

8102300 Johdatus tietokantajärjestelmiin, kevät 2005
Tentti ja välikoe 10.05.2005

TÄRKEÄÄ - LUE TÄMÄ ENSIN!

Tässä samalla paperilla on tehtävät sekä kurssin tenttiin, että toiseen välikokeeseen. Tentti ja välikoe ovat vaihtoehtoiset, voit siis osallistua niistä vain toiseen. Voit itse valita kumpaan osallistut (voit osallistua tenttiin vaikka et olisikaan ilmoittautunut siihen).

Merkitse jokaiseen vastauspaperiin, oletko suorittamassa tenttiä vai välikoetta! Mikäli suoritat välikokeen, tee vain tehtävät 1 ja 2. Mikäli suoritat tentin, tee kaikki tehtävät.

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 lähtökohtaansa.

Tehtävä 1. (6p)

Etelä- ja Väli-Amerikassa toimii Mayojen jälkeläisten nykyaikaistunut kultti, joka tarvitsee vuosittaisia Pelota-kisoja varten tietokantajärjestelmän. Pelin tarkoituksena ei ole sen vähempää kuin varmistaa koko universumin olemassaolo. Konkreettisenä lähitavoitteena on varmistaa, että elintärkeät sateet tulevat ajallaan. Muutoksena muinaiseen peliin, pelataan vuosittain kokonainen sarja pelejä ja ainoastaan koko sarjan voittajajoukkueen kapteeni saa kunnian kuolla uhrauksena jumalille. Tehtävän järjestelmän avulla on tarkoitus mm. hoitaa ilmoittautumiset sekä seurata ja tilastoida vuosittaisten pelisarjojen kulkua, joukkueiden ja yksittäisten pelaajien menestystä, sekä sitä tulivatko sateet ajallaan.

Kutakin Pelota-seuraa voi edustaa useita joukkueita, joista jokaisesta tallennetaan järjestelmään nimi, sähköpostiosoite, maskotin nimi sekä pelaajat ja toimihenkilöt (valmentaja, huoltaja, jne.). Kustakin seurasta tallennetaan lisäksi nimi, kotipaikka ja perustamisvuosi. Pelaajista ja toimihenkilöistä tallennetaan nimien (etunimi, sukunimi, lempinimi) lisäksi kuvaus, syntymäaika, puhelinnumero ja sähköpostiosoite. Eri vuosina joukkueiden kokoonpanot voivat vaihdella ja pelaajat voivat pelata eri pelinumeroilla. Joukkueessa voi olla tiettyä vuonna sinällään vaikka kuinka monta pelaajaa, mutta yksittäisessä pelissä pelaajia on tasan neljä per joukkue. Edes vakavan vammautumisen johdosta pelaajia ei vaihdeta. Näistä neljästä yksi on kapteeni.

Peli pelataan jollain tietyllä areenalla (joka on jollakin paikkakunnalla) ja sillä on tarkka alkamisaika. Toinen joukkueista on kotijoukkue ja toinen vierasjoukkue. Peliä tuomaroi ylipappi, sekä kaksi hänen alaisuudessaan toimivaa apupappia. Heistä tallennetaan samat perustiedot kuin joukkueiden toimihenkilöistä. Peli päättyy kun ylipappi näkee pelissä jumalien antaman merkin tulevasta kohtalosta ja toteaa pelin päättyneeksi tai kun toinen joukkueista menehtyy yksittäinen peli saattaa kestää päiväkausia). Myös päättymisajankohta ja päättymisen syy kirjataan järjestelmään.

Järjestelmään tallennetaan myös kunkin joukkueen (ja yksittäisen pelaajan) tekemät maalit, saadut rangaistukset, sattuneet loukkaantumiset. Olenaiset kysymykset näissä kaikissa ovat milloin, kuka ja missä tilanteessa (esim. rangaistusten ja loukkaantumisten syyt.). Lisäksi tallennetaan pelikohtaisesti mahdolliset muut ylipapin päätökset ja näyt sekä koko vuoden osalta tieto siitä milloin sateet tulivat.

Järjestelmässä tehdään mm. seuraavanlaisia hakuja ja raportteja:

1. Listataan tietyn pelaajan pelaamat pelit ajan mukaan järjestyksessä. Jokaisesta pelistä tulostetaan pelipaikka, alkamisajankohta, päättymisajankohta, koti- ja vierasjoukkueiden nimet, sekä tehdyt koti- ja vierasmaalit.
2. Listataan tietyn pelin tapahtumat (aloitus, maalit, loukkaantumiset, rangaistukset, jne.) aikajärjestyksessä. Kustakin tapahtumasta halutaan tietää mitä tapahtui, milloin, kuka/ketkä olivat osapuolina ja missä tilanteessa (vapaamuot. selitys).
3. Lasketaan joukkueittain kaikkien pelattujen pelien maalimäärien keskiarvot vuosittain (== yksi luku per vuosi). Tulosteeseen tulee siis joukkueen nimi, vuosi ja tietyn vuoden pelien maalimäärien keskiarvo.

Laadi edellä kuvatun järjestelmän käsitekaavio UML:n luokkaavionaatiota käyttäen. Käytä periytymistä ja/tai koosteita, jos se on mahdollista. Ominaisuuksista riittää pelkkä nimi.

Tehtävä 2. (6p)

Laadi tehtävän 1 käsitekaavion pohjalta vähintään kolmannessa normaalimuodossa oleva relaatiokaavio, jossa on merkittynä pää- ja vierasavaimet. Perustele normaalisuus.

Tehtävä 3. (6p)

Laadi tekemäsi relaatiokaavion pohjalta standardin mukaisena SQL:nä seuraavat tehtävässä 1 esiintyneet kyselyt:

- a) Kysely 1 (1p)
- b) Kysely 2 (1p)
- c) Kysely 3 (1p)
- d) Anna joukkueen X valmentajalle oikeudet lisätä, muuttaa ja poistaa oman joukkueen tietoja (ei pelaajien henk. koht. tietoja) (1p)
- e) Luo indeksi, joka tehostaa kyselyä 3. Perustele vastauksesi! (2p)

Tehtävä 4. (6p)

Selitä jokainen kohta maksimissaan kymmenellä rivillä:

- a) Miksi näkymän (view) kautta ei ole aina mahdollista päivittää tietoja?
- b) Avainehdokka (candidate key)
- c) Ulkoliitos (Outer Join)
- d) Mitä tarkoittaa tietokannan eheys?
- e) Mikä on tietokantojen yhteydessä tapahtuma (transaction)?
- f) Miksi hakemisto (Index) nopeuttaa kyselyjä?

Voit selvittää selitystäsi esimerkillä (siis sisältäen siihen kymmenen riviin).

Muistakaa käydä antamassa kurssista palautetta O-infossa! Hyvää kesää!