**Andmebaas Triiniksi tantsuklubile**

**Töö valdkond**

Töö tulemuseks on andmebaas tantsugrupile. Antud rühmatöö keskendub konkreetsele tantsugrupile Triiniks, kus üks rühmaliikmetest (Veronika) ka juhendaja on.

**Antud valdkonna probleem**

Raske on omada head ülevaadet kõikidest rühmadest, liikmetest ja võistlustest ilma andmebaasita. Praegu märgitakse õpilaste kohalolu lihtsalt paberile ja saavutuste ülesmärkimiseks pole samuti head süsteemi. Ka lapsevanemate kontaktandmed peavad kuskil eraldi olema. Palju lihtsam oleks, kui kõik tantsurühmadega seotud andmed oleksid ühes kohas, kust neid lihtsalt kätte saab.

Samuti on keeruline teha näiteks hooaja kohta kokkuvõtteid ja analüüse (näiteks kes ja kui palju trennis käis), sest paberkandjal info töötlemine on ajamahukas ja vead on kerged tekkima.

**Probleemi lahendus**

Andmebaasiga on pidev juurdepääs varasemate treeninghooaegade andmetele (palju trennides käidi, mis kohti võistlustel saavutati). Kui näiteks soovitakse tunnustada aktiivsemaid õpilasi, siis saaks päringuga leida kiirelt kõige aktiivsemalt trennis käinud õpilased. Samuti saab kiirelt kätte näiteks õpilaste või nende vanemate kontaktid.

**Andmebaasi kasutajad**

Andmebaasi kasutavad tantsurühmade juhendajad. Nad saavad sinna märkida igas trennis kohalolijad ning saavutused võistlustel.

**Graafiline mudel**

Graafilisest mudelist eksporditud SQL-laused on lisatud faili lõppu.

**Mudeli kirjeldus**

Graafiline mudel on koostatud Sybase PowerDesigner-iga. Õpilaste kohta hoiame andmebaasis nende eesnime, perenime, aadressi, telefoni, e-maili, isikukoodi ja sünnikuupäeva. Lapsevanemal on atribuutideks on eesnimi, perenimi, aadress, telefon, e-mail ning kommentaar, kuhu on soovi korral lisatud lisainfo ja/või teine kontakt. Iga õpilane on seotud ühe lapsevanemaga.

Õpilasi ja Trenne ühendab Kohalolu, milles on kirjas, kas õpilane oli antud trennis või mitte. Trennidel on asukoht, toimumisaeg ning juhendaja.

Võistlustel on asukoht, aeg, nimi ning ühel võistlusel võib osaleda mitu rühma, aga nad saavutavad erinevad kohad. Rühma ja võistlust seob Võistleb, mille atribuudiks on saavutatud tulemus.

Rühmal on nimetus ja vanuseaste ning kõik õpilased on seotud mingi rühmaga. Rühmad on seotud ka trennidega (rühmal on palju trenne, aga iga trenn on seotud ühe kindla rühmaga).

Trennidega on seotud ka juhendajad. Juhendajate kohta hoiame andmebaasis järgmiseid andmeid: eesnimi, perenimi, aadress, telefon, e-mail ja isikukood. Ühel trennil on üks juhendaja, kuid üks juhendaja võib anda mitut trenni.

**Relatsiooniline mudel**

Õpilased(**ID**, eesnimi, perenimi, vanus, isikukood,aadress, telefon, e-mail, lapsevanem\_ID)

Juhendajad(**ID**, eesnimi, perenimi, aadress, telefon, e-mail, isikukood)

Rühmad(**ID**, nimetus, vanuseaste)

Trennid(**ID**, asukoht, toimumisaeg, juhendaja\_ID, rühm\_ID)

Kohalolu(**õpilane\_ID, trenn\_ID**)

Lapsevanemad(**ID**, eesnimi, perenimi, e-mail, telefon, aadress, kommentaar)

Võistlused(**ID**, asukoht, aeg, nimi)

Võistleb (**võistlus\_ID, rühm\_ID,** tulemus)

KuulubRühma(**rühm\_ID, õpilane\_ID**)

**Kolmas normaalkuju ja funktsionaalsed sõltuvused**

Kõik relatsioonid on kolmandal normaalkujul, kuna ei leidu sellist kolmikut, mis rikuks kolmanda normaalkuju nõudeid. Relatsioonidel on võtmeteks ID-d, mis katavad ka kõik funktsionaalsed sõltuvused.

**Päringud**

Kasutatud on SQL verisooni 17.0.4

**1)** Võistlustel saavutatud tulemused konkreetsel aastal. Parameetriks tuleb protseduurile anda soovitud aastaarv.Väljastatakse võistluse nimi, aeg, asukoht, tulemus ja rühma nimi.

CREATE PROCEDURE sp\_tulemused (IN a\_aasta integer)

RESULT (Võistlus varchar(30), Aeg date, Asukoht varchar(30), Tulemus integer, Rühm varchar(30))

BEGIN

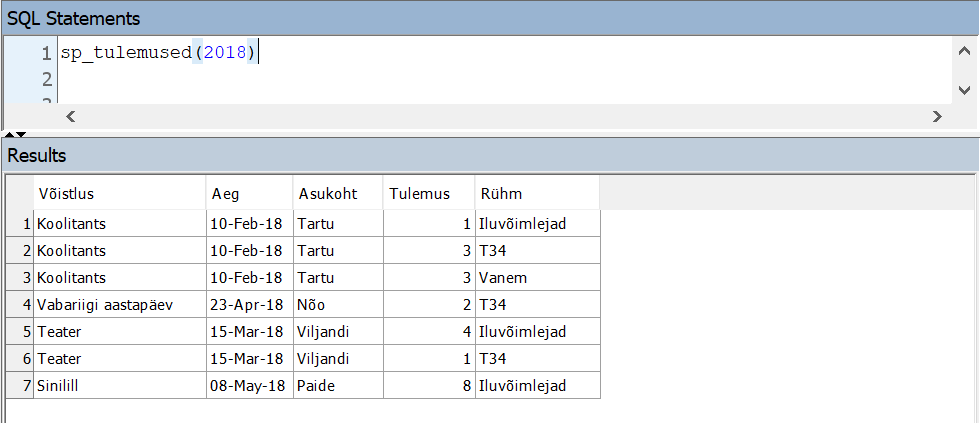
SELECT nimi, aeg, asukoht, saavutatud\_tulemus, nimetus

FROM võistlused, võistleb, rühmad

WHERE Võistleb.Rühm\_id = Rühmad.id AND võistleb.võistlus\_id = võistlused.id

AND year(võistlused.aeg) = a\_aasta

END



**2)** Top 3 trennides käijad konkreetsest rühmast. Parameetriks rühma nimi. Väljastatakse kolme kõige rohkem trennis käinud õpilase nimed ning trennide arv.

CREATE PROCEDURE sp\_top3\_aktiivsus(IN a\_rühm varchar(30))

RESULT(Nimi varchar(30), Trenne integer)

BEGIN

SELECT TOP 3 eesnimi + ' ' + perenimi as nimi, count()

FROM Õpilased, Kohalolu, Trennid, Rühmad

WHERE Õpilased.ID = Kohalolu.Õpilane\_ID

AND Kohalolu.Trenn\_ID = Trennid.ID

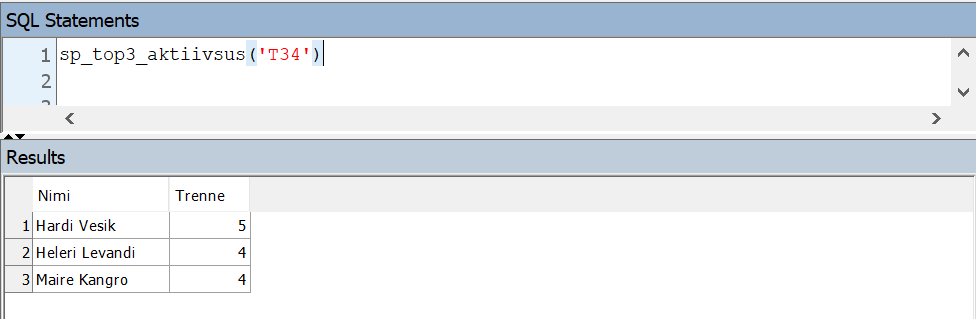
AND Trennid.Rühm\_ID = Rühmad.ID

AND Rühmad.Nimetus = a\_rühm

GROUP BY nimi

ORDER BY count() DESC

END



3) Trennis osalenud õpilased. Parameetriteks antakse trenni toimumisaeg ning rühma nimi ja väljastatakse õpilaste eesnimed ja perenimed.

CREATE PROCEDURE sp\_trennis\_osalejad(IN a\_aeg varchar(30), a\_rühm varchar(30))

RESULT (Eesnimi varchar(30), Perenimi varchar(30))

BEGIN

SELECT Õpilased.Eesnimi, Õpilased.Perenimi

FROM Kohalolu, Õpilased, Trennid, Rühmad, On\_rühmas

WHERE Õpilased.id = Kohalolu.Õpilane\_ID

AND Trennid.ID = Kohalolu.Trenn\_ID

AND Trennid.Rühm\_ID = On\_rühmas.Rühm\_ID

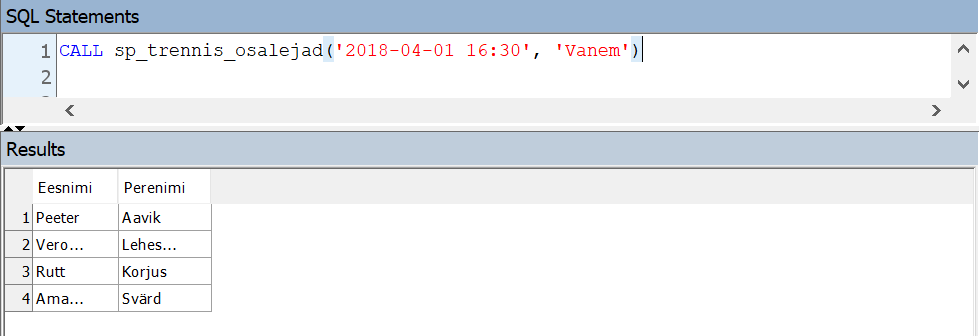
AND On\_rühmas.Rühm\_ID = Rühmad.ID

AND On\_rühmas.Õpilane\_ID = Õpilased.ID

AND Trennid.Toimumisaeg = DATETIME(a\_aeg)

AND Rühmad.Nimetus = a\_rühm

END



**4)** Lapsevanema telefoninumber õpilase nime põhjal.

CREATE PROCEDURE sp\_lapsevanema\_telefon(IN a\_õpilane varchar(30))

RESULT (Nimi varchar(30), Telefon integer)

BEGIN

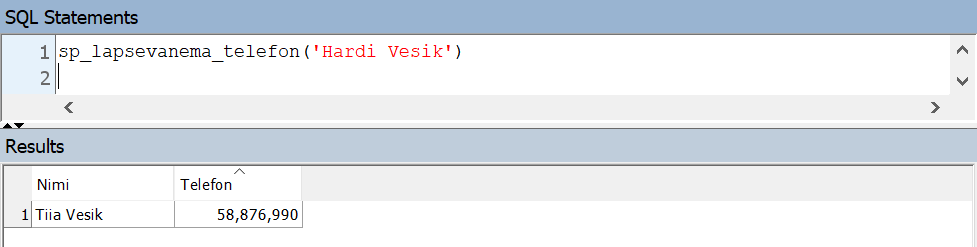
SELECT Lapsevanemad.Eesnimi + ' ' + Lapsevanemad.Perenimi, Lapsevanemad.Telefon

FROM Lapsevanemad, Õpilased

WHERE Õpilased.Lapsevanem\_ID = Lapsevanemad.ID

AND Õpilased.Eesnimi + ' ' + Õpilased.Perenimi = a\_õpilane

END



5) Rühmade liikmete arvu väljastamine.

CREATE PROCEDURE sp\_rühmas\_õpilasi()

RESULT (Nimetus varchar(30), Liikmeid integer)

BEGIN

SELECT Nimetus, count() FROM Rühmad, On\_rühmas

WHERE Rühmad.ID = On\_rühmas.Rühm\_ID

GROUP BY Nimetus

ORDER BY Nimetus ASC

END

