

Za posmatrani sistem smatrati da su u tabeli **VRSTA_RACUNA** naknadno uvedene redundantne kolone *UkupnoUplate* i *UkupnoIsplate*. U tim okolnostima, za jednokratnu upotrebu treba sastaviti java funkciju koja primenom jdbc i resultset-a nad tabelom VRSTA_RACUNA usaglašava navedene redundantne podatke sa podacima u ostatku baze podataka. Smatrati da je konekcija već uspostavljena.

VRSTA_KOMITENTA (IDVrK, Naziv)

MESTO (IDMes, Naziv)

KOMITENT (IDKom, Naziv, Adresa, IDMes, IDVrK)

VRSTA_RACUNA (IDVrR, Naziv, *UkupnoUplate*, *UkupnoIsplate*)

RACUN (IDRac, IDVrR, IDKom)

STAVKA (IDSta, DatumVreme, IDRac)

STAVKA_UPLATE (IDSta, IznosUplate)

STAVKA_ISPLATE (IDSta, IznosIsplate)

MestoKomitenta_VrstaRacuna (IDMes, IDVrR, StanjeIznos)

VRSTA_KOMITENTA (IDVrK, Naziv)
MESTO (IDMes, Naziv)
KOMITENT (IDKom, Naziv, Adresa, IDMes, IDVrK)
VRSTA_RACUNA (IDVrR, Naziv, UkupnoUplate, UkupnoIsplate)
RACUN (IDRac, IDVrR, IDKom)
STAVKA (IDSta, DatumVreme, IDRac)
STAVKA_UPLATE (IDSta, IznosUplate)
STAVKA_ISPLATE (IDSta, IznosIsplate)
MestoKomitenta_VrstaRacuna (IDMes, IDVrR, StanjeIznos)

Java funkcija Redudansa glasi (izostavljene su import klauzule):

```
void Redudansa ( Connection Konekcija )
{
    long   vIDVrR,vIDRac ;
    double vUplateVrsteRacuna,vIsplateVrsteRacuna ;

    ResultSet RezultatRacuni = null ;
    ResultSet RezultatZbirStavki = null ;
    Statement NaredbaRacuna = Konekcija.createStatement() ;
    Statement NaredbaZbiraStavki = Konekcija.createStatement() ;

    PreparedStatement NaredbaVrsteRacuna = Konekcija.prepareStatement
        ("SELECT IDVrR,UkupnoUplate,UkupnoIsplate FROM VRSTA_RACUNA",
         ResultSet.TYPE_FORWARD_ONLY,
         ResultSet.CONCUR_UPDATABLE") ;

    ResultSet RezultatVrsteRacuna = NaredbaVrsteRacuna.executeQuery() ;

    while ( RezultatVrsteRacuna.next() )
    {
        vIDVrR = RezultatVrsteRacuna.getLong("IDVrR") ;

        vUplateVrsteRacuna = 0 ;
        vIsplateVrsteRacuna = 0 ;

        RezultatRacuni = NaredbaRacuna.executeQuery
            ("SELECT IDRac FROM RACUN WHERE IDVrR = " + string(vIDVrR)) ;

        while ( Racuni.next() )
        {
            vIDRac = Racuni.getLong("IDRac") ;

            RezultatZbirStavki = NaredbaZbiraStavki.executeQuery
                ("SELECT SUM(IznosUplate) FROM STAVKA_UPLATE SU, STAVKA S" +
                 "WHERE SU.IDSta = S.IDSta AND S.IDRac = " + string(vIDRac)) ;

            vUplateVrsteRacuna = vUplateVrsteRacuna + RezultatZbirStavki.getDouble(1) ;

            RezultatZbirStavki.close() ;

            RezultatZbirStavki = NaredbaZbiraStavki.executeQuery
                ("SELECT SUM(IznosIsplate) FROM STAVKA_ISPLATE SI, STAVKA S" +
                 "WHERE SI.IDSta = S.IDSta AND S.IDRac = " + string(vIDRac)) ;

            vIsplateVrsteRacuna = vIsplateVrsteRacuna + RezultatZbirStavki.getDouble(1) ;

            RezultatZbirStavki.close() ;
        }

        RezultatRacuni.close() ;

        RezultatVrsteRacuna.updateDouble(2,vUplateVrsteRacuna) ;
        RezultatVrsteRacuna.updateDouble(3,vIsplateVrsteRacuna) ;
        RezultatVrsteRacuna.updateRow() ;
    }

    RezultatVrsteRacuna.close() ;

    NaredbaZbiraStavki.close() ;
    NaredbaRacuna.close() ;
}
```