

Санкт-Петербургский государственный политехнический университет
Кафедра компьютерных систем и программных технологий

Отчёт по лабораторной работе

Дисциплина: Базы данных

Тема: Язык SQL-DDL

Выполнил студент гр. 43501/3

(подпись) В.Е. Бушин

Руководитель

(подпись) А.В. Мяснов

“ _ ” _____ 2015 г.

Санкт-Петербург

2015

1. Цель работы

Познакомить студентов с основами проектирования схемы БД, языком описания сущностей и ограничений БД SQL-DDL.

2. Язык SQL

Язык SQL (Structured Query Language) - язык структурированных запросов. Он позволяет формировать весьма сложные запросы к базам данных. В SQL определены два подмножества языка:

- SQL-DDL (Data Definition Language) - язык определения структур и ограничений целостности баз данных. Сюда относятся команды создания и удаления баз данных; создания, изменения и удаления таблиц; управления пользователями и т.д.
- SQL-DML (Data Manipulation Language) - язык манипулирования данными: добавление, изменение, удаление и извлечение данных, управления транзакциями.

3. Программа работы

- 1) Самостоятельное изучение SQL-DDL
- 2) Создание скрипта БД в соответствии с согласованной схемой (должны присутствовать первичные и внешние ключи, ограничения на диапазоны значений). Продемонстрировать скрипт преподавателю.
- 3) Создайте скрипт, заполняющий все таблицы БД данными
- 4) Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя. Продемонстрировать их работу преподавателю.
- 5) Изучите основные возможности IBExpert. Получите ER-диаграмму созданной БД с помощью Database Designer.
- 6) Автоматически сгенерируйте данные при помощи IBExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц)

4. Ход работы

- 1) Был самостоятельно изучен язык SQL-DDL.
- 2) Создан скрипт, создающий БД в соответствии с согласованной схемой:

```
connect 'D:\database\FOOTBALL.fdb' user 'SYSDBA' password 'masterkey';
drop database;
create database 'D:\database\FOOTBALL.fdb' user 'SYSDBA' password
'masterkey';
create table Club(
    id int primary key,
    club_name varchar(25),
    city varchar(25)
);

create table Player(
    id int primary key,
    player_name varchar(25),
    player_surname varchar(25),
    number integer NOT null,
    clubID int not null,
    country varchar(25)
```

```

);

alter table Player add constraint player_to_club
foreign key (clubID) references Club(id);

create domain StageDomain
AS integer check (value in (1,2,4,8,16,32));

create table Match(
id int primary key,
matchDate DATE,
stage StageDomain,
home int not null,
away int not null,
result varchar(25),
winner varchar(25)
);

alter table Match add constraint match_to_home
foreign key (home) references Club(id);
alter table Match add constraint match_to_away
foreign key (away) references Club(id);

create domain ActTypeDomain
AS varchar(15)
check (value in ("goal", "assist", "yellow card", "red card"));

create table Act (
id int primary key,
playerID int not null,
actType ActTypeDomain,
match_minute int
);

alter table Act add constraint acts_to_player
foreign key (playerID) references Player(id);

create table MatchStat (
id int primary key,
matchID int not null,
home_scored integer,
away_scored integer,
actsID int not null
);

alter table MatchStat add constraint matchStat_to_match
foreign key (matchID) references Match(id);
alter table MatchStat add constraint match_to_act
foreign key (actsID) references Act(id);

commit;

```

3) Создан скрипт, заполняющий БД данными:

```
connect 'D:\database\FOOTBALL.fdb' user 'SYSDBA' password 'masterkey';

insert into Club (id, club_name, city) values (1, "Arsenal", "London");
insert into Club (id, club_name, city) values (2, "Brighton & Hove Albion FC",
"Brighton");
insert into Club (id, club_name, city) values (3, "Reading", "Reading");
insert into Club (id, club_name, city) values (4, "Cardiff City", "Cardiff");

insert into Player values (1, "Theo", "Walcott", 14, 1, "England");
insert into Player values (2, "Samuel", "Baldock", 9, 2, "England");
insert into Player values (3, "Pavel", "Pogrebnyak", 7, 3, "Russia");
insert into Player values (4, "Kenwyne", "Jones", 9, 4, "Trinidad and Tobago");

insert into Match values (1, '2015-01-25', 16, 2, 1, "2:3", "Arsenal");
insert into Match values (2, '2015-01-24', 16, 4, 3, "1:2", "Reading");

insert into Act values (1, 1, "goal", 2);
insert into Act values (2, 2, "goal", 75);
insert into Act values (4, 4, "goal", 25);

insert into MatchStat values (1, 1, 2, 3, 1);
insert into MatchStat values (2, 2, 1, 2, 4);

commit;
```

4) Создан скрипт, модифицирующий БД в соответствии с заданием преподавателя:

- Добавить учет трансферов игроков;
- Добавить возможность учета чемпионатов с хранением турнирной таблицы.

```
connect 'D:\database\FOOTBALL.fdb' user 'SYSDBA' password 'masterkey';

alter table Player add transferPrice int not null;

create table Transfer(
    id int primary key,
    playerID int not null,
    price int,
    fromClub int not null,
    toClub int not null
);

alter table Transfer add constraint transfer_to_player
foreign key (playerID) references Player(id);

create table Championship(
    id int primary key,
    clubID int not null,
    played int,
    won int,
    drawn int,
    lost int,
```

```

points int,
goalsFor int,
goalsAgainst int
);

```

```

alter table Championship add constraint champ_to_club
foreign key (clubID) references Club(id);

```

```

commit;

```

- 5) С помощью инструмента DatabaseDesigner в IBExpert получена ER-диаграмма, созданной БД (рис. 1).

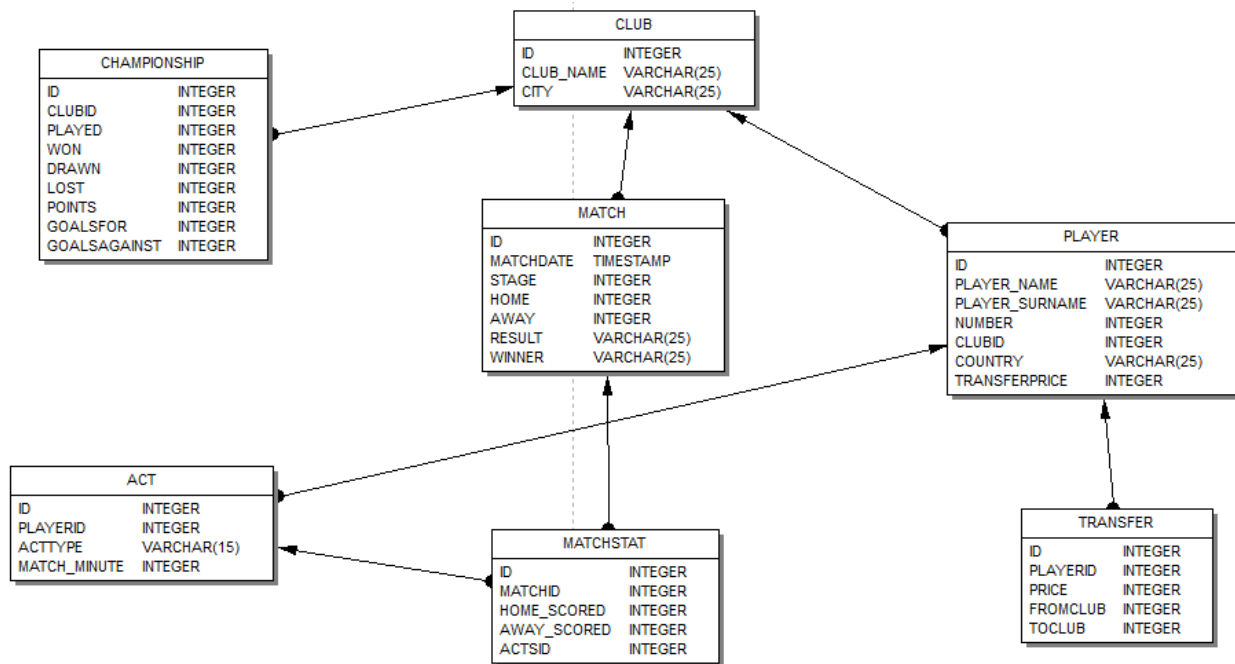


Рис.1 ER-диаграмма БД

- б) С помощью инструмента Test Data Generator в IBExpert сгенерированы тестовые данные (по 10000 записей) для 3 таблиц БД: таблица игроков Player, таблица действий Act и таблица матчей Match.

5. Вывод

В результате работы был изучен язык SQL-DDL. Были написаны скрипты для создания, заполнения и модификации БД.

SQL-DDL (Data Definition Language) - язык определения структур и ограничений целостности баз данных. Использовались такие команды языка DDL как create (создать), alter (изменить), drop (удалить).

Основой языка SQL является язык реляционных баз данных. Язык SQL позволяет управлять структурой БД, это обеспечивает гибкость БД к изменяющимся требованиям. Изучены некоторые возможности среды IBExpert: генерирование ER-диаграммы созданной БД и заполнение таблиц тестовыми данными.