Московский Авиационный Институт

(Национальный исследовательский университет)

**Лабораторная работа**

**По курсу:**

**«Базы Данных»**

Студент: Сорокин Д.М.

Группа: 8О-304Б

Оценка:

Москва, 2017

**Постановка задачи:** разработать и реализовать Базу Данных по индивидуальному заданию в инструментальной среде разработчика Microsoft VisualFoxPro

**Вариант:** база данных игроков, игр и статистики хоккейной команды ХК «Авиаторы» МАИ.

# Таблица и схема данных

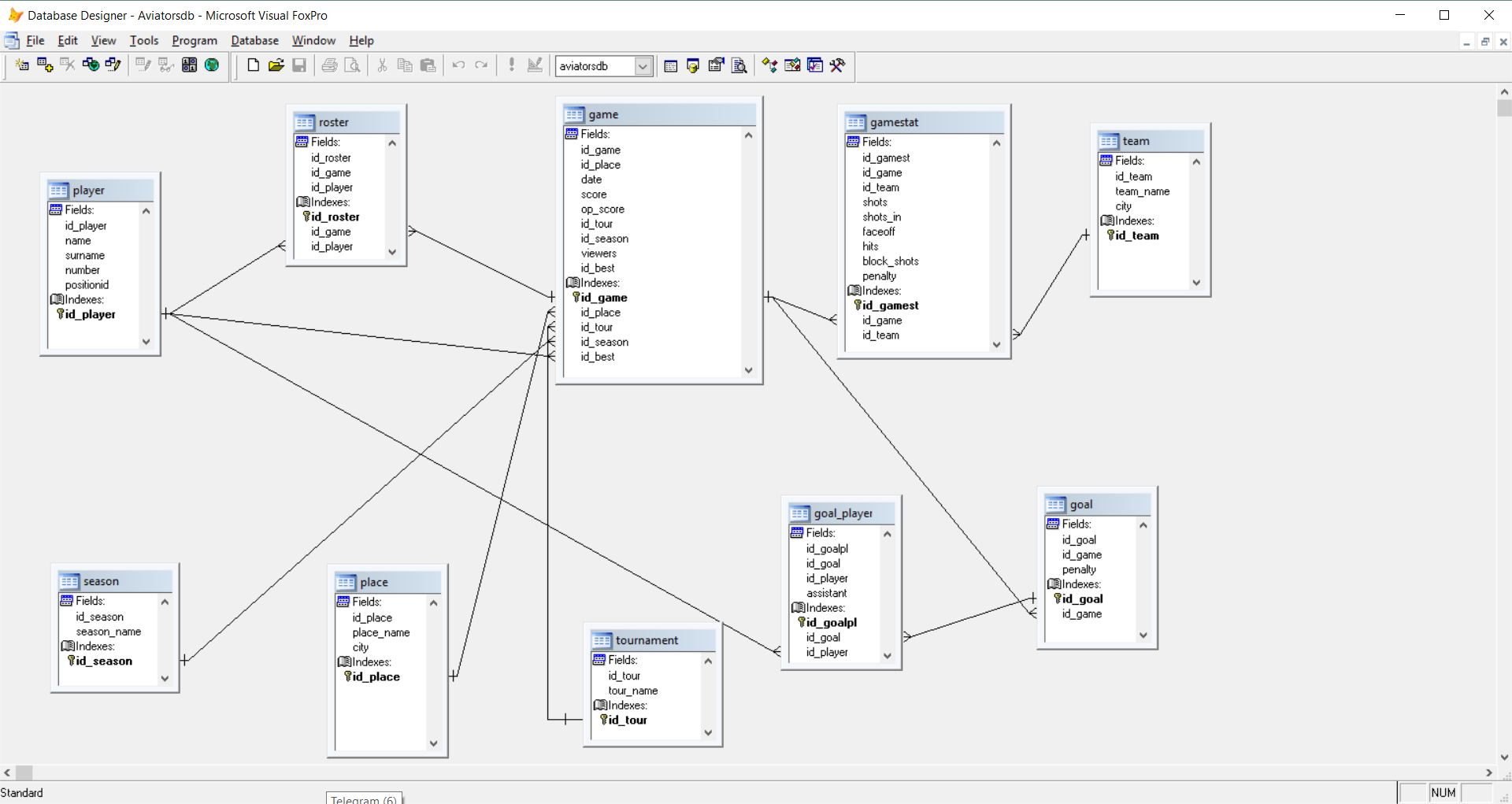
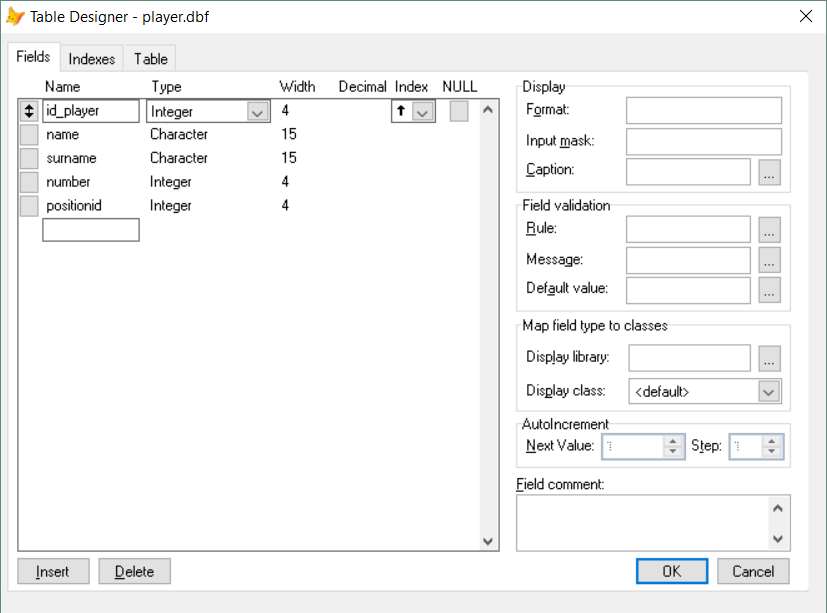
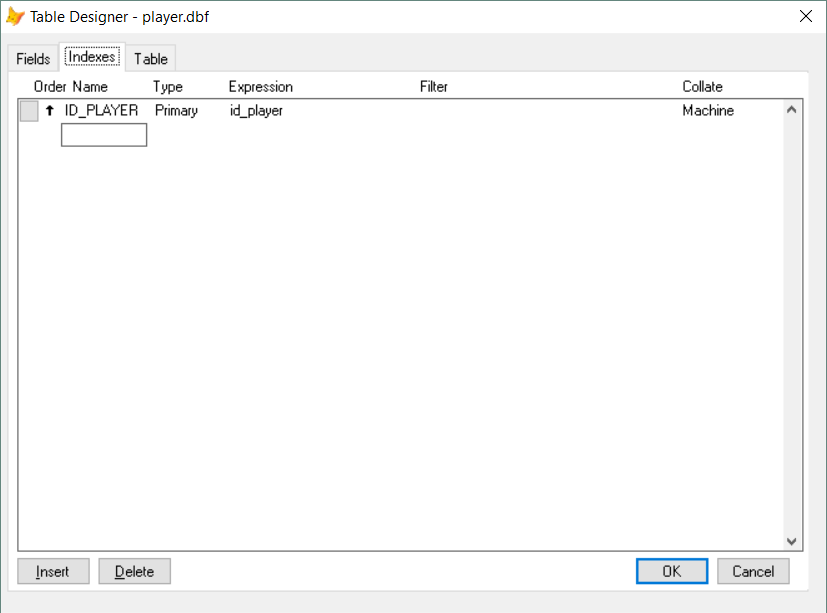


Таблица Player

Таблица содержит имена, фамилии, номера и позиции всех игроков когда-либо игравших за команду.





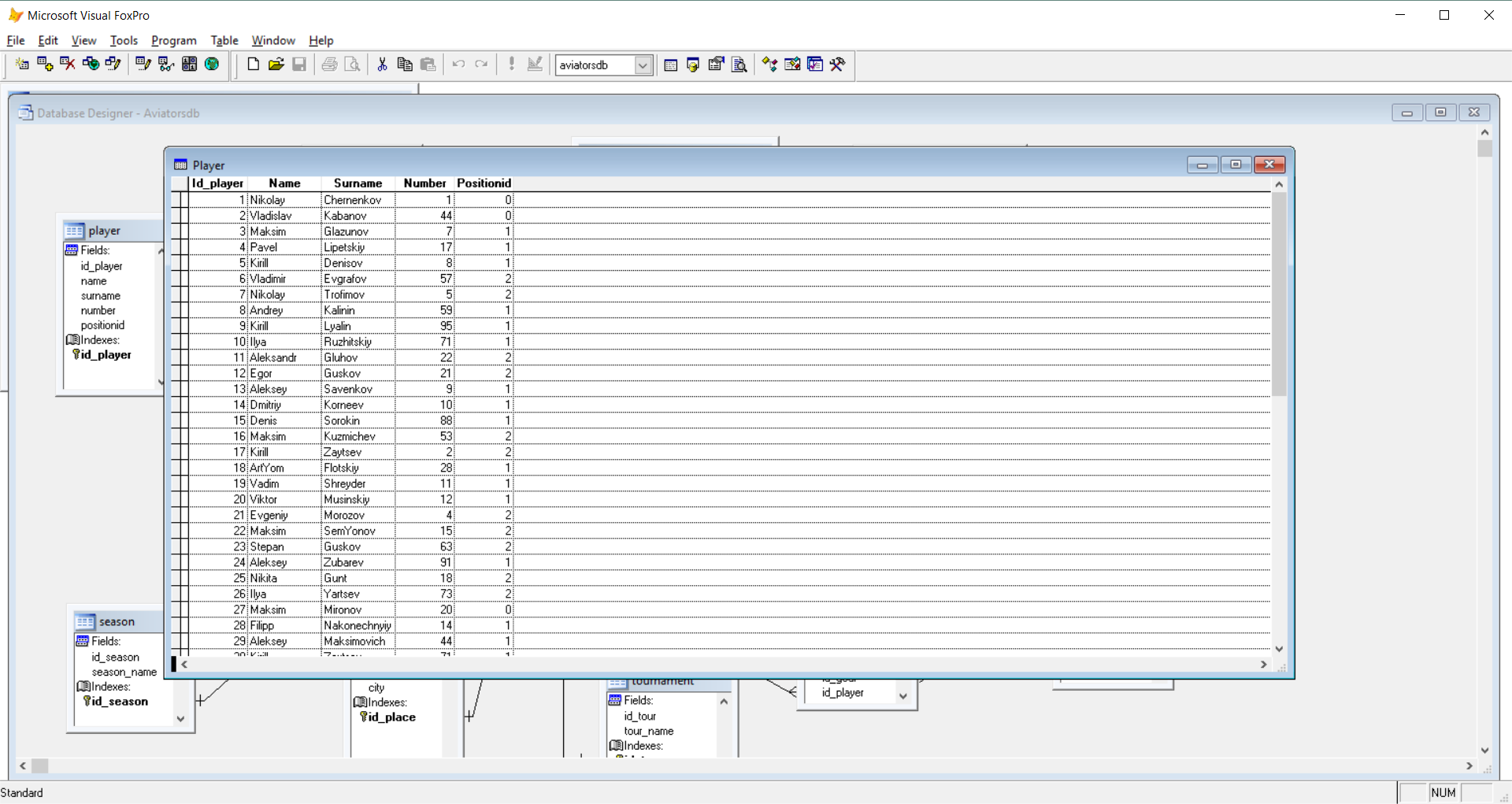
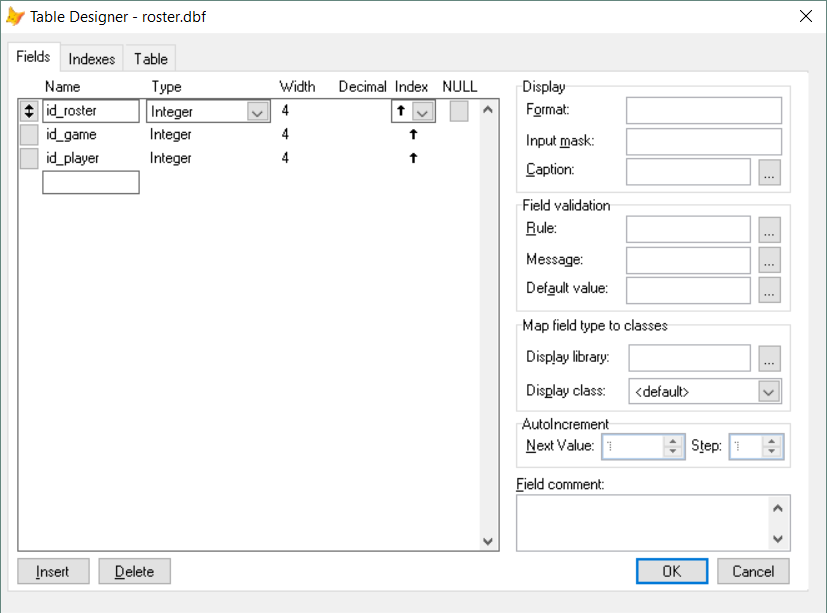
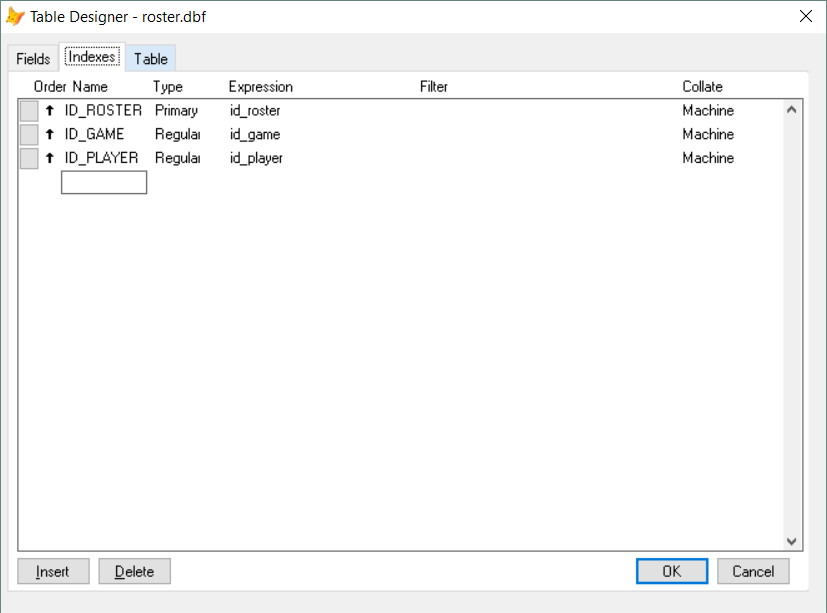


Таблица Roster

Таблица содержит информацию по составу на конкретную игру (игрок, игра)





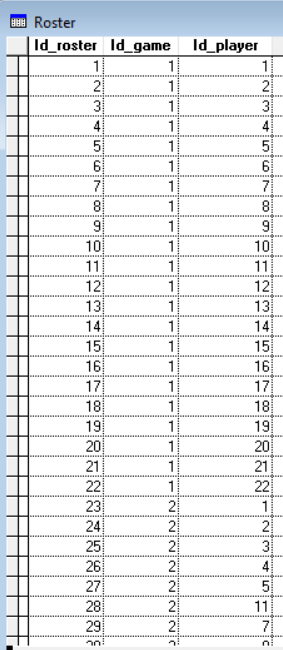
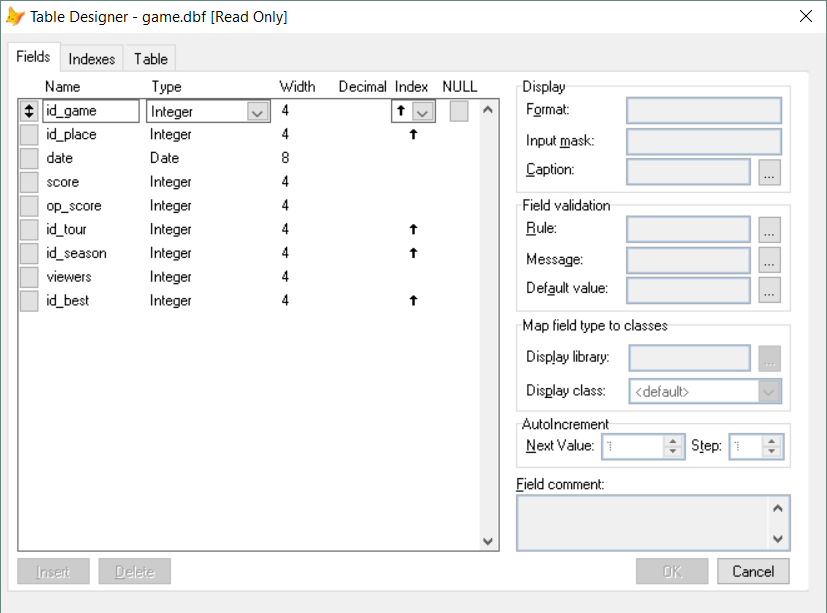
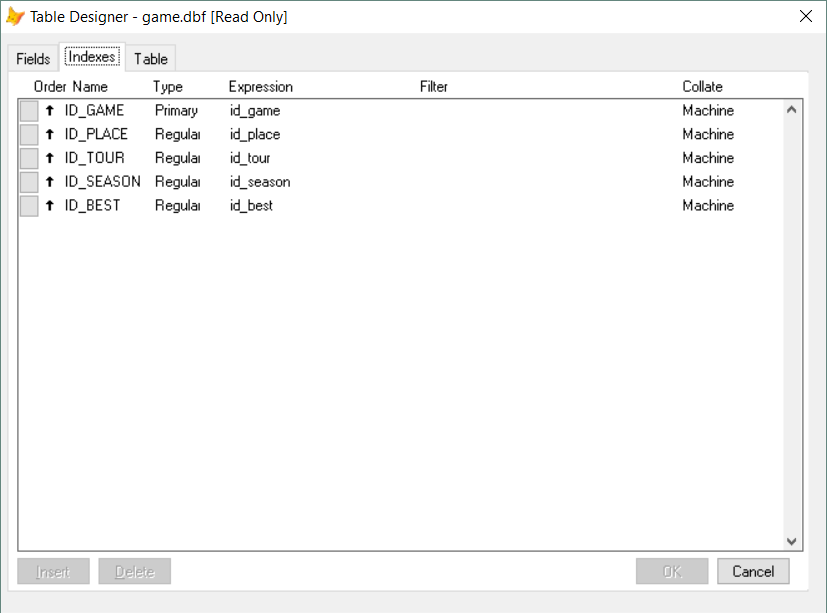


Таблица Game

Таблица содержит информацию об игре (место проведения, дата, счет «Авиаторов», счет соперника, турнир, сезон, количество зрителей и лучшего игрока).





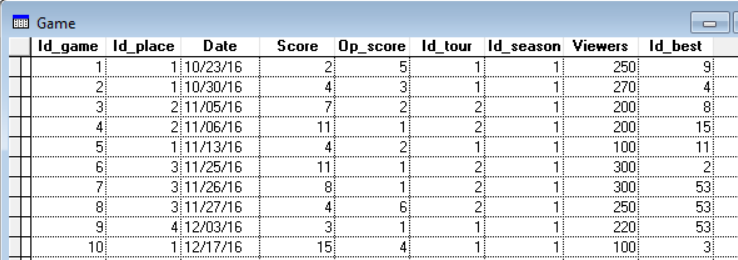
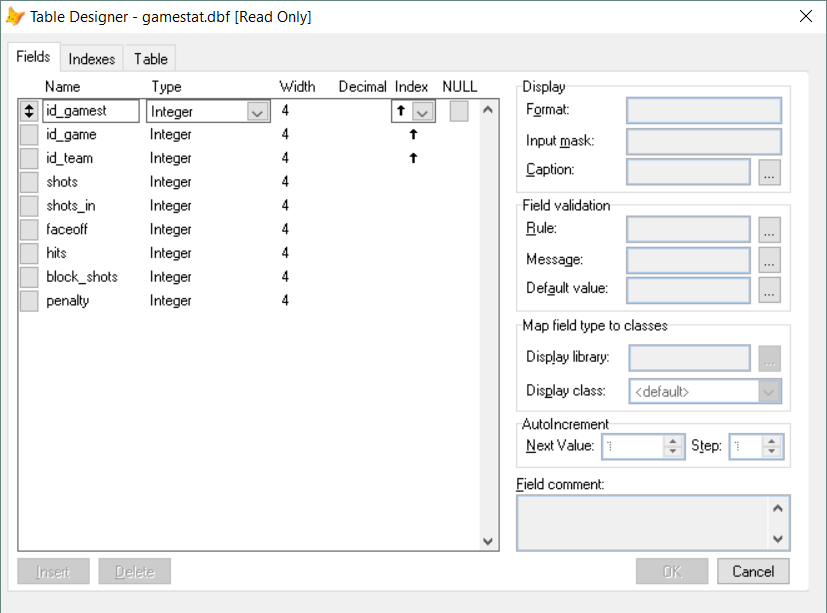
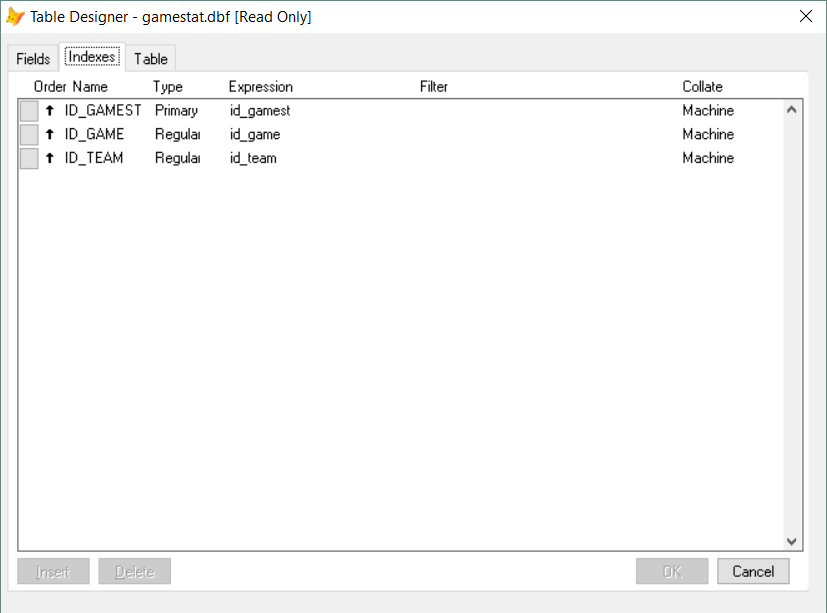
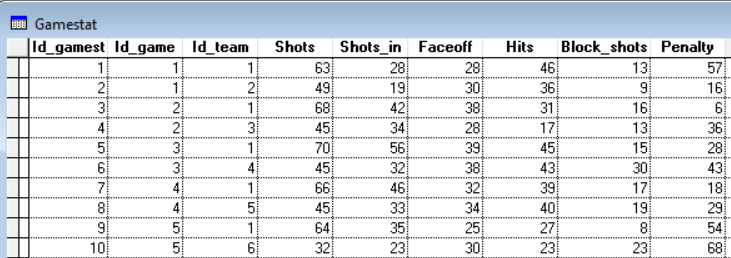


Таблица GameStat

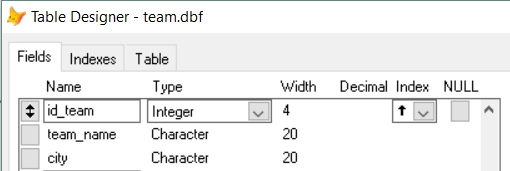
Таблица содержит информацию о статистистике игры (команда к которой относится статистика, броски, броски в створ, выигранные вбрасывания, силовые приёмы, заблокированные броски, штрафы).

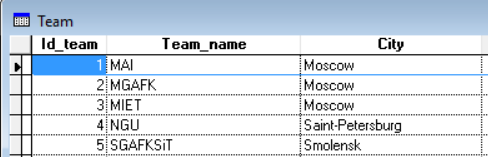
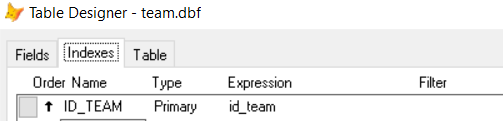


Таблицы Team, Season, Tournament, Place

Таблицы имеют схожую стуктуру, поэтому я приведу структуру одной из них. Таблицы содержат название и, может быть, название города.





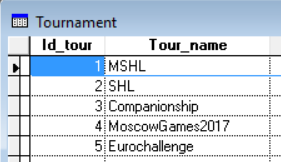






Таблица Goal

Таблица содержит информацию о голе, в какой игре он забит и является ли буллитом (штрафным броском).

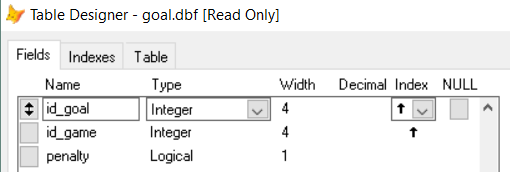
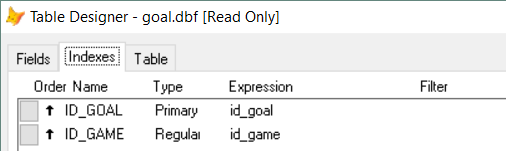
  

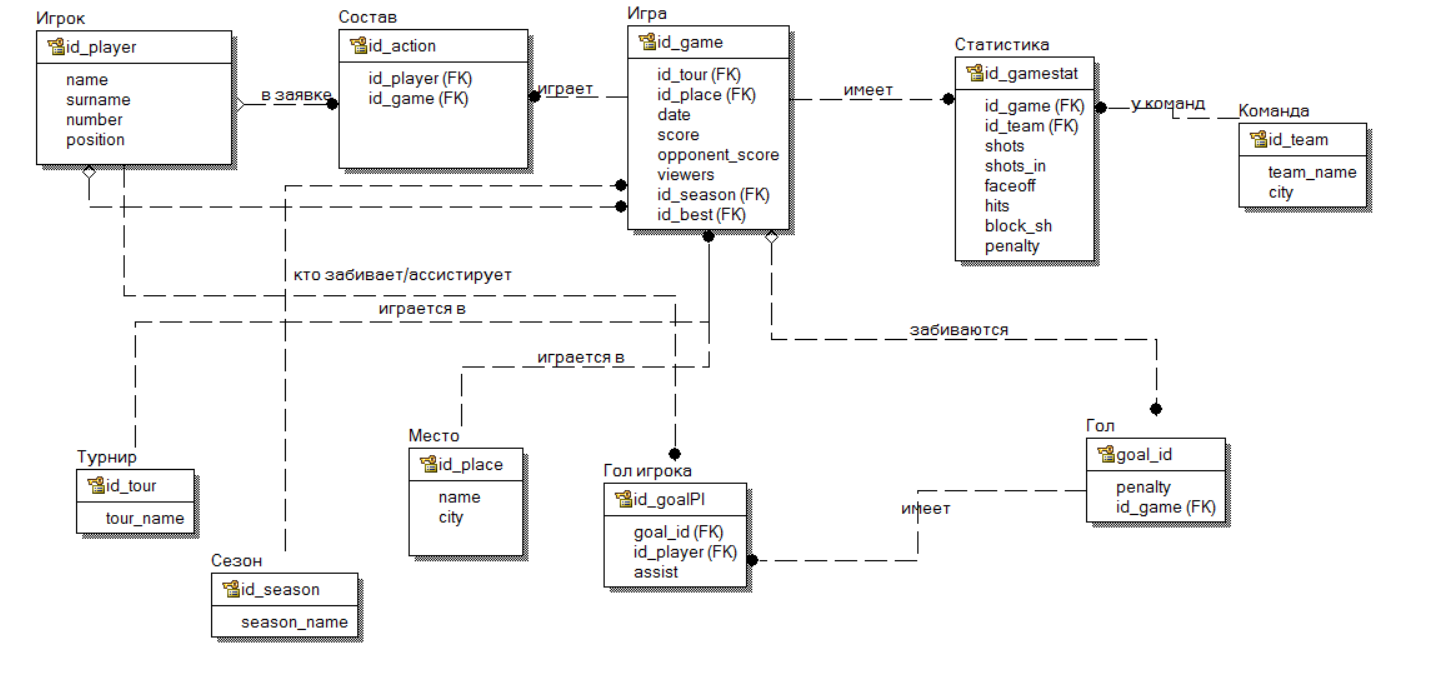
Таблица GoalPlayer

Таблица содержит информацию об игроках, участвовавших в голах (какой это гол, игрок, является ли игрок ассистентом в этом голе).

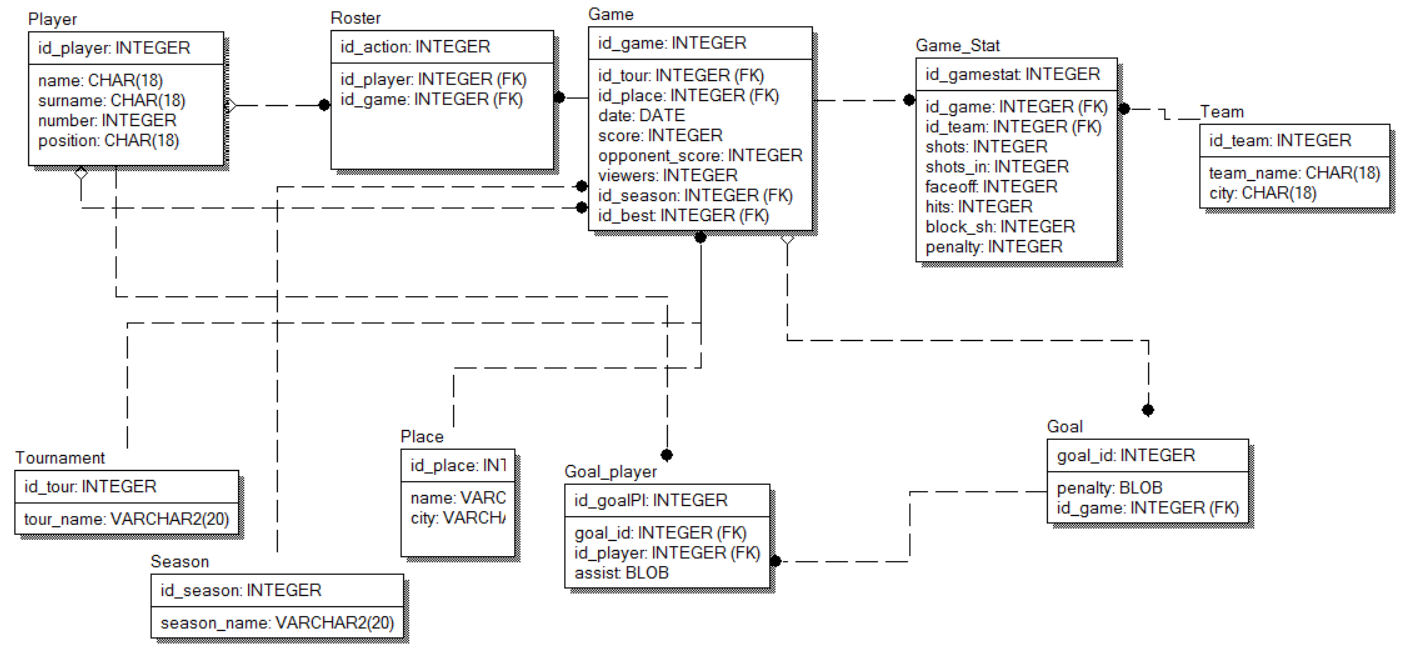


# Схема в ErWin

## Логический уровень:



## Физический уровень:



## Описание сгенерированной БД на DDLSQL для VisualFoxPro:

**CREATE TABLE Game**

**(**

**id\_game INTEGER NOT NULL ,**

**id\_tour INTEGER NOT NULL ,**

**id\_place INTEGER NOT NULL ,**

**date DATE NULL ,**

**score INTEGER NULL ,**

**opponent\_score INTEGER NULL ,**

**viewers INTEGER NULL ,**

**id\_season INTEGER NOT NULL ,**

**id\_best INTEGER NULL**

**);**

**CREATE UNIQUE INDEX XPKИгра ON Game**

**(id\_game ASC);**

**ALTER TABLE Game**

**ADD CONSTRAINT XPKИгра PRIMARY KEY (id\_game);**

**CREATE TABLE Game\_Stat**

**(**

**id\_gamestat INTEGER NOT NULL ,**

**id\_game INTEGER NOT NULL ,**

**id\_team INTEGER NOT NULL ,**

**shots INTEGER NULL ,**

**shots\_in INTEGER NULL ,**

**faceoff INTEGER NULL ,**

**hits INTEGER NULL ,**

**block\_sh INTEGER NULL ,**

**penalty INTEGER NULL**

**);**

**CREATE UNIQUE INDEX XPKСтатистика ON Game\_Stat**

**(id\_gamestat ASC);**

**ALTER TABLE Game\_Stat**

**ADD CONSTRAINT XPKСтатистика PRIMARY KEY (id\_gamestat);**

**CREATE TABLE Goal**

**(**

**goal\_id INTEGER NOT NULL ,**

**penalty BLOB NULL ,**

**id\_game INTEGER NULL**

**);**

**CREATE UNIQUE INDEX XPKГол ON Goal**

**(goal\_id ASC);**

**ALTER TABLE Goal**

**ADD CONSTRAINT XPKГол PRIMARY KEY (goal\_id);**

**CREATE TABLE Goal\_player**

**(**

**id\_goalPl INTEGER NOT NULL ,**

**goal\_id INTEGER NOT NULL ,**

**id\_player INTEGER NOT NULL ,**

**assist BLOB NULL**

**);**

**CREATE UNIQUE INDEX XPKГол\_игрока ON Goal\_player**

**(id\_goalPl ASC);**

**ALTER TABLE Goal\_player**

**ADD CONSTRAINT XPKГол\_игрока PRIMARY KEY (id\_goalPl);**

**CREATE TABLE Place**

**(**

**id\_place INTEGER NOT NULL ,**

**name VARCHAR2(20) NULL ,**

**city VARCHAR2(20) NULL**

**);**

**CREATE UNIQUE INDEX XPKМесто ON Place**

**(id\_place ASC);**

**ALTER TABLE Place**

**ADD CONSTRAINT XPKМесто PRIMARY KEY (id\_place);**

**CREATE TABLE Player**

**(**

**id\_player INTEGER NOT NULL ,**

**name CHAR(18) NULL ,**

**surname CHAR(18) NULL ,**

**number INTEGER NULL ,**

**position CHAR(18) NULL**

**);**

**CREATE UNIQUE INDEX XPKИгрок ON Player**

**(id\_player ASC);**

**ALTER TABLE Player**

**ADD CONSTRAINT XPKИгрок PRIMARY KEY (id\_player);**

**CREATE TABLE Roster**

**(**

**id\_action INTEGER NOT NULL ,**

**id\_player INTEGER NULL ,**

**id\_game INTEGER NOT NULL**

**);**

**CREATE UNIQUE INDEX XPKСостав ON Roster**

**(id\_action ASC);**

**ALTER TABLE Roster**

**ADD CONSTRAINT XPKСостав PRIMARY KEY (id\_action);**

**CREATE TABLE Season**

**(**

**id\_season INTEGER NOT NULL ,**

**season\_name VARCHAR2(20) NULL**

**);**

**CREATE UNIQUE INDEX XPKСезон ON Season**

**(id\_season ASC);**

**ALTER TABLE Season**

**ADD CONSTRAINT XPKСезон PRIMARY KEY (id\_season);**

**CREATE TABLE Team**

**(**

**id\_team INTEGER NOT NULL ,**

**team\_name CHAR(18) NULL ,**

**city CHAR(18) NULL**

**);**

**CREATE UNIQUE INDEX XPKКоманда ON Team**

**(id\_team ASC);**

**ALTER TABLE Team**

**ADD CONSTRAINT XPKКоманда PRIMARY KEY (id\_team);**

**CREATE TABLE Tournament**

**(**

**id\_tour INTEGER NOT NULL ,**

**tour\_name VARCHAR2(20) NULL**

**);**

**CREATE UNIQUE INDEX XPKТурнир ON Tournament**

**(id\_tour ASC);**

**ALTER TABLE Tournament**

**ADD CONSTRAINT XPKТурнир PRIMARY KEY (id\_tour);**

**ALTER TABLE Game**

**ADD (CONSTRAINT R\_5 FOREIGN KEY (id\_tour) REFERENCES Tournament (id\_tour));**

**ALTER TABLE Game**

**ADD (CONSTRAINT R\_6 FOREIGN KEY (id\_place) REFERENCES Place (id\_place));**

**ALTER TABLE Game**

**ADD (CONSTRAINT R\_7 FOREIGN KEY (id\_season) REFERENCES Season (id\_season));**

**ALTER TABLE Game**

**ADD (CONSTRAINT R\_9 FOREIGN KEY (id\_best) REFERENCES Player (id\_player) ON DELETE SET NULL);**

**ALTER TABLE Game\_Stat**

**ADD (CONSTRAINT R\_3 FOREIGN KEY (id\_game) REFERENCES Game (id\_game));**

**ALTER TABLE Game\_Stat**

**ADD (CONSTRAINT R\_4 FOREIGN KEY (id\_team) REFERENCES Team (id\_team));**

**ALTER TABLE Goal**

**ADD (CONSTRAINT R\_12 FOREIGN KEY (id\_game) REFERENCES Game (id\_game) ON DELETE SET NULL);**

**ALTER TABLE Goal\_player**

**ADD (CONSTRAINT R\_10 FOREIGN KEY (goal\_id) REFERENCES Goal (goal\_id));**

**ALTER TABLE Goal\_player**

**ADD (CONSTRAINT R\_11 FOREIGN KEY (id\_player) REFERENCES Player (id\_player));**

**ALTER TABLE Roster**

**ADD (CONSTRAINT R\_1 FOREIGN KEY (id\_player) REFERENCES Player (id\_player) ON DELETE SET NULL);**

**ALTER TABLE Roster**

**ADD (CONSTRAINT R\_2 FOREIGN KEY (id\_game) REFERENCES Game (id\_game));**

# Запросы:

## 1. Пара игроков в одной игре

**SELECT** first.id\_player, second.id\_player, first.id\_game;

**FROM** roster first, roster second;

**WHERE** first.id\_game = second.id\_game;

AND first.id\_player > second.id\_player

## 2. Игроки не забившие голов за турнир

**SELECT** \* **FROM** player;

**WHERE** id\_player **NOT** **IN**;

(**SELECT** id\_player **FROM** goal\_player, goal, game;

**WHERE** goal\_player.id\_goal = goal.id\_goal;

**AND** goal.id\_game = game.id\_game;

**AND** game.id\_tour = 1)



## 3. Игроки забившие максимальное число голов за сезон

**SELECT** id\_player , **COUNT**(\*) **AS** goals;

**FROM** goal\_player, goal, game **INTO** CURSOR tmp;

**WHERE** goal\_player.id\_goal = goal.id\_goal;

**AND** goal\_player.assistant = 'F';

**AND** goal.id\_game = game.id\_game;

**AND** game.id\_season = 1;

**GROUP** **BY** id\_player

**SELECT** player.id\_player, player.name, player.surname, tmp.goals;

**FROM** player, tmp;

**WHERE** player.id\_player = tmp.id\_player **ORDER** **BY** goals **DESC**



## 4. Наиболее опасные команды (больше всего забитых голов)

**SELECT** gamestat.id\_team , **SUM**(game.op\_score) **AS** GoalsQTY **FROM** game, gamestat **INTO** CURSOR tmp;

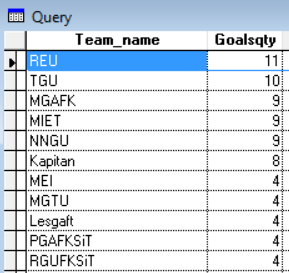
**WHERE** game.id\_game = gamestat.id\_game **AND** gamestat.id\_team != 1;

**GROUP** **BY** gamestat.id\_team

**SELECT** team.team\_name, tmp.GoalsQTY **FROM** team, tmp;

**WHERE** team.id\_team = tmp.id\_team;

**ORDER** **BY** GoalsQTY **DESC**



## 5. Самые удачные турниры (больше всего побед)

**SELECT** id\_tour, **SUM** (**CAST**(game.score > game.op\_score **AS** **INT**)) **AS** wins, **COUNT**(game.id\_game) **AS** games;

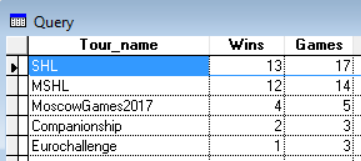
**FROM** game **INTO** CURSOR tmp;

**GROUP** **BY** id\_tour

**SELECT** tour\_name, wins, games **FROM** tournament, tmp;

**WHERE** tournament.id\_tour = tmp.id\_tour;

**ORDER** **BY** wins **DESC**



## 6. Турниры, побед в которых выше среднего

**SELECT** id\_tour, **SUM** (**CAST**(game.score > game.op\_score **AS** **INT**)) / **COUNT**(game.id\_game) **AS** WtoG;

**FROM** game **INTO** CURSOR tmp;

**GROUP** **BY** id\_tour

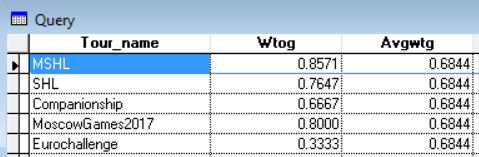
**SELECT** **SUM**(WtoG) / **COUNT**(WtoG) **AS** avgWtG **FROM** tmp **INTO** CURSOR ratmp

**SELECT** tournament.tour\_name, WtoG, avgWtG;

**FROM** tournament, tmp, ratmp;

**WHERE** tournament.id\_tour = tmp.id\_tour;

**AND** tmp.id\_tour > ratmp.avgWtG;



## 7. Игроки не участвовавшие в данном турнире

**SELECT** \* **FROM** player;

**WHERE** id\_player **NOT** **IN** ;

(**SELECT** id\_player **FROM** roster, game ;

**WHERE** roster.id\_game = game.id\_game **AND** game.id\_tour = 1)



Выводы

Данная лабораторная работа еще ближе познакомила меня с базами данных. Хоть моя база и взята из нашего с братом проекта, я в ней ничего не понимал, а её разработкой занимался мой напарник. Теперь же я разобрался, как она работает! Узнал, что такое первичные и вторичные ключи, зачем они нужны. Познакомился с языком SQL, научился писать запросы.

Это было увлекательное задания, выполнив которое, я многому научился. Данные знания и навыки безусловно пригодятся мне в курсовом проекте и дальнейших разработках.