

Anna kunkin tehtävän vastaus erillisellä paperilla (puolikaskonseptikin käy).
Kirjoita jokaiseen vastauspaperiin kurssin nimi (TIKAPE), tenttipäivä, nimesi (selvästi),
syntymäaikasi, nimikirjoituksesi.

1. Tarkastellaan tauluja:

```
Hotelli(tunnus, nimi, osoite, puhelinnumero)
Huonevaraus(varausnumero, hotelli->hotelli, nimi, osoite, puhelinnumero,
    syntymäaika, alkaen_pvm, kesto, varausaika)
Huone(hotelli->hotelli,huonenumero, varustus)
Majoittuminen((hotelli,huonenumero)->huone, varausnumero->huonevaraus,
    saapui,lähti)
```

Taulujen koot ovat Hotelli 20, Huonevaraus 20000, Huone 1000 ja Majoittuminen X riviä.
Asiakkaiden henkilötiedot tallennetaan varauksen yhteyteen. Vanhoja varauksia säilytetään
jonkin aikaa, samoin tietoja vanhoista majoittumisista.

- a) Minkä kokoinen on luonnollisen liitoksen Hotelli*Hotellivaraus tulos?
Tyhjä. Liitos tehdään sarakkeiden nimi, osoite ja puhelinnumero perusteella, taulussa Hotelli
nämä kuvaavat hotellia, taulussa Huonevaraus asiakasta.
Jos vastaus 20 tai 400 000, niin Op. 20 000 antaa 1p.
- b) Mitä kokoluokkaa on taulu Majoittuminen? Onko se pienempi, yhtä suuri vai suurempi kuin
taulu Huonevaraus. Miksi?
Taulu Majoittuminen on todennäköisesti hieman pienempi kuin huonevaraus. Jokaiselle sen
riville pitää olla huonevaraus. Kaikille tulevaisuuden huonevarauksille ei kuitenkaan ole vielä
majoittumista. Määritely avain sallisi periaatteessa useita majoittumisia / varaus. Mutta
tämän vaikutus on pieni.
Pistemäärä riippuu perustelusta.
- c) Miten suhtautuvat toisiinsa taulujen $\pi_{\text{hotelli,huonenumero}}$ (Huone) ja
 $\pi_{\text{hotelli,huonenumero}}$ (Majoittuminen) rivimäärät?
Ensimmäinen on suurempi tai yhtä suuri kuin jälkimmäinen. Jokainen majoittuminen vaatii
huoneen. Kaikkia huoneita ei ihan välttämättä ole käytetty. Yhtä suuresta kuitenkin 1p
- d) Montako riviä on taulussa Hotelli $\bowtie_{\text{tunnus=hotelli}}$ Majoittuminen?
X.= 2p muut vaihtoehdot 0 p.
- e) Montako riviä on taulussa Hotelli $\bowtie_{\text{tunnus}\neq\text{hotelli}}$ Huone? (10p)
19000.
Vastaus noin 20000 antaa myös täydet pisteet. muuten 0.

Tarkastellaan reseptitietokantaan sisältyviä tauluja (tehtävät 2-4)

```
ruokalaji (tunnus, nimi, helppous, annoksia, valmistusaika)
luokitus (ruokalaji->ruokalaji, luokka)
raaka_aine (ainetunnus, nimi, tyyppi, yksikkö, yksikköhinta)
ainekset (ruokalaji->ruokalaji, ainetunnus->raaka_aine, määrä )
ohje (ruokalaji->ruokalaji, vaiheNumero, kuvaus)
suurin_lajitunnus (isoin)
```

Ruokalaji taulun sarake *annoksia* ilmoittaa kuinka monta annosta reseptistä tulee. *Valmistusaika* ilmoitetaan minuutteina. *Luokitus* taulussa ruokalaji voidaan määrittellä vaikkapa *keitoksi*, *salaatiksi*, *alkuruoaksi* tai *pääruoaksi*. Raaka-aineen *tyyppi* sarakkeessa mahdollisia arvoja ovat esimerkiksi *'kala'*, *'liha'* ja *'kasvis'*. Ainekset taulussa *määrä* ilmoitetaan raaka-aineen *yksikkö*-sarakkeen (arvona esim. *kg*, *hyppysellinen*, *kpl*) mukaisissa yksiköissä. Määrä ilmoittaa koko reseptiin tarvittavan määrän. Taulu *suurin_lajitunnus* sisältää suurimman käytössä olevan ruokalajitunnuksen.

2. Anna seuraavat SQL-kyselyt. Määrittele kyselyiden tuloksille tarkoituksenmukainen järjestys.

SQL-tehtävien arvostelusta:

3p, jos tehtävä on pääpiirteissään oikein (sallitaan pieniä avainsanavirheitä, painovirhetyyppisiä sarakenimivirheitä ja muuta epäoleellista)

2p, jos vastauksesta puuttuu tai on väärin jokin oleellinen osa mutta perusidea on oikein.

Vastaus on liian monimutkainen.

1p, jos vastauksesta löytyy jotain positiivista eikä perusidea ole täysin väärin. Paljoin puuteita

Jokin perusasia kuten yhteenvetotiedon ja detaljien yhdistely väärin.

0p perusidea virheellinen tai kysely ei ole ymmärrettävissä.

Järjestämisen puuttumisesta sakotetaan vain kerran (1p). Samasta virheestä sakotetaan vain kerran.

a) Laadi luettelo alkuruoaksi sopivista keitoista.

```
select tunnus, nimi
from ruokalaji, luokitus keitto, luokitus alkuruoka
where ruokalaji.tunnus = keitto.ruokalaji and
      ruokalaji.tunnus = alkuruoka.ruokalaji and
      keitto.luokka='keitto' and
      alkuruoka.luokka='alkuruoka'
order by nimi.
```

b) Mitä raaka-aineita käytetään johonkin alkuruokaan mutta ei yhteenkään jälkiruokaan?

```
select nimi
from raaka_aine
where ainetunnus in
      (select ainetunnus
       from ainekset, luokitus
       where ainekset.ruokalaji=luokitus.ruokalaji and
            luokitus.luokka='alkuruoka'
      )
and
ainetunnus not in
      (select ainetunnus
       from ainekset, luokitus
       where ainekset.ruokalaji=luokitus.ruokalaji and
            luokitus.luokka='jälkiruoka'
      )
order by nimi
```

c) Listaa ruokalajit, joiden raaka-ainetiedoista puuttuu määrä tai jostain raaka-aineesta yksikkö. (9p)

```
select nimi
from ruokalaji
where tunnus in
      (select ruokalaji
       from ainekset
```

```

where määrä is null or
ainetunnus in
    (select ainetunnus from raaka_aine where yksikkö is null)
)
order by nimi

```

3. Anna seuraavat SQL-kyselyt. Määrittele kyselyiden tuloksille tarkoituksenmukainen järjestys. (yhden rivin vastauksia ei tarvitse järjestää)

a) Mitkä ovat raaka-ainekustannukset yhdessä annoksessa kaalilaatikkaa?

```

select sum (yksikköhinta*määrä/annoksia)
from ruokalaji, ainekset, raaka_aine
where ruokalaji.nimi='kaalilaatikko' and
    ruokalaji.tunnus=ainekset.ruokalaji and
    ainekset.ainetunnus=raaka_aine.ainetunnus

```

b) Mihin ruokalajiin tarvitaan eniten erilaisia raaka-aineita? Anna ruokalajin nimi ja käytettyjen raaka-aineiden lukumäärä.

```

select tunnus, nimi, count(*)
from ruokalaji, ainekset
where ruokalaji.tunnus= ainekset.ruokalaji
group by tunnus,nimi
having count(*) >=
    (select count(*)
     from ruokalaji, ainekset
     where ruokalaji.tunnus=ainekset.ruokalaji
     group by tunnus,nimi)

```

c) Tarkastellaan valmistusaikaa 10 minuutin tarkkuudella. Laadi raportti, josta käy ruokalajityypeittäin (alku-, pää-, ja jälkiruoka) selville montako tiettyyn valmistusaikaryhmään kuuluvaa reseptiä kannasta löytyy. Esimerkiksi 1-9 minuuttia vaativat on yksi ryhmä, 10-19 vaativa toinen ryhmä, jne. Vihje: funktio *trunc(lauseke)* antaa *lausekkeen* kokonaislukuosan. (9p)

```

select luokka, trunc(valmistusaika /10) aika, count(*)
from ruokalaji, luokitus
where tunnus=ruokalaji and
    luokka in ('alkuruoka', 'pääruoka' jälkiruoka')
group by luokka, aika

```

4. Hernekeitosta (ruokalaji 10335) on kehitetty uusi herkullinen muunnelma (tomaattinen hernekeitto), johon hernekeiton aiempien aineiden lisäksi tarvitaan 2 tomaattia. Oletetaan, että tomaatti on raaka-aineena jo olemassa (ainetunnus A332). Mitä operaatioita tarvitaan muunnelman kirjaamiseksi? Ruokalajin valmistusvaiheita ei tarvitse käsitellä. Anna operaatiot myös SQL:llä. (8p)

```

// kasvatetaan laskuria
update suurin lajitunnus
    set isoin=isoin+1;
// kopioidaan hernekeiton perustiedot uuteen ruokalajiin
// vaihetaan tunnus ja nimi - tunnus saadaan suurin lajitunnus taulusta
insert into ruokalaji
    select isoin, 'tomaattinen hernekeitto', helppous, annoksia,
        valmistusaika

```

```

    from suurin_lajitunnus, ruokalaji
    where nimi='hernekeitto';
// kopioidaan hernekeiton raaka-aineet ainekset tauluun,
// hernekeiton tunnus korvataan lisätyn ruokalajin tunnuksella.
insert into ainekset
    select isoin, raaka_aine, määrä
    from suurin_lajitunnus, ainekset
    where ainekset.ruokalaji=10335;
// lisätään aineslistaan tomaatit, oletetaan että yksikkö oli kappale
insert into ainekset
    select isoin, 'A332', 2
    from suurin_lajitunnus;
// vahvistetaan tapahtuma.
commit;

```

Pisteytyksestä:

Jos operaatiot on kerrottu oikein mutta SQL puuttuu 4p.

Jos SQL-lauseet ovat oikein mutta selitys puuttuu niin silti täydet pisteet.

Jos commit puuttuu niin -2.

Jos suurin_lajitunnus taulua ei osata käyttää niin -2

Jos samassa insert tai update lauseessa yritetään muuttaa useaa taulua niin -3

Jos updatella yritetään insertiä niin -2

5. Videovuokraamoon on suunniteltu vuokraustoimintaa varten seuraava taulu:
kopio (kopion_tunnus, elokuvan_tunnus, elokuvan_nimi, ohjaaja, valmistumisvuosi, media,
vuokrausajankohta, vuokraajan_henkilötunnus, vuokraajan nimi, päivähinta)

Jos kopio ei ole vuokralla ovat vuokraajatiedot tyhjiä ja vuokrauspäivä on '1.1.2200'.

- a) Mitä tarkoittaisi käytännössä riippuvuus *ohjaaja* -> *elokuvan nimi* ?

Täsmällisesti tämä olisi: 'Kultakin ohjaajalta on vuokraamon valikoimassa vain yhden nimisiä elokuvia' mutta tulkintaa 'Kultakin ohjaajalta on valikoimassa vain yksi elokuva' voidaan myös pitää oikeana. (2p) 'Ohjaaja yksilöi elokuvan nimen' = 1p, 'Elokuvalla on vain yksi ohjaaja' = 0p.

- b) Miten esittäisit säännön 'Ohjaajalta on tarjolla enintään yksi samana vuonna valmistunut elokuva'?

Ohjaaja, valmistumisvuosi -> elokuvan_tunnus (3p)

1 piste mm. ratkaisusta:

ohjaaja, elokuvan_tunnus-> valmistumisvuosi

elokuvan_tunnus, ohjaaja-> valmistumisvuosi

- c) Olkoon taulussa voimassa riippuvuudet:

- kopion_tunnus -> elokuvan_tunnus,
- kopion_tunnus-> media,
- kopion_tunnus -> päivähinta,
- elokuvan_tunnus -> elokuvan_nimi,
- elokuvan_tunnus -> ohjaaja,
- elokuvan_tunnus -> valmistumisvuosi,
- vuokraajan_henkilötunnus -> vuokraajan_nimi,
- kopion_tunnus, vuokrausajankohta -> vuokraajan_henkilötunnus.

Laadi Boyce-Codd normaalimuodossa olevat relaatiokaaviot (9p)

(4p)

kopio(kopion_tunnus, elokuvan_tunnus->elokuva, media, päivähinta)

elokuva(elokuvan_tunnus, elokuvan_nimi, ohjaaja, valmistumisvuosi)

vuokraaja(vuokraajan_henkilötunnus, vuokraajan nimi)

vuokraus(kopion_tunnus->kopio,vuokrausajankohta, vuokraajan henkilötunnus->vuokraaja)

Koko aineiston avain on kopion_tunnus, vuokrausajankohta, joten avainta varten ei tarvita lisätaulua (tämän maininnan puuttuminen -1p), muuten pisteytys riippuu siitä, miten pahasti tauluja on sotkettu. Tehtävässä ei vaadittu avainten tai viiteavainten merkitsemistä, joten niiden puuttumisesta ei sakoteta..