Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерной техники

**Дисциплина:** Базы данных

Отчёт по лабораторной работе № 4

**Выполнил:** До Зыонг Мань

Ву Минь Хиеу

**Студент группы**: P33201

**Преподаватель:** Машина Екатерина Алексеевна

Санкт-Петербург

2021 г.

1. **Текст задания**

**Для выполнения лабораторной работы №4 необходимо:**

* Реализовать разработанную в рамках лабораторной работы №3 даталогическую модель в реляционной СУБД PostgreSQL.
* Заполнить созданные таблицы данными.
* Обеспечить целостность данных при помощи средств языка DDL.
* В рамках лабораторной работы должны быть разработаны скрипты для создания/удаления требуемых объектов базы данных, заполнения/удаления содержимого созданных таблиц.

**Отчёт по лабораторной работе должен содержать:**

* титульный лист;
* текст задания;
* описание предметной области;
* DDL-скрипты, часть DML-скриптов;
* выводы по работе.

1. **РеализациядаталогическоймоделинаSQL**

**Типы перечислений**

1. **Position**

CREATE TYPE **POSITION** AS ENUM (‘GK’, 'CB', 'LB', ‘RB’, ‘CDM’, ‘CAM’, ‘CM’, ‘CF’, ‘RF’, ‘LF’, ‘LM’, ‘RM’);

1. **Level\_league**

CREATE TYPE **LEVEL** AS ENUM (‘CLUB’, 'INTERNATIONAL');

**Стержневые сущности**

1. **Attribute**

CREATE TABLE IF NOT EXISTS **attribute** (

id SERIAL PRIMARY KEY,

attack INTEGER DEFAULT 50,

defense INTEGER DEFAULT 50,

dribble INTEGER DEFAULT 50,

pass INTEGER DEFAULT 50,

speed INTEGER DEFAULT 50,

power INTEGER DEFAULT 50,

stamina INTEGER DEFAULT 50,

);

1. **Player**

CREATE TABLE IF NOT EXISTS **player** (

id SERIAL PRIMARY KEY,

full\_name VARCHAR(50) NOT NULL,

height INTEGER NOT NULL,

weight DOUBLE NOT NULL,

position POSITION NOT NULL,

nationality VARCHAR(50) NOT NULL,

shirt\_number INTEGER NOT NULL,

birthday DATE NOT NULL,

image VARCHAR(100),

id\_attribute INTEGER REFERENCES attribute NOT NULL

);

1. **Sponsor**

CREATE TABLE IF NOT EXISTS **sponsor** (

id SERIAL PRIMARY KEY,

name VARCHAR(50) NOT NULL,

headquarter VARCHAR(50) NOT NULL,

management VARCHAR(50) NOT NULL,

assets BIGINT NOT NULL

);

1. **Goal**

CREATE TABLE IF NOT EXISTS **goal** (

id SERIAL PRIMARY KEY,

time TIMESTAMP NOT NULL,

video VARCHAR(100) NOT NULL,

rating INTEGER

);

1. **League**

CREATE TABLE IF NOT EXISTS **league** (

id SERIAL PRIMARY KEY,

name VARCHAR(50) NOT NULL,

region VARCHAR(50) NOT NULL,

level LEVEL NOT NULL

);

1. **Stadium**

CREATE TABLE IF NOT EXISTS **stadium** (

id SERIAL PRIMARY KEY,

name VARCHAR(50) NOT NULL,

coordinate VARCHAR(50) NOT NULL,

capacity BIGINT NOT NULL,

owner VARCHAR(50) NOT NULL

);

1. **Coach**

CREATE TABLE IF NOT EXISTS **coach** (

id SERIAL PRIMARY KEY,

full\_name VARCHAR(50) NOT NULL,

birthday DATE NOT NULL,

nationality VARCHAR(50) NOT NULL

);

1. **Referee**

CREATE TABLE IF NOT EXISTS **referee** (

id SERIAL PRIMARY KEY,

full\_name VARCHAR(50) NOT NULL,

birthday DATE NOT NULL,

nationality VARCHAR(50) NOT NULL,

rating FLOAT4 NOT NULL

);

1. **Club**

CREATE TABLE IF NOT EXISTS **club** (

id SERIAL PRIMARY KEY,

full\_name VARCHAR(50) NOT NULL,

nationality VARCHAR(50) NOT NULL,

owner VARCHAR(50) NOT NULL,

logo VARCHAR(100) NOT NULL,

id\_coach INTEGER REFERENCES couch NOT NULL,

id\_stadium INTEGER REFERENCES stadium NOT NULL

);

1. **Match**

CREATE TABLE IF NOT EXISTS **match** (

id SERIAL PRIMARY KEY,

special\_name VARCHAR(50) NOT NULL UNIQUE,

time TIMESTAMP NOT NULL,

weather VARCHAR(50) NOT NULL,

audience INTEGER NOT NULL

);

**Ассоциативные сущности**

1. **Player\_Club**

CREATE TABLE IF NOT EXISTS **player\_club** (

id\_player INTEGER REFERENCES player NOT NULL,

id\_team INTEGER REFERENCES club NOT NULL

);

1. **Club\_League**

CREATE TABLE IF NOT EXISTS **club\_league** (

club\_id INTEGER REFERENCES club NOT NULL,

league\_id INTEGER REFERENCES league NOT NULL

);

1. **Club\_Match**

CREATE TABLE IF NOT EXISTS **club\_match** (

club\_id INTEGER REFERENCES club NOT NULL,

match\_id INTEGER REFERENCES match NOT NULL

);

1. **Match\_League**

CREATE TABLE IF NOT EXISTS **match\_league**(

match\_id INTEGER REFERENCES match NOT NULL,

league\_id INTEGER REFERENCES league NOT NULL

);

1. **Match\_Goal**

CREATE TABLE IF NOT EXISTS **match\_goal**(

match\_id INTEGER REFERENCES match NOT NULL,

goal\_id INTEGER REFERENCES goal NOT NULL,

);

1. **Match\_Referee**

CREATE TABLE IF NOT EXISTS **match\_referee** (

match\_id INTEGER REFERENCES match NOT NULL,

referee\_id INTEGER REFERENCES referee NOT NULL

);

1. **Match\_Stadium**

CREATE TABLE IF NOT EXISTS **match\_stadium** (

match\_id INTEGER REFERENCES match NOT NULL,

stadium\_id INTEGER REFERENCES stadium NOT NULL

);

1. **Player\_Goal**

CREATE TABLE IF NOT EXISTS **player\_goal** (

player\_id INTEGER REFERENCES player NOT NULL,

goal\_id INTEGER REFERENCES goal NOT NULL

);

1. **Player\_MOTM**

CREATE TABLE IF NOT EXISTS **play\_MOTM** (

player\_id INTEGER REFERENCES player NOT NULL,

match\_id INTEGER REFERENCES match NOT NULL

);

1. **Sponsor\_Club**

CREATE TABLE IF NOT EXISTS **sponsor\_club** (

sponsor\_id INTEGER REFERENCES sponsor NOT NULL,

club\_id INTEGER REFERENCES club NOT NULL

);

1. **Заполнение данными**

Я использовал генератор случайных запросов по следующей ссылке

1. **Вывод**

Мы научились вычленять данные из моего предметной области, составлять инфологическую и даталогическую таблицы.