

Министерство науки и высшего образования РФ  
Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Казанский (Приволжский) федеральный университет»

Институт Вычислительной математики и информационных технологий  
Кафедра системного анализа и информационных технологий

## **ОТЧЕТ ПО КУРСУ «ТЕХНОЛОГИИ БАЗ ДАННЫХ»**

Разработка базы данных «Спорт» и клиентского приложения для доступа к  
ней

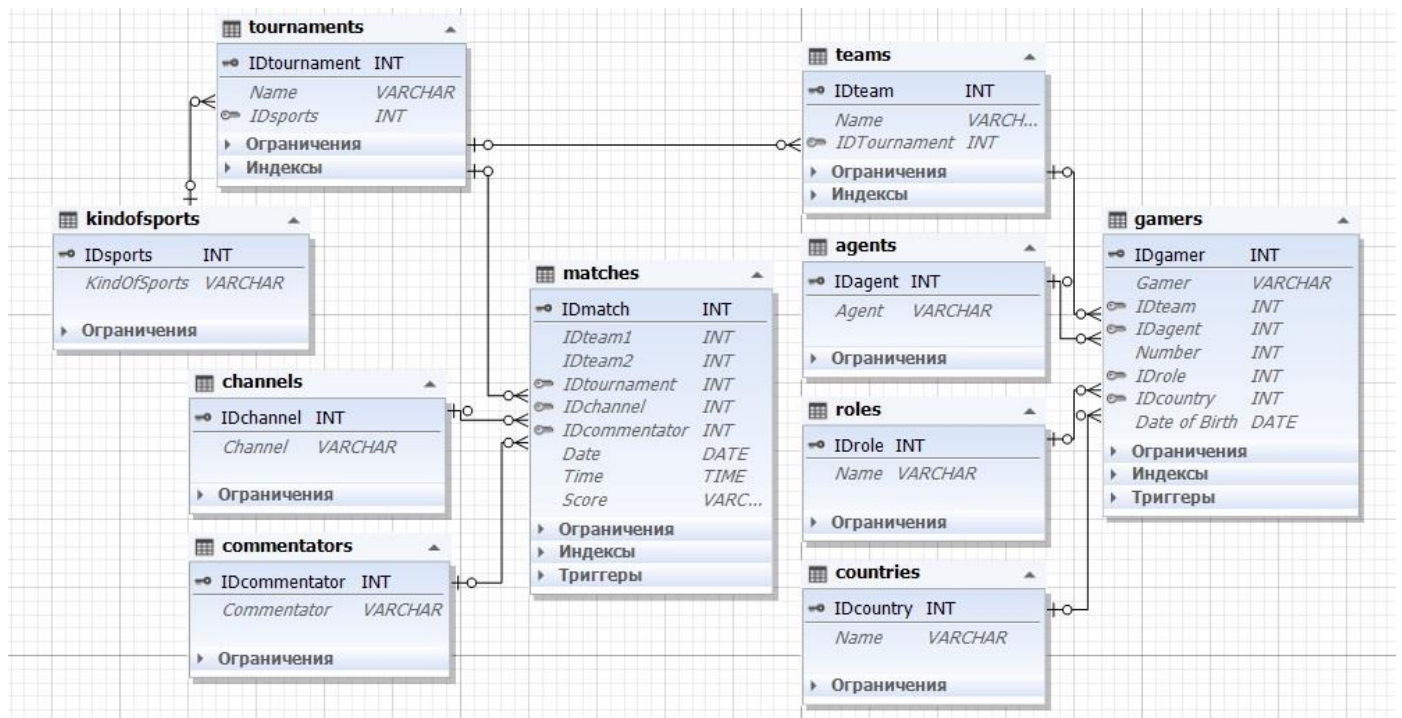
Студент 3 курса

Группа 09-641

Десятов А.Г.

Казань-2018

# 1. Описание реляционной модели данных. Диаграмма базы данных.



С помощью таблицы «gamers» можно узнать данные о игроке: из какой он страны (по «IDcountry» переходим в таблицу «countries»), на какой позиции (по «IDrole» переходим в таблицу «roles») и в какой команде (по «IDteam» переходим в таблицу «teams») он играет, кто является его агентом (по «IDagent» переходим в таблицу «agents»), когда он родился («Date of Birth»). Таблица «teams» содержит список команд, каждой команде соответствует турнир (по «IDtournament» переходим в таблицу «tournaments»), в котором она принимает участие. Таблица «tournaments» содержит список турниров, каждый турнир из определенного вида спорта (по «IDsports» переходим в таблицу «kindofsports»). Таблица «matches» содержит список матчей. Каждый матч комментирует определенный комментатор (по «IDcommentator» переходим в таблицу «commentators»), каждый матч показан по определенному каналу (по «IDchannel» переходим в таблицу «channels»), в матче определенного турнира «IDtournament» участвуют команды «IDteam1» и «IDteam2». В таблице «matches» также есть дата («Date») и время («Time») матча, итоговый счет матча («Score»).

## 2. Запросы к базе данных.

- 1) К таблице команд присоединяется столбик с турниром, в котором она принимает участие (RIGHT JOIN).

```
SELECT IDteam, tournaments.IDTournament
FROM teams RIGHT JOIN tournaments
ON teams.IDTournament = tournaments.IDtournament;
```

- 2) К таблице команд, которые имеют игроков в возрасте от 20 до 28 лет, присоединяется столбик с турниром и видом спорта, сортируется по убыванию по команде, число таких команд ограничивается десятью (INNER JOIN, ORDER BY, LIMIT, NOW(), YEAR(), BETWEEN).

```
SELECT
teams.Name AS Команда,
tournaments.Name AS Турнир,
kindofsports.KindOfSports AS Вид_спорта
FROM sport.teams
RIGHT JOIN sport.tournaments ON teams.IDtournament =
tournaments.IDtournament
INNER JOIN sport.kindofsports ON tournaments.IDsports =
kindofsports.IDsports
WHERE teams.IDteam IN
(SELECT IDteam
FROM sport.gamers
WHERE YEAR(NOW()) - YEAR(Date of Birth) BETWEEN 20 AND
28)
ORDER BY Команда DESC
LIMIT 10;
```

- 3) К таблице игроков добавляется столбик команд, за которые они выступают (LEFT JOIN).

```
SELECT
gamers.Gamer AS Игрок2,
teams.Name AS Команда2
FROM sport.gamers
LEFT JOIN sport.teams ON teams.IDteam = gamers.IDteam
ORDER BY Команда2;
```

- 4) Вывести команды, в которых есть вратари, позицию, которую рассматриваем (в данном случае «вратарь»), вид спорта, из которого команда.

```
SELECT
```

```

teams.Name,
roles.Name,
kindofsports.KindOfSports
FROM gamers
INNER JOIN teams
    ON gamers.IDteam = teams.IDteam
INNER JOIN roles
    ON gamers.IDrole = roles.IDrole
INNER JOIN tournaments
    ON teams.IDTournament = tournaments.IDtournament
INNER JOIN kindofsports
    ON tournaments.IDsports = kindofsports.IDsports
GROUP BY teams.IDteam,
    kindofsports.KindOfSports
HAVING roles.Name = "вратарь"

```

5) Вывести игроков, возраст которых не превосходит 22 года  
(Вложенный запрос).

```

SELECT
    gamers.Gamer,
    gamers.`Date of Birth`
FROM gamers
WHERE IDgamer IN (SELECT IDgamer
    FROM gamers
    WHERE YEAR(NOW()) - YEAR(`Date of Birth`) < 22)

```

6) Вывести всех защитников (дважды вложенный запрос)

```

SELECT
    gamers.Gamer,
    roles.Name
FROM sport.gamers
INNER JOIN sport.roles
    ON gamers.IDrole = roles.IDrole
WHERE gamers.IDgamer IN (SELECT
    gamers.IDgamer
    FROM sport.gamers
    WHERE gamers.IDrole IN (SELECT
        roles.IDrole
        FROM sport.roles
        WHERE roles.Name = "Защитник"))

```

7) Вывести всех игроков, у которых командные номера от 5 до 15.

```

SELECT
  gamers.*
FROM gamers
  WHERE IDgamer IN
    (SELECT IDgamer FROM gamers
     WHERE Number BETWEEN 5 AND 15)

```

8) Все игроки, которых обслуживает агент с ID = 5.

```

SELECT
  gamers.Gamer,
  agents.Agent
FROM gamers
  INNER JOIN agents
    ON gamers.IDagent = agents.IDagent
WHERE gamers.IDagent IN (SELECT
  agents.IDagent
  FROM agents
  WHERE agents.IDagent = 5)

```

9) Все игроки, которые выступают за команду с ID = 1.

```

SELECT * FROM sport.gamers
  WHERE IDgamer IN (SELECT IDgamer
  FROM sport.gamers
  WHERE IDteam = 1);

```

10) Вывести игроков, которые играют на позиции с ID = 1, и их дату рождения.

```

SELECT Gamer, `Date of Birth`
FROM gamers
  WHERE IDrole = 1

```

11) Изменить время во всех матчах, где время = «null», на «00:00:00» (UPDATE, SET).

```

UPDATE matches
SET Time = '00:00:00'
WHERE Time = null

```

12) Изменить счёт определенного матча.

```

UPDATE matches
SET Score = "1-0"
WHERE IDmatch = 4

```

- 13) Вывести игроков со всех видов спорта, у которых командный номер = 1.

```
SELECT
gamers.Gamer,
kindofsports.KindOfSports,
gamers.Number
FROM gamers
INNER JOIN teams
ON gamers.IDteam = teams.IDteam
INNER JOIN tournaments
ON teams.IDTournament = tournaments.IDtournament
INNER JOIN kindofsports
ON tournaments.IDsports = kindofsports.IDsports
WHERE gamers.Number = 1
```

- 14) Вывести всех игроков из страны с ID = 2.

```
SELECT
gamers.Gamer,
countries.Name
FROM gamers
INNER JOIN countries
ON gamers.IDcountry = countries.IDcountry
WHERE gamers.IDcountry = 2
```

- 15) Список команд из турнира с ID = 1.

```
SELECT Name
FROM sport.teams
WHERE IDtournament = 1
```

- 16) Комментаторы с ID от 3 до 5.

```
SELECT
commentators.Commentator,
commentators.IDcommentator
FROM commentators WHERE IDcommentator BETWEEN 3 AND 5
```

- 17) Вывести футбольные турниры

```
SELECT Name
FROM tournaments
LEFT JOIN sport.kindofsports
ON tournaments.IDsports = kindofsports.IDsports
WHERE KindOfSports = 'Футбол'
```

- 18) Вывести информацию о матче в виде слов, а не в виде ID, как в базе.

```
SELECT k.KindOfSports, Date, Time,  
t1.Name AS team1,  
t2.Name AS team2,  
Score AS score,  
tournaments.Name AS tournament,  
ch.Channel AS channel,  
com.Commentator AS commentator  
FROM sport.matches  
LEFT JOIN sport.teams t1  
ON matches.IDteam1 = t1.IDteam  
LEFT JOIN sport.teams t2  
ON matches.IDteam2 = t2.IDteam  
LEFT JOIN sport.tournaments  
ON matches.IDtournament = tournaments.IDtournament  
LEFT JOIN sport.channels ch  
ON matches.IDchannel = ch.IDchannel  
LEFT JOIN sport.commentators com  
ON matches.IDcommentator = com.IDcommentator  
LEFT JOIN sport.kindofsports k  
ON tournaments.IDsports = k.IDsports  
ORDER BY k.KindOfSports, tournament, date, Time
```

- 19) Вывести список команды, сообщив всю информацию об игроках в виде слов

```
SELECT roles.Name AS Ампула,  
Number AS Номер,  
Gamer AS Игрок,  
countries.Name AS Страна,  
'Date of Birth' AS Дата  
FROM sport.gamers  
LEFT JOIN sport.roles  
ON gamers.IDrole = roles.IDrole  
LEFT JOIN sport.countries  
ON gamers.IDcountry = countries.IDcountry  
WHERE IDteam = 1  
ORDER BY roles.IDrole, Номер, Игрок
```

- 20) Вывести страны, которые начинаются на «Р» (LIKE, «%»).

```
SELECT  
countries.Name
```

```
FROM countries
WHERE countries.Name LIKE "P%"
```

21) Удаление матча по его ID (DELETE).  
DELETE FROM matches WHERE IDmatch = 3

### 3. Серверные процедуры и функции.

#### 1) Процедуры

А) На вход передается ID агента, выводится средний возраст игроков, которых он обслуживает.

```
CREATE DEFINER = 'root'@'localhost'
PROCEDURE sport.middle_age(IN IdAgent0 int, OUT middle_age float)
BEGIN
    SELECT
        AVG(YEAR(NOW()) - YEAR(`Date of Birth`)) INTO middle_age
    FROM gamers
    WHERE IDagent = IdAgent0;
END
```

Б) На вход строкой передается вид спорта. Выводим список всех команд этого вида спорта из базы данных.

```
CREATE DEFINER = 'root'@'localhost'
PROCEDURE sport.teams(IN kindSport_str text)
BEGIN
    SELECT teams.Name FROM teams
    INNER JOIN tournaments
    ON teams.IDTournament = tournaments.IDtournament
    INNER JOIN kindofsports
    ON KindOfSports.IDsports = tournaments.IDsports
    WHERE KindOfSports = kindSport_str;
END
```

В) На вход максимальный возраст игрока, сначала выводим всех игроков, не старше данного возраста, а затем выводим количество таких игроков.

```
CREATE DEFINER = 'root'@'localhost'
PROCEDURE sport.younger(IN age int, OUT count INT)
BEGIN
    SELECT
        *
```



```
FROM gamers  
WHERE YEAR(NOW()) - YEAR(`Date of Birth`) <= age;
```

```
SELECT  
COUNT(*) INTO count  
FROM gamers  
WHERE YEAR(NOW()) - YEAR(`Date of Birth`) <= age;
```

```
END
```

## 2) Функции

А) На вход получаем строку с именем игрока, или его фамилией, или частью его имени или фамилии. У первого найденного такого игрока определяется, из какой он страны. Возвращается страна.

```
CREATE DEFINER = 'root'@'localhost'  
FUNCTION sport.where_is_he_from(name_gamer varchar(255))  
  RETURNS varchar(255) CHARSET utf8  
  READS SQL DATA  
BEGIN  
  DECLARE idCountryp int;  
  DECLARE NameCountry varchar(255);  
  SELECT IDcountry INTO idCountryp  
    FROM gamers  
    WHERE POSITION(name_gamer IN gamers.Gamer) != 0  
    LIMIT 1;  
  SELECT Name INTO NameCountry  
    FROM countries  
    WHERE IDcountry = idCountryp  
    LIMIT 1;  
  RETURN NameCountry;  
END
```

Б) По ID матча составляется строка не в виде ID канала, комментатора, команд и так далее, а в виде слов. CONCAT объединяет в одну строку несколько подстрок. Подстроки получаются с помощью select-запросов. Возвращается итоговая строка.

```
CREATE DEFINER = 'root'@'localhost'  
FUNCTION sport.get_str_about_match(idmatchp int)  
  RETURNS varchar(255) CHARSET utf8  
  READS SQL DATA  
BEGIN
```

```
DECLARE str VARCHAR(255);
DECLARE nameteam1,
        nameteam2,
        nametournament,
        namechannel,
        namecommentator VARCHAR(255);
DECLARE idteam1p,
        idteam2p,
        idtournamentp,
        idchannelp,
        idcommentatorp int;
```

```
SELECT
    IDteam1, IDteam2, IDtournament, IDchannel, IDcommentator
    INTO
    idteam1p, idteam2p, idtournamentp, idchannelp, idcommentatorp
FROM matches
WHERE IDmatch = idmatchp
LIMIT 1;
```

```
SELECT
    Name INTO nameteam1
FROM teams
WHERE IDteam = idteam1p
LIMIT 1;
```

```
SELECT
    Name INTO nameteam2
FROM teams
WHERE IDteam = idteam2p
LIMIT 1;
```

```
SELECT
    Name INTO nametournament
FROM tournaments
WHERE IDtournament = idtournamentp
LIMIT 1;
```

```
SELECT
    Channel INTO namechannel
FROM channels
WHERE IDchannel = idchannelp
LIMIT 1;
```

```
SELECT
    Commentator INTO namecommentator
```

```
FROM commentators
WHERE IDcommentator = idcommentatorp
LIMIT 1;
```

```
SET str = CONCAT(namechannel, ". ", nameteam1, " - ",nameteam2, ". ",
nametournament, ". ", namecommentator);
RETURN str;
END
```

В) На вход ID канала, на выходе: сколько матчей было показано, на этом канале.

```
CREATE DEFINER = 'root'@'localhost'
FUNCTION sport.amount_matches_on_channel(idchannelp INT)
RETURNS int(11)
READS SQL DATA
BEGIN
DECLARE amount int;
SELECT COUNT(*) INTO amount
FROM matches
WHERE IDchannel = idchannelp;
RETURN amount;
END
```

### 3) Триггеры

А) Перед вставкой нового игрока, если в IDagent был передан «null», то присваивается IDagent = 1. Что соответствует «Нет агента». Реализация значения по умолчанию.

```
CREATE
DEFINER = 'root'@'localhost'
TRIGGER sport.tr_bef_ins_gamer
BEFORE INSERT
ON sport.gamers
FOR EACH ROW
BEGIN
IF NewIDagent = NULL THEN
SET New.IDagent = 1;
END IF;
END
```

Б) После обновления матчей, подсчитывается у скольких матчей изменилась дата проведения. Значение записывается в переменную, чтобы потом вывести куда-нибудь, например, в debug.

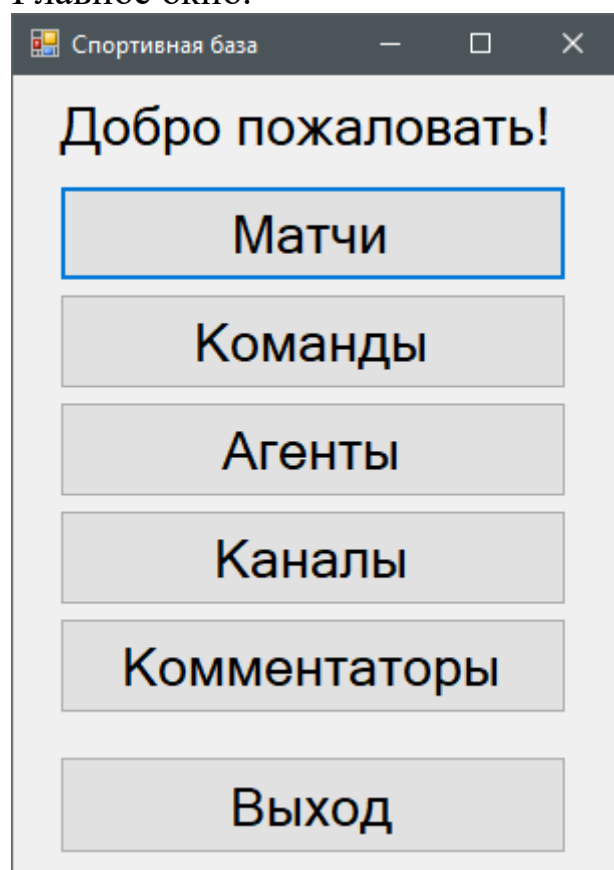
```
CREATE
    DEFINER = 'root'@'localhost'
    TRIGGER sport.tr_aft_upd_date_match
    AFTER UPDATE
    ON sport.matches
    FOR EACH ROW
BEGIN
    DECLARE amount int;
    DECLARE str varchar(255);
    SELECT COUNT(*) INTO amount
    FROM matches
    WHERE OLD.Date != NEW.Date;
    SET str = CONCAT(amount, " матчей перенесено");
END
```

В) Перед вставкой матча, если в Score был передан «null», то присваивается новый счёт «0-0». Реализация значения по умолчанию.

```
CREATE
    DEFINER = 'root'@'localhost'
    TRIGGER sport.tr_bef_ins_match
    BEFORE UPDATE
    ON sport.matches
    FOR EACH ROW
BEGIN
    IF New.SCORE = NULL THEN
        SET New.Score = "0-0";
    END IF;
END
```

#### 4. Клиентское приложение.

Главное окно.



Раздел «Матчи». С помощью кнопки «Изменить счет на» можно менять текущий счет матча, новый счёт сразу обновится в базе данных. С помощью кнопки «Добавить матч» можно перейти в окно добавления матча в базу данных.

	Вид	Турнир	Дата	Время	Хозяева	Гости	Счёт	Канал	Комментатор
▶	Футбол	РФПЛ	25.10.2018	14:30:00	ЦСКА	Зенит	1-0	Eurosport 1	Александр Шмурнов
	Футбол	РФПЛ	06.12.2018	18:00:00	Динамо	Арсенал	0-0	МАТЧ! ФУТБОЛ 1	Денис Казанский
	Футбол	РФПЛ	06.12.2018	20:30:00	Спартак	Локомотив	0-2	Eurosport 1	Константин Генич
	Футбол	РФПЛ	06.12.2018	23:00:00	Оренбург	Ахмат	3-1	Eurosport News	Александр Ткачёв
	Футбол	РФПЛ	09.12.2018	22:00:00	Анжи	Енисей	0-3	МАТЧ! ФУТБОЛ 3	Юрий Розанов
	Футбол	ФНЛ	04.12.2018	00:00:00	Томь	Тамбов	3-3	МАТЧ! ФУТБОЛ 2	Нобель Арустамян
	Хоккей	КХЛ	03.12.2018	20:00:00	Локомотив	СКА	1-3	Телеканал КХЛ	Олег Мосалёв
	Волейбол	Суперлига	05.12.2018	20:40:00	Зенит-Казань	Зенит СПб	3-2	МАТЧ! ИГРА	Виктор Гусев
	Волейбол	Суперлига	12.12.2018	21:35:00	Факел	Урал	3-1	МАТЧ ТВ	Александр Ткачёв
	Баскетбол	Лига ВТБ	06.12.2018	21:30:00	Химки	УНИКС	76-81	МАТЧ! АРЕНА	Владимир Гомельский
	Баскетбол	Лига ВТБ	10.12.2018	22:45:00	Нижний Новгород	Енисей	105-95	Eurosport Gold	Владимир Гомельский
*									

Изменить счёт на    1-0    Добавить матч

Окно «Добавить матч».

Добавить матч

Дата и время: 14.12.2018 20:30

Вид спорта: Баскетбол

Турнир: Лига ВТБ

Хозяева: Зенит

Гости: ЦСКА

Счёт: 85-95

Комментатор: Владимир Гомельский

Канал: МАТЧ ТВ

Добавить

Декабрь 2018

Пн	Вт	Ср	Чт	Пт	Сб	Вс
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

Сегодня: 14.12.2018

Раздел «Команды». Здесь можно выбрать команду и посмотреть список ее игроков. А также здесь можно узнать средний возраст игроков команды.

Команды

Вид спорта: Футбол

Турнир: РФПЛ

Команда: ЦСКА

Средний возраст: 22,82143

	Амплуа	Номер	Игрок	Страна	Дата
▶	Вратарь	35	Игорь Акинфеев	Российская Федерация	08.04.1986
	Защитник	2	Марио Фернандес	Российская Федерация	19.09.1990
	Защитник	3	Никита Чернов	Российская Федерация	14.01.1996
	Защитник	5	Виктор Васин	Российская Федерация	06.10.1988
	Защитник	14	Кирилл Набабкин	Российская Федерация	08.09.1986
	Защитник	23	Хердур Магнуссон	Исландия	11.02.1993
	Защитник	42	Георгий Щенников	Российская Федерация	27.04.1991
	Защитник	50	Родриго Бекао	Бразилия	19.01.1996
	Полузащитник	8	Никола Влашич	Хорватия	04.10.1997
	Полузащитник	10	Алан Дзагоев	Российская Федерация	17.06.1990
	Полузащитник	15	Павел Ефремов	Российская Федерация	01.04.1995

Раздел «Агенты». В данном окне можно выбрать агента, и посмотреть список игроков, которых он обслуживает. А также здесь можно найти игроков, у которых нет агента.

