

# **Jalgpalli Euroopa Meistrite Liiga andmebaas**

Georg Klettenberg, Tambet Osman

8. rühm

## **I ETAPP**

### **I Valdkond**

#### **A. Kirjeldus:**

Jalgpall on ilmselt kõigile tuttav oma populaarsuse tõttu terves maailmas. Pea igas riigis on oma jalgpalliliiga. Euroopa märkimisväärsim turniir on Meistrite Liiga, sellele järgnevad Euroopa liiga ja Konverentsi liiga, kuhu muuseas iga-aastaselt Eesti parimas vormis olevad klubid ka pretendeerida saavad. Kuid tippude tipp on siiski Meistrite Liiga, sinna kvalifitseeruvad Euroopa tippliigadest esimesed 4 parimat, lisaks need meeskonnad, kes tõusevad sinna Euroopa liigast. Euroopa tippjalgpalliliigadeks peetakse Hispaania, Inglismaa, Itaalia, Saksamaa ning Prantsusmaa jalgpalliliigasid. Lisaks saavad eritingimustel kohti veel Hollandi, Šveitsi, Belgia ning Slovakkia kõrgliigade meeskonnad. Sellest aastast on Meistrite liigal uus formaat, kus kokku pannakse 36 kvalifitseerunud meeskonda eri riikidest Euroopas, ning jagatakse need üheksasteks gruppideks kus iga meeskond mängib 8 mängu, kusjuures punktiarvestust peetakse ikkagi 36 meeskonna peale kokku. Varasemalt olid loositud 8 neljast alagruppi, kust edasi said igast alagrupist esimesed kaks meeskonda, siis loositi 16 meeskonna seast turniiriformaadis mängud.

Eesmärgiks on luua andmebaas, mis toetukski antud kirjeldusele, selles oleksid siis eelkõige mängijad, meeskonnad, liigad, punktid, positsioon Meistrite liiga tabelis, mängud jne.

**Mõisted:**

Tiim - meeskond.

Liiga - organiseeritud võistlussüsteem, kus kindel hulk meeskondi mängib hooaja jooksul omavahel teatud arv mängusid, selles peetakse arvestust (võit 3 punkti, viik 1, kaotus 0).

Positsioon - asukoht liigatabelis.

Kodumeeskond, võõrsilmeeskond - meeskond, kes mängib oma kodulinnas (kodustaadionil), meeskond, kes mängib võõral staadionil, eemal oma kodulinnast.

**B. Valdkonna probleem, kitsaskohad, lahendus.**

Andmebaasi abil oleks võimalik teha kiiresti statistikat ning kätte saada üldiseid andmeid lihtsamalt kätte, praegu on probleemiks see, et üldine statistika jalgpalli kohta pole piisavalt kergesti kättesaadav. Iga mängija kohta saab talletada info: väravad, söödud jne. Samuti oleks andmebaasis olemas mängude tabel. Veel saaks kuvada ka teabe kiiresti meeskonna järgmise mängu kohta, mida samuti ei ole märganud tänapäevastes sarnast infot kajastavates rakendustes (andmebaasides). Samuti oleks erinevus tänapäevaste rakendustega see, et me kuvaksime mängude tabelis selle mängu kohtunikku samuti. Terve andmebaas oleks üles ehitatud lihtsusel, et kõikidel allpool välja toodud kasutajatel oleks võimalik oma eluvaldkonna raames neid andmeid arukalt ära kasutada.

**C. Kasutajad**

Eelkõige kõik jalgpallihuvilised (kes parasjagu ei ole mängijad antud liigas) seejärel meeskonnad, mängijad ise, sponsoretttevõtted, kihlveokontorid, hasartmänguhuvilised.

Veel võiksid kasutajate hulka kuuluda kogu jalgpallimeedia, spordiajakirjanikud, andmeteadlased, analüütikud. Lisaks neile erinevad jalgpalliliigade juhid.

## **D. Ärireeglid:**

Iga meeskond saab osaleda ühe korra Meistrite Liigas täpselt ühe jalgpallihooaja jooksul st (September-Mai/Juuni)

Ühe värava saab lüüa täpselt üks mängija, söödu andmisega samamoodi.

Üks mäng toimub täpselt kahe meeskonna vahel

Mängu võib võita täpselt 1 meeskond või see võib lõppeda viigiga

Iga meeskond mängib Meistrite Liiga algfaasis (enne turniiriformaati) täpselt 8 mängu

Parimad 8 meeskonda pääsevad otse play-offi, meeskonnad kohtadel 9-24 mängivad lisaringi ning kohtadel 25-36 langevad turniirist välja.

Igal mängul on täpselt üks peakohtunik.

Igal tiimil on oma kodustaadion.

Igal riigil on täpselt 1 jalgpalliliiga (meie minimaailmasse kitsendus)

Igal mängijal meeskonnast on unikaalne nimi, ei saa olla kahte samanimelist mängijat ühes meeskonnas.

## **Aarsus:**

Üks Meistrite liiga meeskond saab eksisteerida ühe korra tiimide tabelis ja tiim saab eksisteerida ühe korra Meistrite liiga tabelis. (1:1)

Ühes tiimis olla mitu mängijat aga ühel mängijal saab olla üks tiim. (1:n)

Üks meeskond saab olla korraga ainult üks kodumeeskond. (1:1)

Üks meeskond saab olla korraga ainult üks külalismeeskond. (1:1)

Tiim saab olla ainult ühes liigas (rahvusliigas) aga liigas saab olla mitu tiimi. (1:n)

Üks mängija saab olla pärit ühest riigist aga ühest riigist saab olla pärit mitu mängijat (1:n)

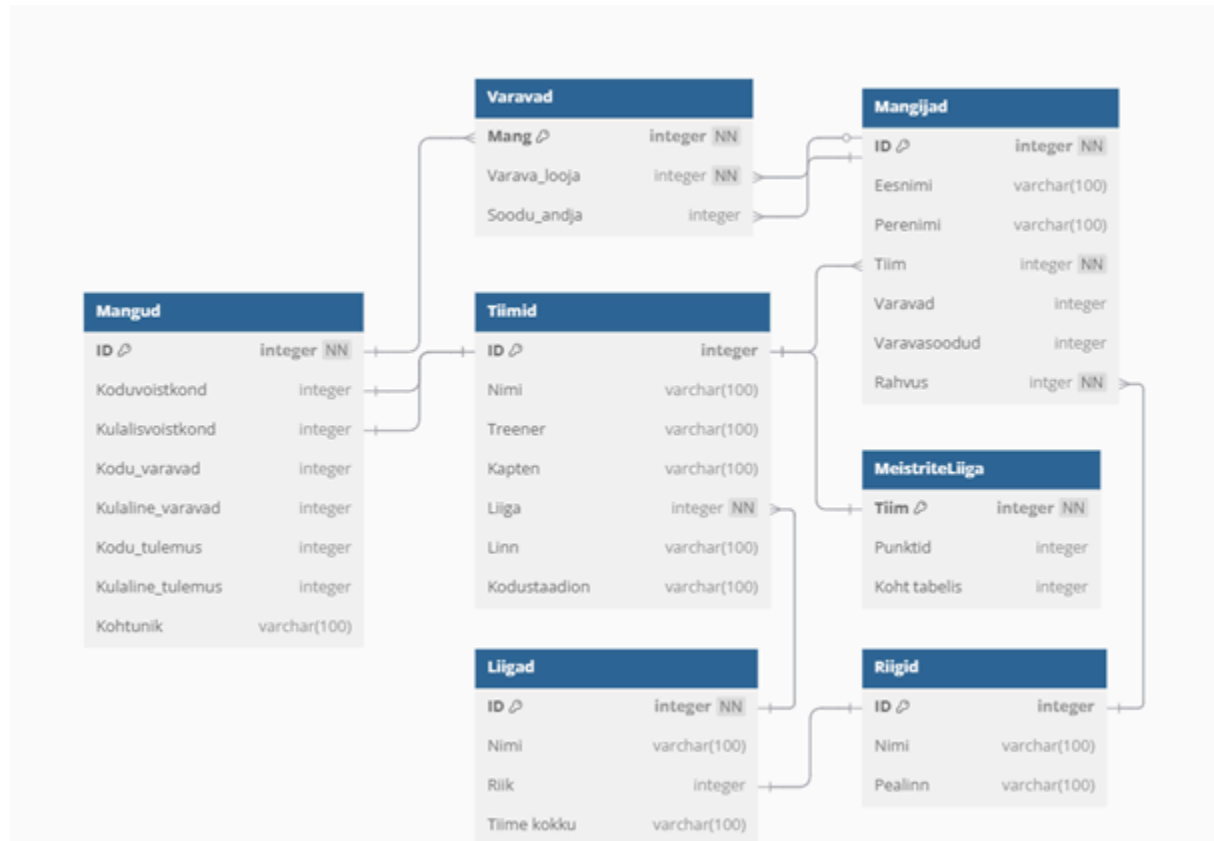
Üks liiga saab olla ühes riigis ja ühes riigis saab olla üks liiga (1:1) (liiga, millest pääsetakse Meistrite liigasse)

Ühel mängul saab olla mitu väravat aga see üks värav saab olla ühes mängus (1:n)

Üks mängija saab anda mitu väravasöötu aga väravasöödul on maksimaalselt 1 sooritaja (1:n)

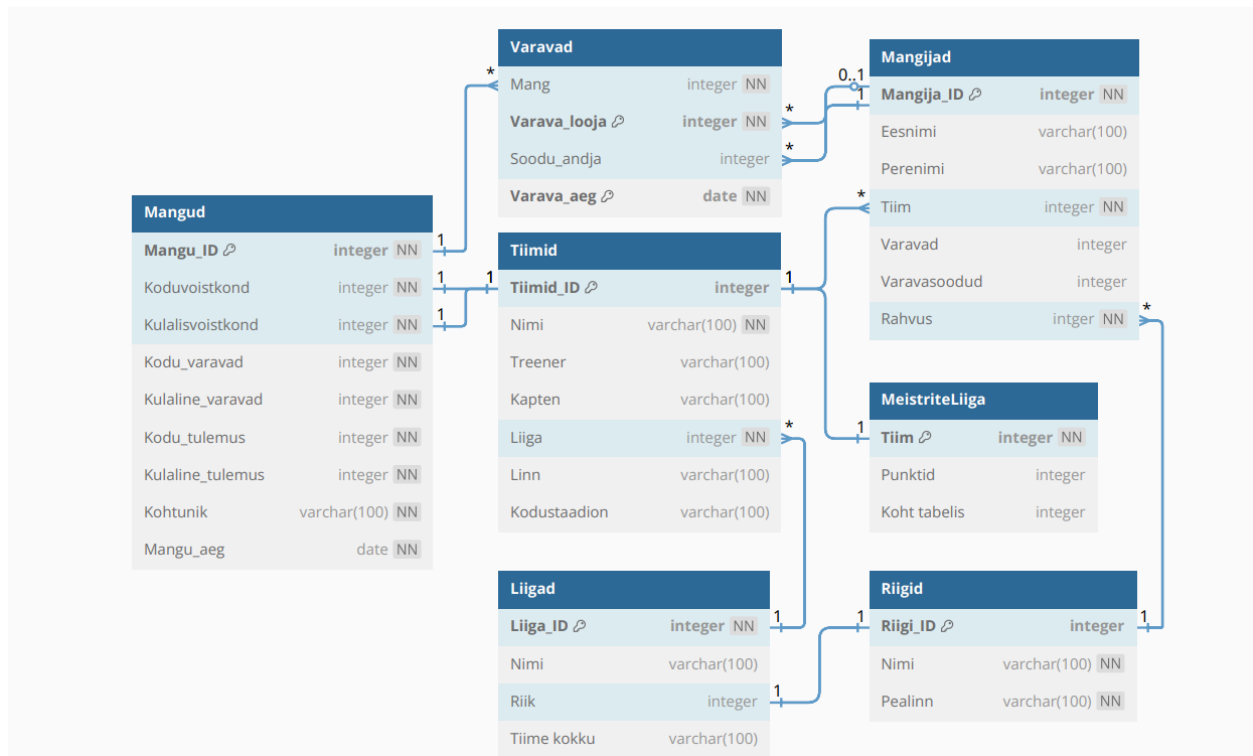
Üks mängija saab lüüa mitu väravat aga ühel väraval on alati üks lööja. (1:n)

## II Olem seose diagramm



Joonis 1. Olem seose diagramm

## II Etapp



Joonis 2. Andmebaasi relatsiooniline kuju(algne)

Relatsioonid ilma kunstlike võtmeteta (primaarvõti **paksus** kirjas):

- Mangud(**koduvoistkond**, **kulalisvoistkond**, kodu\_tulemus, kulaline\_tulemus, kohtunik, **mangu\_aeg**)
- Varavad(mang, **varava\_looja**, soodu\_andja, **varava\_aeg**)
- Mangijad(**eesnimi**, **perenimi**, **tiim**, rahvus)
- Meistriliiga(tiim, punktid, **koht\_tabelis**)
- Riigid(**nimi**, pealinn)
- Liigad(nimi, **riik**)
- Tiimid(**nimi**, **treener**, kapten, liiga, linn, kodustaadion)

## Normaalkujude kontroll:

1. Mangud(**koduvoistkond**, **kulalisvoistkond**, kodu\_tulemus, kulaline\_tulemus, kohtunik, **mangu\_aeg**)  
Koduvoistkond, mangu\_aeg, kulalisvoistkond → kodu\_tulemus  
Kulalisvoistkond, mangu\_aeg, koduvoistkond → kulaline\_tulemus  
Koduvoistkond, kulalisvoistkond, mangu\_aeg → kohtunik  
*Koduvõistkond, külalisvõistkond ja mängu aeg määravad kodu tulemuse. Külalisvõistkond ning koduvõistkond ja mängu aeg määravad külalise tulemuse (sest täpselt samal ajal ei saa olla sama võistkonna poolt samat tulemust teise võistkonna vastu) ning koduvõistkond, külalisvõistkond ja mängu toimumise aeg määravad ära kohtuniku, sest samal ajal ei saa olla samade võistkondade vahel kusagil mujal sama nimega kohtunikku.*  
*Relatsioon on 3. normaalkujus, sest ükski sekundaarne atribuut ei sõltu primaarvõtme alamhulgast ning ei esine ühtegi transitiivset sõltuvust.*
2. Varavad(mang, **varava\_looja(fk)**, soodu\_andja, **varava\_aeg**)  
Varava\_looja, varava\_aeg → mang  
Varava\_looja, varava\_aeg → soodu\_andja  
*Värava lööja ning värava ajaga saame teada, mis mängus värav löödi, sest täpselt sama ajal ei ole võimalik, et sama värava lööja mingis muus mängus ka värava lööb, samuti saab nii teada ka söödu andja (kui värava lööja sai värava löömiseks söödu). Söödu andjaga ei pruugi värava lööjat teada saada, sest igal väraval ei ole söödu andjat.*  
*Relatsioon on 3. Normaalkujus, sest ükski sekundaarne atribuut ei sõltu primaarvõtme alamhulgast ning meil ei esine ühtegi transitiivset sõltuvust.*
3. Mangijad(**eesnimi**, **perenimi**, **tiim(fk)**, rahvus)  
Eesnimi, perenimi, tiim → rahvus  
*Kuna primaarvõti on kolme atribuudiga ja terve relatsioon koosneb neljast, siis meil on automaatselt 3. normaalkuju garanteeritud. Tiim on välisvõtmeks tiimide tabelile ning rahvus välisvõtmeks riikide tabelile. Meie minimaailmas on reegel, et ühes tiimis ei saa mängida kahte samanimelist mängijat.*
4. MeistriteLiiga(tiim, punktid, **koht\_tabelis**)  
*Tabeli primaarvõtmeks on koht\_tabelis, kuna 2 tiimi ei saa omada täpselt sama kohta tabelis, kuigi neil saab olla sama arv punkte (koha määravad siis väravate vahe, teenitud kaardid, omavahelised mängud jne). Punktide arv niisama tiimi nime välja ei anna, seega on tabel kolmandal normaalkujul.*  
*Punktid ja koht\_tabelis on tegelikult arvutatavad, kuid singulaarse atribuudiga tabel oleks kentsakas ning kasutajal oleks mugavam vaadata, kui ta näeb kohe tiimi punkte ja kohta tabelis, siin on tiim välisvõtmeks tiimide relatsioonile seega siin relatsioonis ei olegi otseselt oma primaarvõtit ning ta on meil kolmandal normaalkujul.*

5. Riigid(**nimi**, pealinn)

Nimi → pealinn

*Kuna meie primaarvõti on atomaarse väärtusega, siis see garanteerib meile kohe kolmanda normaalkuju.*

6. Liigad(**nimi**, **riik**)

Riik → nimi

*Kuna meie minimaailma reegel oli, et igas riigis on ainult üks jalgpalliliiga, siis teades riiki määrab see meile kohe ka liiga ning kuna primaarvõti on atomaarne, siis ongi meil kolmas normaalkuju käes*

7. Tiimid(**nimi**, treener, kapten, **liiga**, linn, kodustaadion)

Nimi, liiga → treener, kapten, linn, kodustaadion

*Kuna sama nimega võistkondi on olemas maailmas, kuid üheski liigas mitte, siis just tiimi nimi ja liiga muudavad antud relatsiooni unikaalseks. Tiimi nime ja liiga kaudu on meil võimalik kätte saada linn, kus tiim tegutseb, tema kodustaadion. Samuti nende abil saame teada kapteni ja treeneri, sest ühel võistkonnal on ainult üks treener ja üks kapten ja kuna need kaptenid ja treenerid on samas liigas ja tiimis, siis ei saa tekkida varianti, et kuskil mujal on täpselt sama nimega treener ja kapten ning võistkond.*

*Kuna ei esine sekundaarsete atribuutide sõltuvust primaarvõtme alamhulgast ning kõik atribuudid on atomaarsed ja transitiivseid sõltuvusi ei esine, on relatsioon kolmandal normaalkujul.*

## Relatsioonid kunstlike võtmetega (ID-d lisatud)

- Mangud(**mang\_id**, koduvoistkond, kulalisvoistkond, kodu\_tulemus, kulaline\_tulemus, kohtunik, mangu\_aeg)
- Varavad(mang(fk), **varava\_looja**, soodu\_andja, **varava\_aeg**)
- Mangijad(**mangija\_id**, eesnimi, perenimi, tiim(fk), rahvus(fk), isikukood)
- MeistriteLiiga(tiim(fk), punktid, **koht\_tabelis**)
- Riigid(**riigi\_id**, nimi, pealinn)
- Liigad(**liiga\_id**, nimi, riik(fk))
- Tiimid(**tiimi\_id**, nimi, treener, kapten, liiga(fk), linn, kodustaadion)

## MUUDATUSED VASTAVALT KOMMENTAARIDELE JA PRAKTIKUMIJUHENDAJA TAGASISIDELE:

MeistriteLiiga tabelis tiim muutetud primary keyst foreign keyks ja primary keyks lisatud koht tabelis.

Kõikide atribuutide kirjalpilt muudetud läbivalt väikesteks tähtedeks

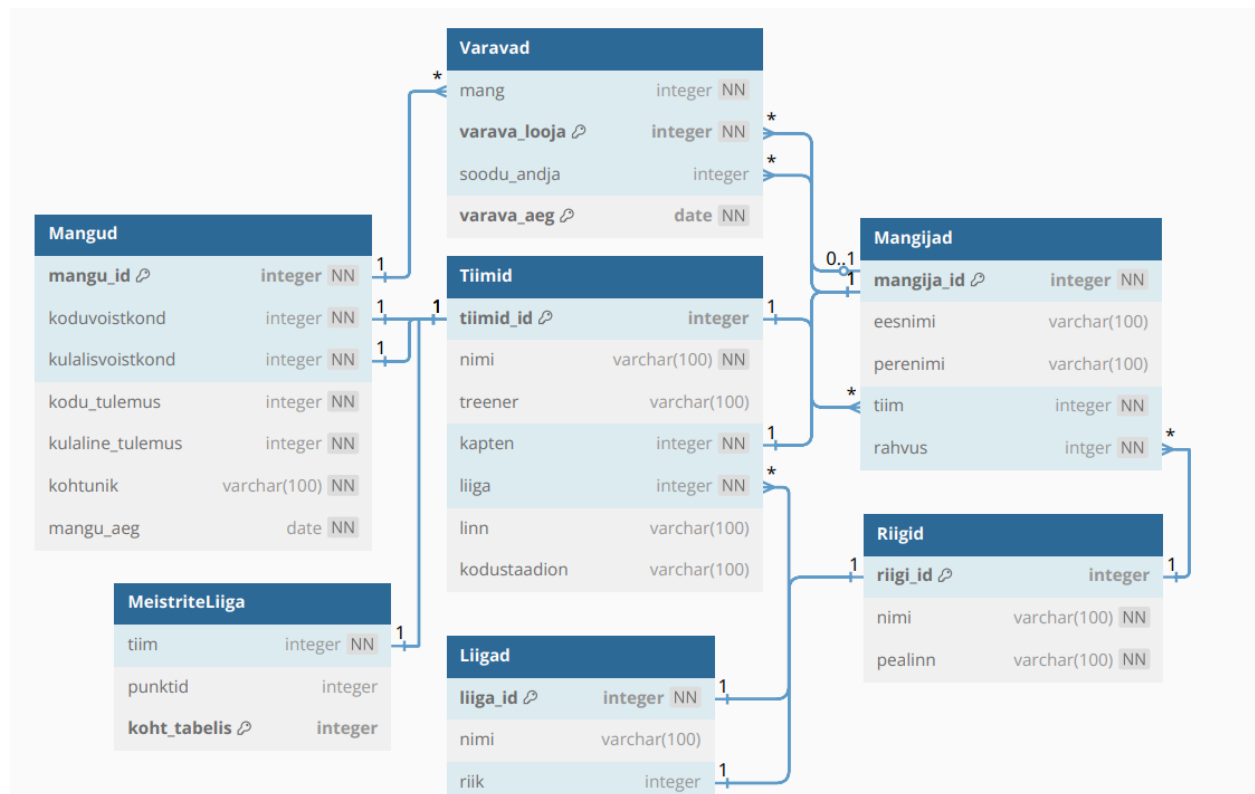
Kapten sai seotud mangijate tabeliga mangija id kaudu

Mangijate tabelis varavad ja varavasoodud eemaldatud (arvutatavad varavate tabeli kaudu)

Eemaldatud kodu varavad külaliste varavad mang tabelist, need on arvutatavad

Eemaldatud tiime kokku liigade tabelist, arvutatav

Lisatud juurde kaks ärireeglit (lillas kirjas)



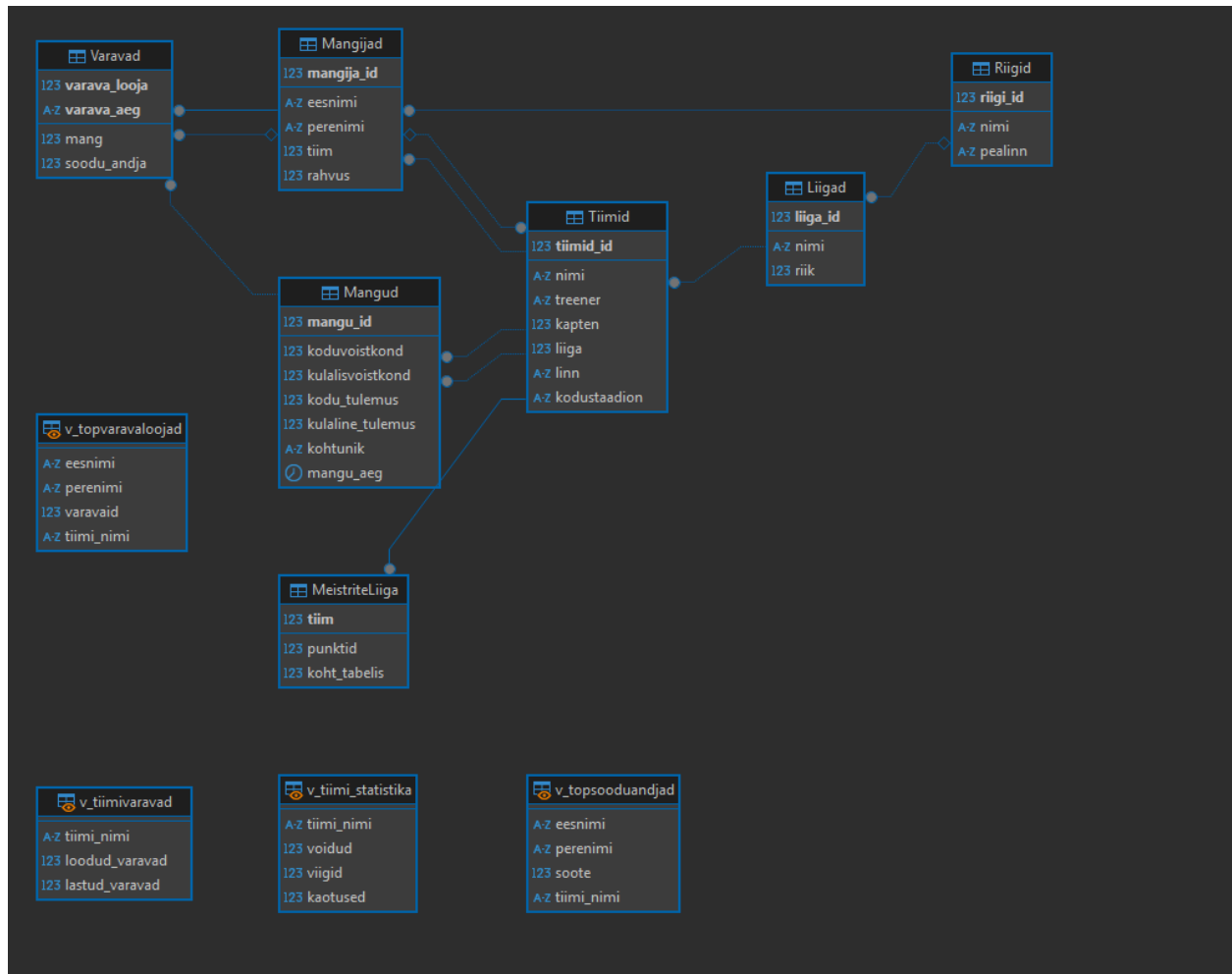
Joonis 3. Andmebaasi relatsiooniline kuju (lõpp)



## III Etapp

### Lühikirjeldus:

Lõime andmebaasi kasutades *db.diagramm.io* tarkvara. Saime kenasti üle kantud eelmisel leheküljel kuvatava relatsioonilise andmebaasi kuju. Lisasime täpselt originaalse arvu korteeže igasse relatsiooni. Tiimide tabelisse sai 32 Euroopa tiimi, mängijate tabelisse 640 mängijat ja umbes kolme-nelja kordselt sama palju andmeid väravate ning mängude tabelisse. Meistriliiga tabelisse pidime genereerima triggeri, mis aitab uuendada andmeid kui lisada mäng (muudab tiimi positsiooni tabelis nt kui võidab mängu liigub kõrgemale jne) aga suures pildis olid triggerid just selle jaoks, et saaks genereerida tiimile koha meistrite liiga tabelis. Samuti sai lisatud 10 suuremat Euroopa jalgpalliliigat erinevatest riikidest ning lisaks neile sai lisatud riikide tabel, kus on 32 riiki.



## Loodud vaated:

- 1) Topvaravaloojad: Vaade on loodud eesmärgiga saada kiirelt kätte statistika parimate väravaküttide jaoks, see verstaapost oli meil ka ärireeglites kehtestatud (kiire statistika). Vaade kuvab mängija eesnime ja perenime, löödud väravate arvu ning tema tiimi.

```
CREATE OR REPLACE VIEW v_topvaravaloojad AS SELECT
eesnimi, perenimi, count(varava_looja) AS varavaid, t.nimi AS tiimi_nimi
FROM public."Mangijad" m
LEFT JOIN public."Varavad" v ON m.mangija_id = v.varava_looja
LEFT JOIN public."Tiimid" t ON m.tiim = t.tiimid_id
GROUP BY m.eesnimi, m.perenimi, t.nimi
ORDER BY varavaid DESC;
```

Tulemus:

	A-Z eesnimi	A-Z perenimi	123 varavaid	A-Z tiimi_nimi
1	Stefan	Santos	3	Liverpool
2	Nikita	Costa	3	FC Barcelona
3	Pavel	Popescu	3	Sporting CP
4	Maxim	Marino	3	Juventus
5	Filip	Sokolov	3	Aalborg BK
6	Erik	Rossi	3	Borussia Dortmund
7	Sebastian	Ivanov	3	Feyenoord
8	Leo	Dupont	3	Atlético Madrid
9	Dominik	Smirnov	2	Arsenal
10	Dominik	Karlsen	2	FC Barcelona

- 2) Topsooduangdjad (sama eesmärk mis vaatel topvaravaloojad)

```
CREATE OR REPLACE VIEW v_topsooduangdjad AS SELECT
eesnimi, perenimi, count(soodu_andja) AS soote, t.nimi AS tiimi_nimi
FROM public."Mangijad" m
LEFT JOIN public."Varavad" v ON m.mangija_id=v.soodu_andja
LEFT JOIN public."Tiimid" t ON m.tiim=t.tiimid_id
GROUP BY eesnimi, perenimi, t.nimi
ORDER BY soote DESC;
```

Tulemus:

	A-Z eesnimi	A-Z perenimi	123 soote	A-Z tiimi_nimi
1	Maxim	Schmitz	3	Hearts
2	Luca	Marino	2	Ajax
3	Marko	Lindberg	2	Atlético Madrid
4	Andreas	Dubois	2	Arsenal
5	Nikolai	Bruno	2	Manchester City
6	Peter	Meier	2	Atlético Madrid
7	Tomas	Pires	2	Olympique Marseille
8	Leo	Kowalski	2	Celtic
9	Patrik	Huber	2	Celtic
10	Kristof	Meier	2	FC Porto

- 3) Tiimi statistika: Näitab tiimi kohta võitude, viikide ning kaotuste arvu. Eesmärk jällegi sama: kiire statistika tiimi kohta, analüüsimise põhi.

```
CREATE OR REPLACE VIEW v_tiimi_statistika AS
SELECT
  t.nimi AS tiimi_nimi,
  SUM(CASE
    WHEN (t.tiimid_id = m.koduvoistkond AND m.kodu_tulemus > m.kulaline_tulemus)
      OR (t.tiimid_id = m.kulalisvoistkond AND m.kulaline_tulemus > m.kodu_tulemus)
    THEN 1 ELSE 0
  END) AS voidud,
  SUM(CASE
    WHEN m.kodu_tulemus = m.kulaline_tulemus
    THEN 1 ELSE 0
  END) AS viigid,
  SUM(CASE
    WHEN (t.tiimid_id = m.koduvoistkond AND m.kodu_tulemus < m.kulaline_tulemus)
      OR (t.tiimid_id = m.kulalisvoistkond AND m.kulaline_tulemus < m.kodu_tulemus)
    THEN 1 ELSE 0
  END) AS kaotused
FROM public."Tiimid" t
JOIN public."Mangud" m
  ON t.tiimid_id = m.koduvoistkond OR t.tiimid_id = m.kulalisvoistkond
GROUP BY t.nimi;
```

Tulemus:

	A-z tiimi_nimi	123 voidud	123 viigid	123 kaotused
1	Rangers	3	1	0
2	SL Benfica	2	0	2
3	Feyenoord	2	0	2
4	RB Leipzig	2	0	2
5	Hearts	1	2	1
6	FC Porto	1	2	1
7	Bayern Munich	2	0	2
8	Olympique Marseille	2	2	0
9	Inter Milan	1	2	1
10	Real Madrid	1	1	2

- 4) Tiimivaravad: Näitab tiimi kohta palju on tiim väravaid löönud ning kui palju on tiimile väravaid löödud. Eesmärk jällegi laias pildis sama: saab teha läbi aastate iga tiimi kohta analüüsi ja otsustada näiteks tiimi juhtkonnal uue treeneri kasuks kui juba mitmendat aastat järjest on lastud väravate arv suurem kui löödud väravate oma.

```
CREATE OR REPLACE VIEW v_tiimivaravad AS
SELECT
  t.nimi AS tiimi_nimi,
  SUM(CASE
    WHEN m.koduvoistkond = t.tiimid_id THEN m.kodu_tulemus
    WHEN m.kulalisvoistkond = t.tiimid_id THEN m.kulaline_tulemus
    ELSE 0
  END) AS loodud_varavad,
  SUM(CASE
    WHEN m.koduvoistkond = t.tiimid_id THEN m.kulaline_tulemus
    WHEN m.kulalisvoistkond = t.tiimid_id THEN m.kodu_tulemus
    ELSE 0
  END) AS lastud_varavad
FROM public."Tiimid" t
JOIN public."Mangud" m
  ON t.tiimid_id = m.koduvoistkond OR t.tiimid_id = m.kulalisvoistkond
GROUP BY t.nimi;
```

Tulemus:

	A-Z tiimi_nimi	123 loodud_varavad	123 lastud_varavad
1	Rangers	12	9
2	SL Benfica	7	9
3	Feyenoord	9	7
4	RB Leipzig	12	12
5	Hearts	10	10
6	FC Porto	13	12
7	Bayern Munich	12	11
8	Olympique Marseille	10	8
9	Inter Milan	8	7
10	Real Madrid	10	11

## Protseduurid

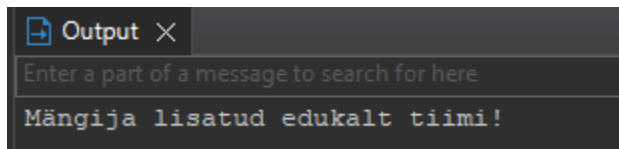
- 1) Mängija lisamine: Kuna meie ärireeglites on kehtestatud, et igal mängijal on unikaalne nimi ning igal mängijal peab olema kindlasti tiim, siis lõime mängija lisamise jaoks protseduuri, mis katkestab tegevuse, kui samanimeline mängija on juba tabelis või kui proovitakse lisada tiimi tühiku või NULLina.

```
CREATE OR REPLACE PROCEDURE lisa_mangija(  
    p_eesnimi TEXT,  
    p_perenimi TEXT,  
    p_tiid integer,  
    p_rahvus integer  
)  
LANGUAGE plpgsql  
AS $$  
BEGIN  
    IF EXISTS (  
        SELECT 1 FROM public."Mangijad"  
        WHERE eesnimi = p_eesnimi AND perenimi = p_perenimi  
    ) THEN  
        RAISE EXCEPTION 'Sellise nimega mängija on juba olemas!';  
    END IF;  
  
    IF p_tiid IS NULL THEN  
        RAISE EXCEPTION 'Igal mängijal peab olema tiim!';  
    END IF;  
  
    INSERT INTO public."Mangijad" (eesnimi, perenimi, tiim, rahvus)  
    VALUES (p_eesnimi, p_perenimi, p_tiid, p_rahvus);  
  
    RAISE NOTICE 'Mängija lisatud edukalt tiimi!';  
END;  
$$;
```

Kutsume välja protseduuri, kus proovime lisada uut mängijat, keda pole tabelis:

```
CALL lisa_mangija('Georg', 'Puu', 9, 1);
```

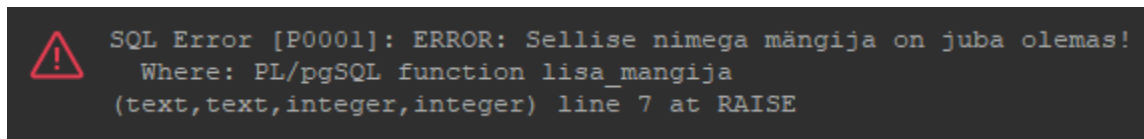
Tulemus:



642	Georg	Puu	9	1
-----	-------	-----	---	---

Kutsume välja protseduuri täpselt samade parameetritega. (Georg Puu on juba tabelis)

Tulemus:



Kutsume välja protseduuri, kus me tiimi jätame ära:

```
CALL lisa_mangija('Tambet', 'Puu', NULL, 1);
```

Tulemus:

```
SQL Error [P0001]: ERROR: Igal mängijal peab olema tiim!  
Where: PL/pgSQL function lisa_mangija  
(text,text,integer,integer) line 11 at RAISE  
  
Error position:
```

- 2) Tiimi lisamine: Ärireeglite tõttu lõime protseduuri, mis takistab lisada tiimi, millel ei ole kodustaadionit või lisada samanimelist tiimi. Nendel juhtudel kaketab protseduur töö, teatab kasutajat ja ei lase tiimi lisada.

```
CREATE OR REPLACE PROCEDURE lisa_tiiim(  
    p_nimi TEXT,  
    p_treener TEXT,  
    p_kapten INTEGER,  
    p_liiga INTEGER,  
    p_linn TEXT,  
    p_kodustaadion TEXT  
)  
LANGUAGE plpgsql  
AS $$  
BEGIN  
  
    IF p_kodustaadion IS NULL OR trim(p_kodustaadion) = '' THEN  
        RAISE EXCEPTION 'Ilma kodustaadionita tiimi ei saa lisada, tiimil peab olema kodustaadion!';  
    END IF;  
  
    IF EXISTS (  
        SELECT 1 FROM public."Tiimid"  
        WHERE nimi = p_nimi  
    ) THEN  
        RAISE EXCEPTION 'See tiim on juba lisatud!';  
    END IF;  
  
    INSERT INTO public."Tiimid" (nimi, treener, kapten, liiga, linn, kodustaadion)  
    VALUES (p_nimi, p_treener, p_kapten, p_liiga, p_linn, p_kodustaadion);  
  
    RAISE NOTICE 'Tiim lisatud edukalt!';  
END;  
$$;
```

Kutsume protseduuri, kus tahame lisada tiimi ilma kodustaadionita:

```
CALL lisa_tiiim('Viljandi Tulevik', 'Raimond Mets', 641, 5, 'Viljandi', NULL );
```

Tulemus:

```
SQL Error [P0001]: ERROR: Ilma kodustaadionita tiimi ei saa  
lisada, tiimil peab olema kodustaadion!  
Where: PL/pgSQL function lisa_tiiim  
(text,text,integer,integer,text,text) line 5 at RAISE
```

Kutsume protseduuri, kus tahame lisada tiimi, mis on juba olemas, enne lisame sama tiimi 'Viljandi Tulevik' kodustaadioniga 'Rannastaadion'.

Tulemused:

Uue tiimi lisamine olemasoleva kodustaadioniga:

**Tiim lisatud edukalt!** (output aken)

33	33	Viljandi Tulevik	Raimond Mets	641	5	Viljandi	Rannastaadion
----	----	------------------	--------------	-----	---	----------	---------------

(Tiimide tabel)

Proovime sama tiimi uuesti lisada:

```
SQL Error [P0001]: ERROR: See tiim on juba lisatud!  
Where: PL/pgSQL function lisa_tiim(text,text,integer,integer,text,text) line 13 at RAISE
```

## Andmebaasi loomine

Mahu tõttu on siit ära kustutatud mitusada risa andmete lisamist tabelisse, muidu oleks dokument lihtsalt sada lehekülge pikk tulnud.

```
--
-- PostgreSQL database dump
--

-- Dumped from database version 17.2
-- Dumped by pg_dump version 17.2

-- Started on 2025-05-02 20:06:03

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET transaction_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;

--
-- TOC entry 4 (class 2615 OID 2200)
-- Name: public; Type: SCHEMA; Schema: -; Owner: -
--

CREATE SCHEMA public;

--
-- TOC entry 4906 (class 0 OID 0)
-- Dependencies: 4
-- Name: SCHEMA public; Type: COMMENT; Schema: -; Owner: -
--
```



COMMENT ON SCHEMA public IS 'standard public schema';

```
--  
-- TOC entry 236 (class 1255 OID 17450)  
-- Name: f_lisa_varavad_uue_mangu_korral(); Type: FUNCTION; Schema: public; Owner: -  
--
```

```
CREATE FUNCTION public.f_lisa_varavad_uue_mangu_korral() RETURNS trigger  
    LANGUAGE plpgsql  
    AS $$  
DECLARE  
    i INTEGER;  
    aeg TEXT;  
    kasutatud_ajad TEXT[] := ARRAY[]::TEXT[];  
    skoor INTEGER;  
    varava_looja INTEGER;  
    soodu_andja INTEGER;  
    tiim_id INTEGER;  
BEGIN  
    -- Kodu v aravad  
    FOR i IN 1..NEW.kodu_tulemus LOOP  
        LOOP  
            aeg := LPAD(FLOOR(random() * 90)::TEXT, 2, '0') || ':' || LPAD(FLOOR(random() *  
60)::TEXT, 2, '0');  
            EXIT WHEN NOT (aeg = ANY(kasutatud_ajad));  
        END LOOP;  
  
        kasutatud_ajad := array_append(kasutatud_ajad, aeg);  
        varava_looja := (NEW.koduvoistkond - 1) * 20 + FLOOR(random() * 20)::INT + 1;  
        IF random() < 0.7 THEN  
            soodu_andja := (NEW.koduvoistkond - 1) * 20 + FLOOR(random() * 20)::INT + 1;  
        ELSE  
            soodu_andja := NULL;  
        END IF;  
  
        INSERT INTO "Varavad"(mang, varava_looja, soodu_andja, varava_aeg)  
        VALUES (NEW.mangu_id, varava_looja, soodu_andja, aeg);  
    END LOOP;
```

```

-- Külalise väravad
FOR i IN 1..NEW.kulaline_tulemus LOOP
  LOOP
    aeg := LPAD(FLOOR(random() * 90)::TEXT, 2, '0') || ':' || LPAD(FLOOR(random() *
60)::TEXT, 2, '0');
    EXIT WHEN NOT (aeg = ANY(kasutatud_ajad));
  END LOOP;
  kasutatud_ajad := array_append(kasutatud_ajad, aeg);
  varava_looja := (NEW.kulalisvoistkond - 1) * 20 + FLOOR(random() * 20)::INT + 1;
  IF random() < 0.7 THEN
    soodu_andja := (NEW.kulalisvoistkond - 1) * 20 + FLOOR(random() * 20)::INT + 1;
  ELSE
    soodu_andja := NULL;
  END IF;
  INSERT INTO "Varavad"(mang, varava_looja, soodu_andja, varava_aeg)
  VALUES (NEW.mangu_id, varava_looja, soodu_andja, aeg);
END LOOP;
RETURN NULL;
END;
$$;

```

```

--
-- TOC entry 224 (class 1255 OID 17444)
-- Name: f_uuenda_meistriteliiga(); Type: FUNCTION; Schema: public; Owner: -
--

```

```

CREATE FUNCTION public.f_uuenda_meistriteliiga() RETURNS trigger
  LANGUAGE plpgsql
  AS $$
BEGIN
  TRUNCATE TABLE "MeistriteLiiga";
  INSERT INTO "MeistriteLiiga" (tiim, punktid, koht_tabelis)
  SELECT
    tiim,
    punktid,
    ROW_NUMBER() OVER (ORDER BY punktid DESC, vv DESC) AS koht_tabelis
  FROM (
    SELECT
      tiim,

```

```

SUM(punktid) AS punktid,
SUM(varavaid_l) - SUM(varavaid_s) AS vv
FROM (
SELECT
    koduvoistkond AS tiim,
    CASE
        WHEN kodu_tulemus > kulaline_tulemus THEN 3
        WHEN kodu_tulemus = kulaline_tulemus THEN 1
        ELSE 0
    END AS punktid,
    kodu_tulemus AS varavaid_l,
    kulaline_tulemus AS varavaid_s
FROM "Mangud"
UNION ALL
SELECT
    kulalisvoistkond AS tiim,
    CASE
        WHEN kulaline_tulemus > kodu_tulemus THEN 3
        WHEN kulaline_tulemus = kodu_tulemus THEN 1
        ELSE 0
    END AS punktid,
    kulaline_tulemus AS varavaid_l,
    kodu_tulemus AS varavaid_s
FROM "Mangud"
) AS andmed
GROUP BY tiim
) AS koond;
RETURN NEW;
END;
$$;

--
-- TOC entry 237 (class 1255 OID 17452)
-- Name: kustuta_varavad_kui_mang_kustub(); Type: FUNCTION; Schema: public; Owner: -
--

CREATE FUNCTION public.kustuta_varavad_kui_mang_kustub() RETURNS trigger
    LANGUAGE plpgsql
    AS $$

```

```
BEGIN
DELETE FROM "Varavad" WHERE mang = OLD.mangu_id;
RETURN NULL;
END;
$$;
```

```
SET default_tablespace = '';
```

```
SET default_table_access_method = heap;
```

```
--
-- TOC entry 219 (class 1259 OID 17349)
-- Name: Liigad; Type: TABLE; Schema: public; Owner: -
--
```

```
CREATE TABLE public."Liigad" (
    liiga_id integer NOT NULL,
    nimi character varying(100),
    riik integer
);
```

```
--
-- TOC entry 221 (class 1259 OID 17359)
-- Name: Mangijad; Type: TABLE; Schema: public; Owner: -
--
```

```
CREATE TABLE public."Mangijad" (
    mangija_id integer NOT NULL,
    eesnimi character varying(100),
    perenimi character varying(100),
    tiim integer NOT NULL,
    rahvus integer NOT NULL
);
```

```
--
-- TOC entry 222 (class 1259 OID 17364)
-- Name: Mangud; Type: TABLE; Schema: public; Owner: -
```

--

```
CREATE TABLE public."Mangud" (  
    mangu_id integer NOT NULL,  
    koduvoistkond integer NOT NULL,  
    kulalisvoistkond integer NOT NULL,  
    kodu_tulemus integer NOT NULL,  
    kulaline_tulemus integer NOT NULL,  
    kohtunik character varying(100) NOT NULL,  
    mangu_aeg date NOT NULL  
);
```

--

```
-- TOC entry 217 (class 1259 OID 17339)  
-- Name: MeistriteLiiga; Type: TABLE; Schema: public; Owner: -  
--
```

```
CREATE TABLE public."MeistriteLiiga" (  
    tiim integer NOT NULL,  
    punktid integer,  
    koht_tabelis integer  
);
```

--

```
-- TOC entry 220 (class 1259 OID 17354)  
-- Name: Riigid; Type: TABLE; Schema: public; Owner: -  
--
```

```
CREATE TABLE public."Riigid" (  
    riigi_id integer NOT NULL,  
    nimi character varying(100) NOT NULL,  
    pealinn character varying(100) NOT NULL  
);
```

--

```
-- TOC entry 218 (class 1259 OID 17344)  
-- Name: Tiimid; Type: TABLE; Schema: public; Owner: -
```

--

```
CREATE TABLE public."Tiimid" (  
    tiimid_id integer NOT NULL,  
    nimi character varying(100) NOT NULL,  
    treener character varying(100),  
    kapten integer,  
    liiga integer NOT NULL,  
    linn character varying(100),  
    kodustaadion character varying(100)  
);
```

--

```
-- TOC entry 223 (class 1259 OID 17369)  
-- Name: Varavad; Type: TABLE; Schema: public; Owner: -  
--
```

```
CREATE TABLE public."Varavad" (  
    mang integer NOT NULL,  
    varava_looja integer NOT NULL,  
    soodu_andja integer,  
    varava_aeg character varying(5) NOT NULL  
);
```

--

```
-- TOC entry 4896 (class 0 OID 17349)  
-- Dependencies: 219  
-- Data for Name: Liigad; Type: TABLE DATA; Schema: public; Owner: -  
--
```

```
INSERT INTO public."Liigad" VALUES (1, 'Premier League', 31);  
INSERT INTO public."Liigad" VALUES (2, 'La Liga', 11);  
INSERT INTO public."Liigad" VALUES (3, 'Bundesliga', 9);  
INSERT INTO public."Liigad" VALUES (4, 'Serie A', 13);  
INSERT INTO public."Liigad" VALUES (5, 'Ligue 1', 10);  
INSERT INTO public."Liigad" VALUES (6, 'Primeira Liga', 12);  
INSERT INTO public."Liigad" VALUES (7, 'Eredivisie', 18);  
INSERT INTO public."Liigad" VALUES (8, 'Belgian Pro League', 17);
```

```
INSERT INTO public."Liigad" VALUES (9, 'Scottish Premiership', 32);
INSERT INTO public."Liigad" VALUES (10, 'Superliga', 5);
```

```
--
-- TOC entry 4898 (class 0 OID 17359)
-- Dependencies: 221
-- Data for Name: Mangijad; Type: TABLE DATA; Schema: public; Owner: -
--
```

```
INSERT INTO public."Mangijad" VALUES (1, 'Martin', 'Ivanov', 1, 22);
INSERT INTO public."Mangijad" VALUES (2, 'Matej', 'Novak', 1, 29);
INSERT INTO public."Mangijad" VALUES (3, 'Daniel', 'Pires', 1, 21);
```

... Siin on veel originaalis 637 rida mängijate lisamist, jätan selle siit ära.

```
--
-- TOC entry 4899 (class 0 OID 17364)
-- Dependencies: 222
-- Data for Name: Mangud; Type: TABLE DATA; Schema: public; Owner: -
--
```

```
INSERT INTO public."Mangud" VALUES (61, 22, 28, 2, 0, 'Kohtunik 50', '2024-10-01');
INSERT INTO public."Mangud" VALUES (62, 9, 30, 1, 3, 'Kohtunik 1', '2024-10-02');
INSERT INTO public."Mangud" VALUES (63, 9, 28, 4, 3, 'Kohtunik 38', '2024-10-03');
INSERT INTO public."Mangud" VALUES (64, 25, 26, 0, 1, 'Kohtunik 89', '2024-10-04');
... Siin on veel originaalis mitusada rida, jätan selle sit ära.
```

```
--
-- TOC entry 4894 (class 0 OID 17339)
-- Dependencies: 217
-- Data for Name: MeistriteLiiga; Type: TABLE DATA; Schema: public; Owner: -
--
```

```
INSERT INTO public."MeistriteLiiga" VALUES (2, 12, 1);
INSERT INTO public."MeistriteLiiga" VALUES (19, 10, 2);
INSERT INTO public."MeistriteLiiga" VALUES (32, 10, 3);
... Siin on veel originaalis 29 rida, jätan ära.
```

--  
-- TOC entry 4897 (class 0 OID 17354)  
-- Dependencies: 220  
-- Data for Name: Riigid; Type: TABLE DATA; Schema: public; Owner: -  
--

INSERT INTO public."Riigid" VALUES (1, 'Eesti', 'Tallinn');  
INSERT INTO public."Riigid" VALUES (2, 'Soome', 'Helsinki');  
INSERT INTO public."Riigid" VALUES (3, 'Rootsi', 'Stockholm');  
... Siin on samuti originaalis veel 29 INSERT INTO lauset, need jätan ära.

--  
-- TOC entry 4895 (class 0 OID 17344)  
-- Dependencies: 218  
-- Data for Name: Tiimid; Type: TABLE DATA; Schema: public; Owner: -  
--

INSERT INTO public."Tiimid" VALUES (1, 'Manchester City', 'Pep Guardiola', 15, 1, 'Manchester', 'Etihad Stadium');  
INSERT INTO public."Tiimid" VALUES (2, 'Arsenal', 'Mikel Arteta', 40, 1, 'London', 'Emirates Stadium');  
INSERT INTO public."Tiimid" VALUES (3, 'Liverpool', 'Jürgen Klopp', 60, 1, 'Liverpool', 'Anfield');  
...Siin veel tiimide lisamisi originaalis.

--  
-- TOC entry 4900 (class 0 OID 17369)  
-- Dependencies: 223  
-- Data for Name: Varavad; Type: TABLE DATA; Schema: public; Owner: -  
--

INSERT INTO public."Varavad" VALUES (1, 126, 130, '28:26');  
INSERT INTO public."Varavad" VALUES (1, 129, NULL, '76:52');  
INSERT INTO public."Varavad" VALUES (1, 138, 128, '72:44');  
...Siin samuti mitusada rida väravate lisamisi, jätan need ära.

--  
-- TOC entry 4726 (class 2606 OID 17353)



-- Name: Liigad Liigad\_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--

ALTER TABLE ONLY public."Liigad"  
ADD CONSTRAINT "Liigad\_pkey" PRIMARY KEY (liiga\_id);

--  
-- TOC entry 4730 (class 2606 OID 17363)  
-- Name: Mangijad Mangijad\_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--

ALTER TABLE ONLY public."Mangijad"  
ADD CONSTRAINT "Mangijad\_pkey" PRIMARY KEY (mangija\_id);

--  
-- TOC entry 4732 (class 2606 OID 17368)  
-- Name: Mangud Mangud\_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--

ALTER TABLE ONLY public."Mangud"  
ADD CONSTRAINT "Mangud\_pkey" PRIMARY KEY (mangu\_id);

--  
-- TOC entry 4722 (class 2606 OID 17343)  
-- Name: MeistriteLiiga MeistriteLiiga\_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--

ALTER TABLE ONLY public."MeistriteLiiga"  
ADD CONSTRAINT "MeistriteLiiga\_pkey" PRIMARY KEY (tiim);

--  
-- TOC entry 4728 (class 2606 OID 17358)  
-- Name: Riigid Riigid\_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--

ALTER TABLE ONLY public."Riigid"

```
ADD CONSTRAINT "Riigid_pkey" PRIMARY KEY (riigi_id);
```

```
--  
-- TOC entry 4724 (class 2606 OID 17348)  
-- Name: Tiimid Tiimid_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public."Tiimid"  
ADD CONSTRAINT "Tiimid_pkey" PRIMARY KEY (tiimid_id);
```

```
--  
-- TOC entry 4734 (class 2606 OID 17430)  
-- Name: Varavad Varavad_pkey; Type: CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public."Varavad"  
ADD CONSTRAINT "Varavad_pkey" PRIMARY KEY (varava_looja, varava_aeg);
```

```
--  
-- TOC entry 4746 (class 2620 OID 17451)  
-- Name: Mangud tg_lisa_varavad; Type: TRIGGER; Schema: public; Owner: -  
--
```

```
CREATE TRIGGER tg_lisa_varavad AFTER INSERT ON public."Mangud" FOR EACH ROW  
EXECUTE FUNCTION public.f_lisa_varavad_uue_mangu_korral();
```

```
--  
-- TOC entry 4747 (class 2620 OID 17445)  
-- Name: Mangud tg_uuenda_meistriteliiga; Type: TRIGGER; Schema: public; Owner: -  
--
```

```
CREATE TRIGGER tg_uuenda_meistriteliiga AFTER INSERT OR DELETE OR UPDATE ON  
public."Mangud" FOR EACH STATEMENT EXECUTE FUNCTION  
public.f_uuenda_meistriteliiga();
```

```
--  
-- TOC entry 4748 (class 2620 OID 17480)  
-- Name: Mangud tr_kustuta_varavad; Type: TRIGGER; Schema: public; Owner: -  
--
```

```
CREATE TRIGGER tr_kustuta_varavad AFTER DELETE ON public."Mangud" FOR EACH  
ROW EXECUTE FUNCTION public.kustuta_varavad_kui_mang_kustub();
```

```
--  
-- TOC entry 4738 (class 2606 OID 17389)  
-- Name: Liigad Liigad_riik_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public."Liigad"  
  ADD CONSTRAINT "Liigad_riik_fkey" FOREIGN KEY (riik) REFERENCES  
public."Riigid"(riigi_id);
```

```
--  
-- TOC entry 4739 (class 2606 OID 17399)  
-- Name: Mangijad Mangijad_rahvus_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public."Mangijad"  
  ADD CONSTRAINT "Mangijad_rahvus_fkey" FOREIGN KEY (rahvus) REFERENCES  
public."Riigid"(riigi_id);
```

```
--  
-- TOC entry 4740 (class 2606 OID 17394)  
-- Name: Mangijad Mangijad_tiim_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--
```

```
ALTER TABLE ONLY public."Mangijad"  
  ADD CONSTRAINT "Mangijad_tiim_fkey" FOREIGN KEY (tiim) REFERENCES  
public."Tiimid"(tiimid_id);
```

```
--
```

-- TOC entry 4741 (class 2606 OID 17404)  
-- Name: Mangud Mangud\_koduvoistkond\_fkey; Type: FK CONSTRAINT; Schema: public;  
Owner: -  
--

```
ALTER TABLE ONLY public."Mangud"  
  ADD CONSTRAINT "Mangud_koduvoistkond_fkey" FOREIGN KEY (koduvoistkond)  
  REFERENCES public."Tiimid"(tiimid_id);
```

--  
-- TOC entry 4742 (class 2606 OID 17409)  
-- Name: Mangud Mangud\_kulalisvoistkond\_fkey; Type: FK CONSTRAINT; Schema: public;  
Owner: -  
--

```
ALTER TABLE ONLY public."Mangud"  
  ADD CONSTRAINT "Mangud_kulalisvoistkond_fkey" FOREIGN KEY (kulalisvoistkond)  
  REFERENCES public."Tiimid"(tiimid_id);
```

--  
-- TOC entry 4735 (class 2606 OID 17374)  
-- Name: MeistriteLiiga MeistriteLiiga\_tiim\_fkey; Type: FK CONSTRAINT; Schema: public;  
Owner: -  
--

```
ALTER TABLE ONLY public."MeistriteLiiga"  
  ADD CONSTRAINT "MeistriteLiiga_tiim_fkey" FOREIGN KEY (tiim) REFERENCES  
  public."Tiimid"(tiimid_id);
```

--  
-- TOC entry 4736 (class 2606 OID 17379)  
-- Name: Tiimid Tiimid\_kapten\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--

```
ALTER TABLE ONLY public."Tiimid"  
  ADD CONSTRAINT "Tiimid_kapten_fkey" FOREIGN KEY (kapten) REFERENCES  
  public."Mangijad"(mangija_id);
```

--  
-- TOC entry 4737 (class 2606 OID 17384)  
-- Name: Tiimid Tiimid\_liiga\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--

```
ALTER TABLE ONLY public."Tiimid"  
    ADD CONSTRAINT "Tiimid_liiga_fkey" FOREIGN KEY (liiga) REFERENCES  
public."Liigad"(liiga_id);
```

--  
-- TOC entry 4743 (class 2606 OID 17481)  
-- Name: Varavad Varavad\_mang\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: -  
--

```
ALTER TABLE ONLY public."Varavad"  
    ADD CONSTRAINT "Varavad_mang_fkey" FOREIGN KEY (mang) REFERENCES  
public."Mangud"(mangu_id) ON DELETE CASCADE;
```

--  
-- TOC entry 4744 (class 2606 OID 17424)  
-- Name: Varavad Varavad\_soodu\_andja\_fkey; Type: FK CONSTRAINT; Schema: public;  
Owner: -  
--

```
ALTER TABLE ONLY public."Varavad"  
    ADD CONSTRAINT "Varavad_soodu_andja_fkey" FOREIGN KEY (soodu_andja)  
REFERENCES public."Mangijad"(mangija_id);
```

--  
-- TOC entry 4745 (class 2606 OID 17419)  
-- Name: Varavad Varavad\_varava\_looja\_fkey; Type: FK CONSTRAINT; Schema: public;  
Owner: -  
--

```
ALTER TABLE ONLY public."Varavad"
```

```
ADD CONSTRAINT "Varavad_varava_looja_fkey" FOREIGN KEY (varava_looja)
REFERENCES public."Mangijad"(mangija_id);
```

```
-- Completed on 2025-05-02 20:06:03
```

```
--
```

```
-- PostgreSQL database dump complete
```

```
--
```