

Napomene

1. Gore pomenuti format za nazivanje fajlova ima sledeću definiciju: p – prvo slovo prezimena, i – prvo slovo imena, gg – poslednje dve cifre iz godine upisa, bbbb – broj indeksa
2. Za komunikaciju sa bazom možete koristiti JPA ili JDBC.
3. Obavezno je korišćenje MS SQL servera