

Tehtävät 1 - 3 muodostavat kokonaisuuden, joten jos jossain kohdassa tuntuu hankalalta, pyri tekemään mielestäsi käyttökelpoinen ja realistinen ratkaisu. Pääsääntönä on, että kukin tehtävä arvostellaan suhteessa lähtökohtaan.

Tehtävä 1.

Olet ohjelmistosuunnittelijana *CheapAssSoftware Ltd:ssä (CAS Ltd)*, joka valmistaa ja myy erilaisia ohjelmistotuotteita yritysten ja yhdistysten käyttöön. Olet saanut tehtäväksesi suunnitella yhdistyksille myytävän *Memberst* jäsenrekisteriohjelmiston tietokannan. *Memberst*n avulla yhdistykset voivat paitsi pitää kirjaa yhdistyksensä jäsenistä, myös mm. hoitaa jäsenille suunnatun laskutuksen ja laskujen seurannan.

*Memberst*istä tullaan myymään kahta eri versiota, jotka eroavat toisistaan ainoastaan käytävissä olevien toimintojen osalta. Tietokannan rakenne on siis molemmissa sama. Toista versiota voi käyttää yksittäisen ja toista useampien yhdistysten tarpeisiin. Laajemmassa versiossa on myös mahdollista, että yhdistyksellä on alayhdistyksiä, sekä yhdistys ja yritysjäseniä. Huomaa, että yhdistyksen jäsen voi olla jäsenenä myös muissa yhdistyksissä.

Kustakin jäsenrekisteriä pitävästä yhdistyksestä ja alayhdistyksestä tallennetaan yhdistyksen perustiedot, kuten yhdistyksen nimi, Y-tunnus, sähköpostiosoite, osoite, tilinumero, jne. Henkilöjäsenistä tallennetaan perustietoina etunimi, sukunimi, puhelinnumero, sähköpostiosoite ja osoite. Näiden lisäksi henkilöjäsenille voidaan tallentaa tietoa erilaisista ominaisuuksista, kuten järjestysmieheksi, abc-ajokortti, jne. Yritys ja yhdistysjäsenistä tallennetaan nimi, puhelinnumero, sähköpostiosoite, osoite ja yhteyshenkilö (josta tallennetaan samat perustiedot kuin henkilöjäsenestä).

Järjestelmä mahdollistaa erityyppiset jäsenyydet, kuten kannatus-, seniori-, juniori-, aikuis-, kunniajäsenet. Mahdolliset jäsenyydytyypit ovat yhdistyskohtaisia. Jäsenyys voi olla voimassa kausittain (esim. vuosittainen), tai se voi olla ikuinen. Jäsenmaksu laskutetaan jäsenyyden voimassaolokausittain ja se riippuu sekä jäsenyydytyypistä että jäsenen tyypistä (henkilö tai yhteisöjäsen).

Jäsenyyden voimassaolokausi voi olla eri kuin yhdistyksen toimintakausi (joka on usein sama kuin tilivuosi). Tietynä toimintakautena henkilöjäsen voi olla yhdistyksessä tietyssä roolissa, kuten puheenjohtaja, sihteeri, tilintarkastaja, jne. Toimintakauden aikana yhdistys voi järjestää erilaisia tapahtumia, kuten koulutusta ja leirejä, joihin ilmoittautumiset kerätään myös järjestelmään. Osallistujille lähetetään näistä tapahtumista tarvittaessa järjestelmää käyttäen tapahtumakohtainen lasku.

Kaikkilla lähetetyillä laskuilla on yhdistyksen tilinumeron lisäksi myös laskun numeron, päiväs, eräpäivä, loppusumma ja viitenumero. Mikäli laskua ei ole maksettu eräpäivän mennessä, siitä lähetetään karhulasku, johon on mahdollista lisätä (yhdistyskohtaisesti) karhuamis- ja viivästyskuluja. Kun lasku on maksettu, se merkitään tietokantaan.

Järjestelmässä tehdään mm. seuraavanlaisia hakuja ja raportteja:

1. Listataan yhdistyksen jäsenet jäsenyyteittäin siten, että kustakin jäsenestä listataan kaikki perustiedot ja onko jäsen maksanut viimeisimmän kauden jäsenmaksun vai ei.
2. Listataan tietyn jäsenen laskut tietyiltä kaudelta siten, että kustakin laskusta näkyy sekä laskutettu että maksettu summa. Raportin lopussa on lisäksi rivi, jossa on laskettu yhteen koko kauden aikana laskutettu summa sekä kaikkien maksamattomien maksujen summa.
3. Lasketaan kaikkien erityyppisten laskujen keskiarvot tietyltä jäsenyydekaudelta.

Laadi edellä kuvatun järjestelmän käsittekaavio UML:n luokkakaavionotaatiota käyttäen. Käytä periitymistä ja/tai koosteita, jos se on mahdollista. Ominaisuuksista riittää pelkkä nimi.

Tehtävä 2.

a) Laadi tehtävän 1 käsittekaavion pohjalta relaatiokaavio, jossa on merkittynä pää- ja vierasavaimet.

b) Normalisoi tietokannan relaatiot neljanteen normaalimuotoon (kirjoita uudestaan vain kohdat, joita täytyy muuttaa). **Perustele vastauksesi** – myös siinä tapauksessa, että relaatio on jo mielestäsi normalisoitu.

Tehtävä 3.

a) Valitse relaatiokaavioistasi se relaatio, mistä on eniten vierasavainviittauksia ja kerro minkälaiset cheyspolitiikat valitsisit sen vierasavaimille. **Perustele vastauksesi.**

b) Milloin cheysrajoitteiden ylläpitoon voidaan tarvita triggereitä? Anna esimerkki ja **perustele vastauksesi.**

Tehtävä 4.

a) Selitä lyhyesti miten relaatiotietokantojen yhteydessä käsitteellinen suunnittelu eroaa loogisesta suunnittelusta ja looginen suunnittelu fyysisestä suunnittelusta.

b) Selitä lyhyesti, miten tietovarastojen (data warehouse) suunnittelu eroaa operatiivisen järjestelmän tietokannan suunnittelusta.

Tehtävä 5.

a) Selitä esimerkkien avulla (ja perustele), miksi vääränlainen indeksi voi hidastaa (relaatiotietokantojen yhteydessä) tiedonhakua levyiltä.

b) Selitä esimerkkien avulla (ja perustele), miten voit arvioida (relaatiotietokantojen yhteydessä) indeksin vaikutusta tiedonhaun nopeuteen.