Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждения высшего образования

«Национальный исследовательский технологический университет «МИСиС»

Контрольное домашнее задание

по дисциплине

«Базы данных»

на тему:

«Создание базы данных «Деятельность спортивного клуба»»

Направление подготовки:

09.03.01 «Информатика и вычислительная техника»  
Семестр 3

|  |  |
| --- | --- |
| Выполнил: | Проверила: |
| Коншин Кирилл Павлович  (Ф.И.О. студента) | Валова Анастасия Александровна  (Ф.И.О преподавателя) |
| БИВТ-21-10  (№ группы) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка) |
| 23.12.2022  (дата сдачи) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата проверки) |
| Подпись: \_\_\_\_\_\_\_\_\_\_\_\_ | Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_ |

Оглавление

Постановка задачи 3

Описание структуры БД 4

Описание структуры БД (Вербальная модель) 4

Описание структуры БД (Реляционная модель) 6

Описание структуры БД (Анализ функциональных зависимостей) 8

Заполнение БД информацией 9

Описание представлений 11

Описание функций 14

Описание хранимых процедур 17

Описание триггеров 19

Пример работы с БД с использованием созданных объектов 20

Используемая литература 22

**Постановка Задачи**

Своя тема. Предметная область: МССИ (Московские Студенческие Спортивные Игры (Мини-футбол)).

Создание базы данных по тематике «МССИ», создание 10 связанных таблиц, заполнение этих таблиц данными и разработка функциональных возможностей посредством триггеров, функция, процедур и представлений.

Эта база данных может помочь администраторам команд укомплектовывать составы команд, а также функционерам автоматизировать процесс обработки данных, рядовому пользователю следить за проведением игр.

**Описание структуры БД**

*Вербальная модель*

База данных состоит из 10 таблиц и содержит (описывает) следующие сущности:

1). Игроки.

Здесь содержится информация об игроках: его уникальный идентификатор, идентификатор команды, полное имя, возраст, вес, дата получения лицензии, игровой номер, показано наличие травмы.

2). Тренеры.

Содержится информация о тренере. Его идентификатор, полное имя и описание тренерского опыта.

3). Команды.

Содержит информацию о командах: идентификатор команды, ее название, количество игроков в составе, дивизион, идентификаторы тренера и спонсора, а также количество очков в текущем розыгрыше турнира.

4). Спонсоры.

Содержится информация Спонсорах команд. Идентификатор, название команды, а также контакт представителя и его телефонный номер для связи – атрибуты этой сущности.

5). Травмы

Содержит информацию о травмах, полученных всеми спортсменами. Указывается какой спортсмен получил травму.

6). Квалификация судей

Содержит информацию о квалификации судей, обслуживающих матчи МССИ: тип квалификации, описание типа, заработная плата за проведенный матч.

7). Судьи

Содержит информацию о судьях. Полное имя, квалификация, количество обслуживаемых матчей в сезоне, гонорар за розыгрыш кубка

8). Группы

Содержит информацию о дивизионах в розыгрыше МССИ данного года. Наполнение: идентификатор, название дивизиона

9). Арены

Содержит информацию о спортивных объектах. А именно название, вместимость, адрес

10). Информация о спонсорстве

Сущность, иллюстрирующая информацию о спонсорском контракте команды и компании-спонсора. Содержит Составной ключ идентификатор спонсора-идентификатор команды и бюджет, выделяемый на команду

11). Матчи

Сущность для работы с информацией по матчам. Содержит идентификатор матча, арены, команд, судьи., результат матча, идентификатор команды-победителя, время и дата проведения

*Реляционная модель*

Ниже приведён список соответствия каждой сущности и таблиц

1). Игроки – Players

2). Тренеры - Coaches

3). Команды - Teams

4). Спонсоры. - Sponsors

5). Травмы - InjuriesType

6). Группы - GROUPS

7). Квалификация судей - Qualification

8). Судьи - Judges

9). Арены - Arena

10) Информация о спонсорстве – SponsoringInfo

11) ). Матчи – Matches

На рисунке №1 представлена схема и связи между таблицами

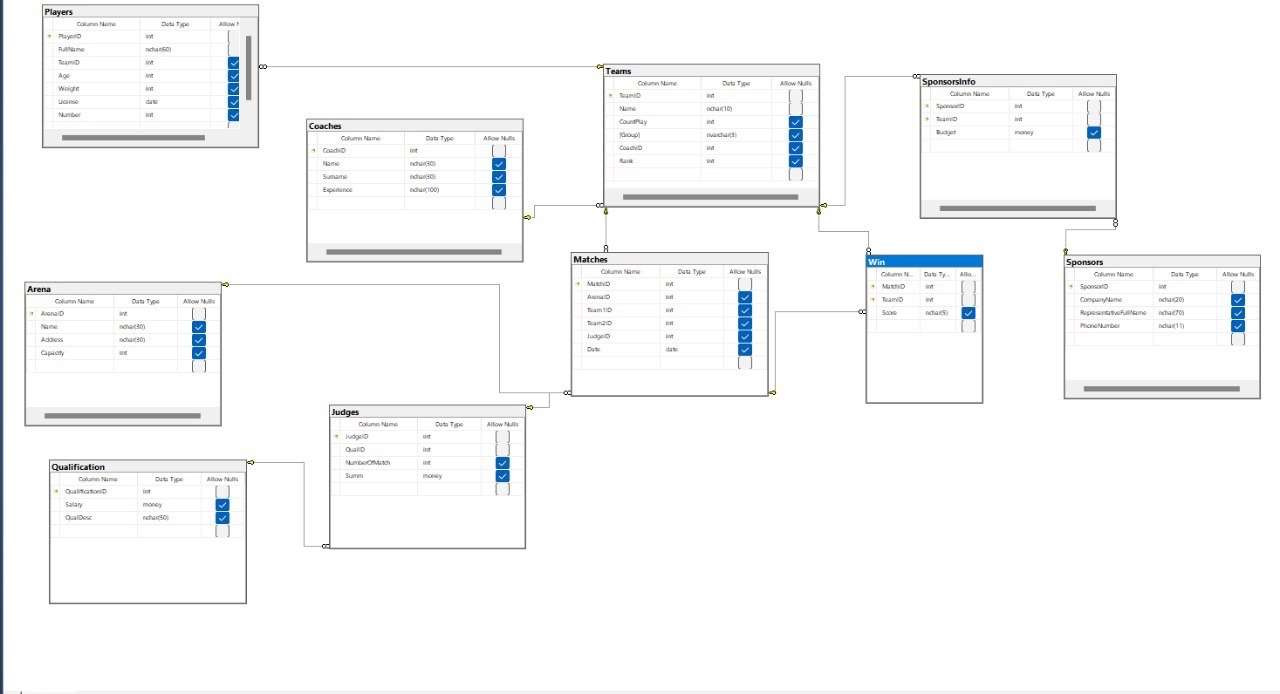


Рисунок №1.

**Анализ функциональных зависимостей**

В данной работе соблюдаются функциональные зависимости, нет дубликатов данных и каких-то несоответствий. Дизайн схемы отношений понятен и прост. В разработанной базе данных соблюдена 3-я нормальная форма и форма Бойса-Кодда.

Например, в таблице SponsoringInfo создан составной ключ для поддержания нормальной формы и правильной связи между отношениями.

**Заполнение БД информацией**

Основным источником данных является ручное заполнение.

Пример заполнения представлен на рисунке 2.

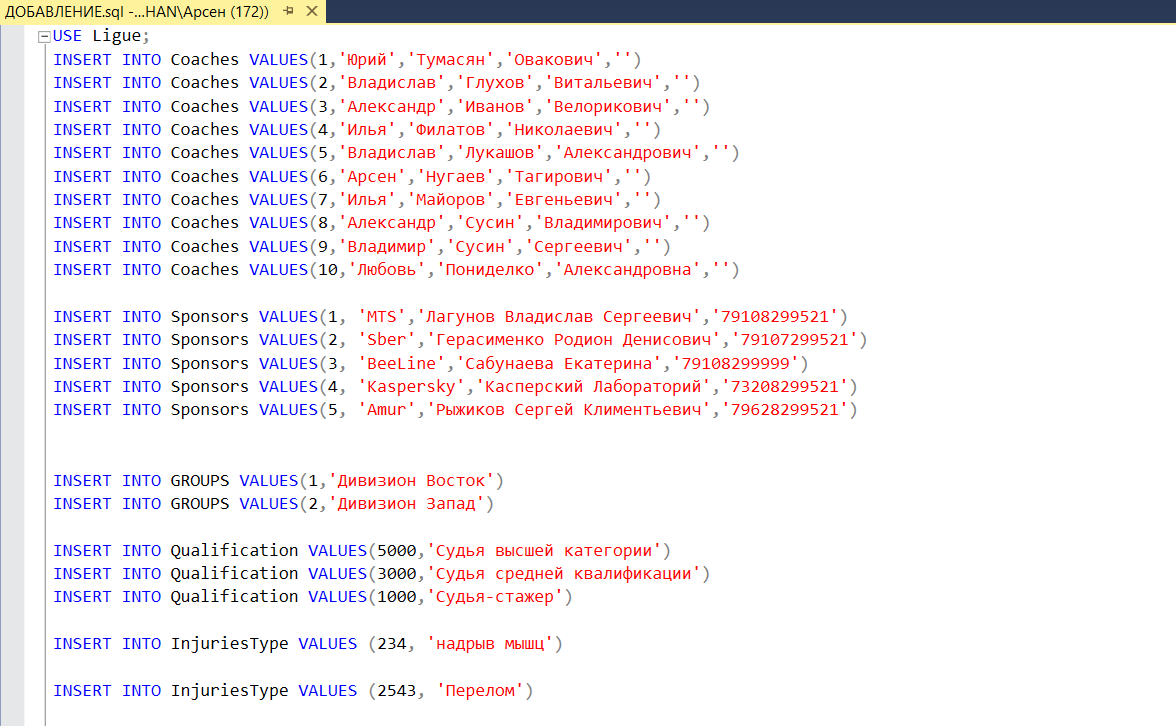


Рисунок 2

**Описание представлений**

Представление №1. (рисунок №3)

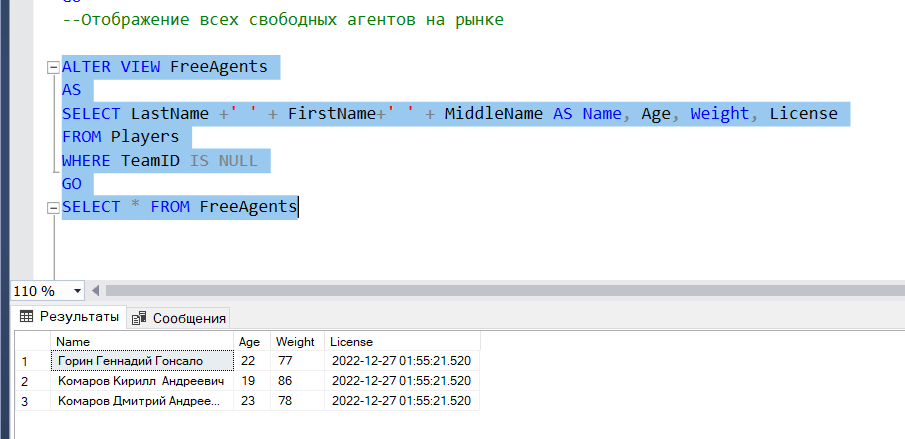


Рисунок 3

Данное представление отображает всех свободных агентов, которые зарегистрированы в лиге

Представление №2. (рисунок №4)

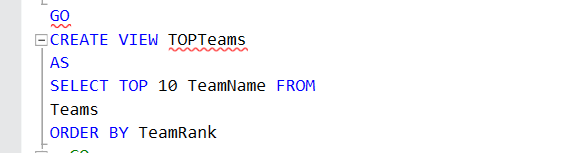


Рисунок 4

Данное представление выводит список топ – 10 команд лиги, которое будет удобно отобразить на сайте МССИ

Представление №3 (рисунок №5)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5

Данное представление выводит виды спорта, в которых спортивный клуб получил наибольшее числа медалей.

Итого:

Сами представления очень удобны, если необходимо посмотреть информацию, содержащуюся в нескольких таблицах, именно для этого они и нужны.

Теория:

Представление (VIEW) — объект базы данных, являющийся результатом выполнения запроса к базе данных, определенного с помощью оператора SELECT, в момент обращения к представлению.  
  
Представления иногда называют «виртуальными таблицами». Такое название связано с тем, что представление доступно для пользователя как таблица, но само оно не содержит данных, а извлекает их из таблиц в момент обращения к нему. Если данные изменены в базовой таблице, то пользователь получит актуальные данные при обращении к представлению, использующему данную таблицу; кэширования результатов выборки из таблицы при работе представлений не производится. При этом, механизм кэширования запросов (query cache) работает на уровне запросов пользователя безотносительно к тому, обращается ли пользователь к таблицам или представлениям.

**Описание функций**

Функция №1 (рисунок №6)

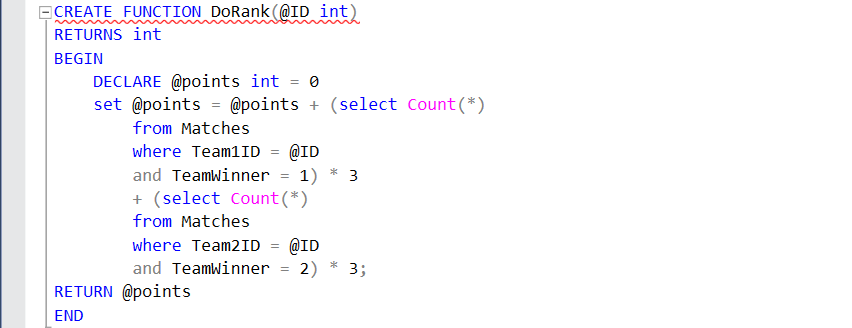


Рисунок №6

По уникальному идентификатору команда функция считает количетсво очков, которые заработала команда в текущем розыгрыше

Функция №2 (рисунок №7)

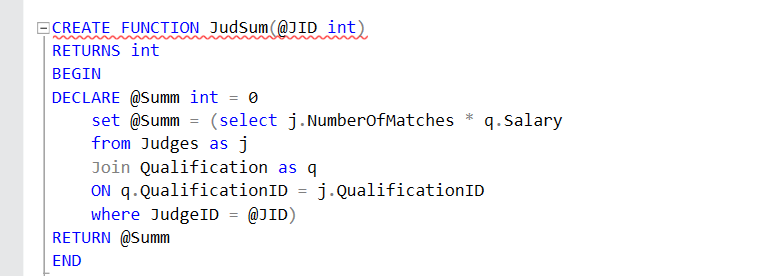


Рисунок №7

Данная функция выводит сумму, которую получит судья за отработанные матчи в течении сезона.

Функция №3 (рисунок №8)

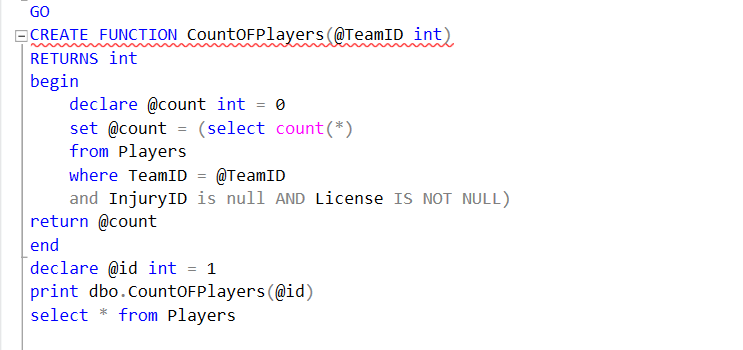


Рисунок №8

Данная скалярная функция принимает на вход идентификатор команды и выводит количество игроков, фактически выступающих за команду (нет травм, и есть лицензия).

Теория:

Определяемая пользователем функция представляет собой подпрограмму, которая принимает параметры, выполняет действия, такие как сложные вычисления, а затем возвращает результат этих действий в виде значения. Возвращаемое значение может быть скалярным значением или таблицей.

Скалярные функции – это функции, которые возвращают одно значение заданного типа данных;

Табличные функции – это функции, которые возвращают табличные данные, т.е. данные типа TABLE.

**Описание хранимых процедур**

Процедура №1 (рисунок №9)

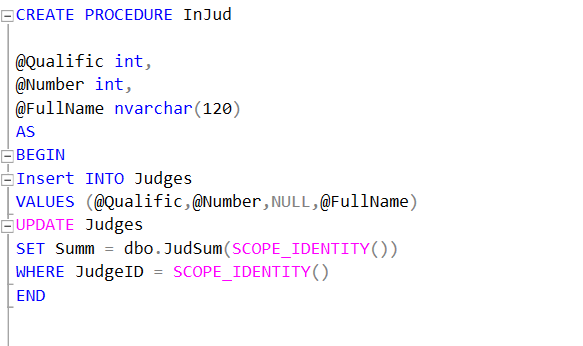


Рисунок №9

Эта процедура добавляет нового судью и сразу считает его гонорар.

Процедура №2 (рисунок №10)

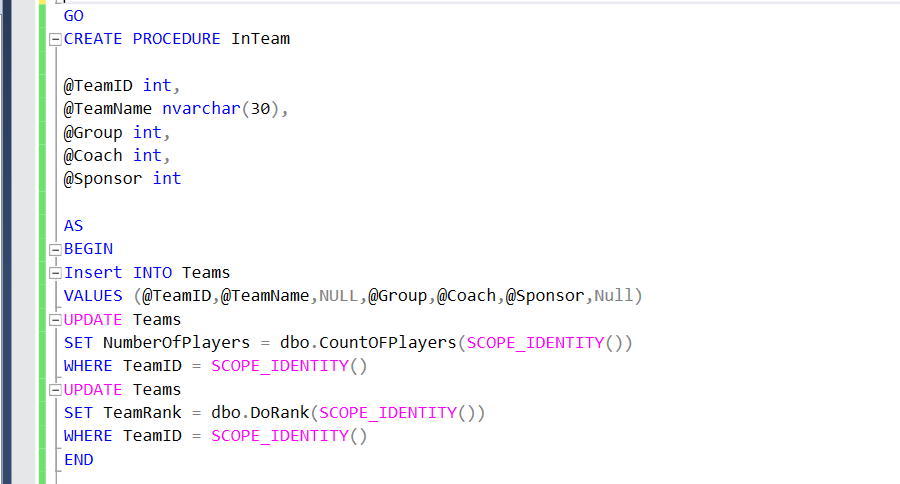


Рисунок №10

Процедура добавляет новую команду в турнир, считая сразу ее количество очков, и фактическое количество игроков.

Процедура №3 (рисунок №11)

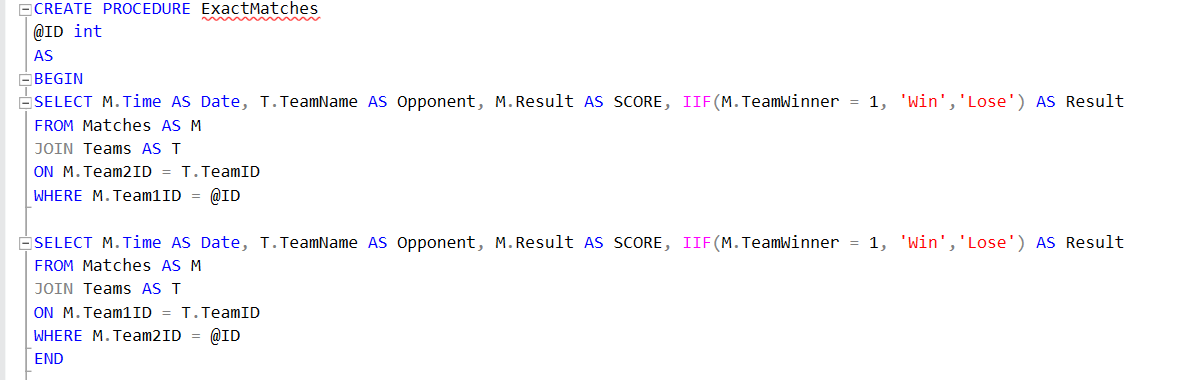


Рисунок №11

Процедура создана, чтобы получать информацию по всем матчам команды

Процедура№4(рисунок 12,13)

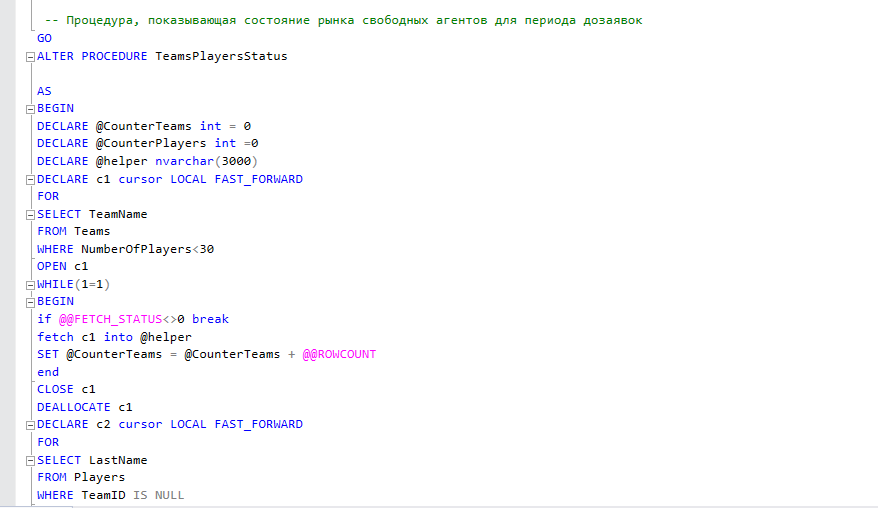


Рисунок 12

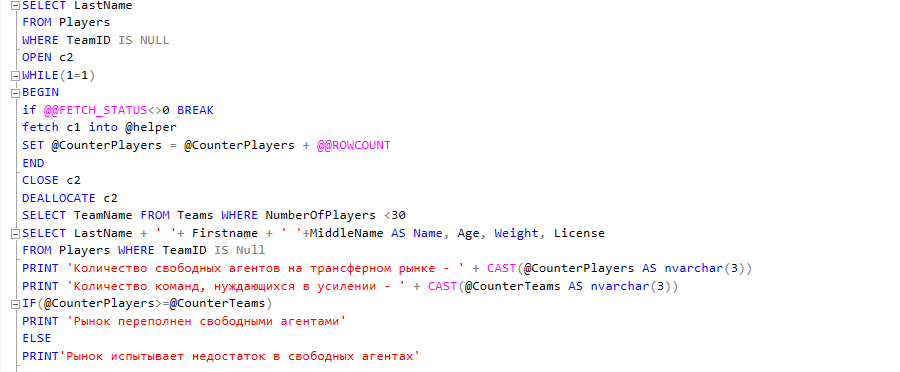


Рисунок №13

Данная процедура дает информацию о заполненности команд в текущем розыгрыше, количество игроков, претендующих на дозаявку, а также состояние рынка.

Теория:

Хранимая процедура в SQL Server представляет собой группу из одного или нескольких операторов Transact-SQL или ссылку на метод среды CLR Microsoft .NET Framework.

**Описание триггеров**

Триггер №1 (рисунок №14)

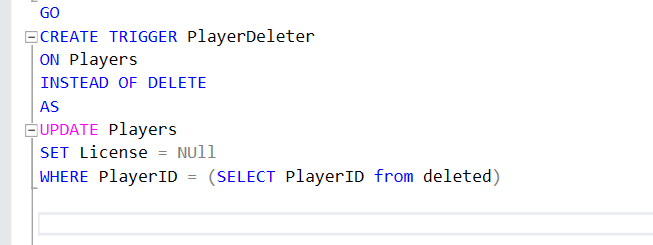


Рисунок №14

Данный триггер срабатывает при удалении игроков, вместо удаления он убирает лицензию на игры, позволяя функционерам заявить его в дальнейшем

Триггер №2 (рисунок №15)

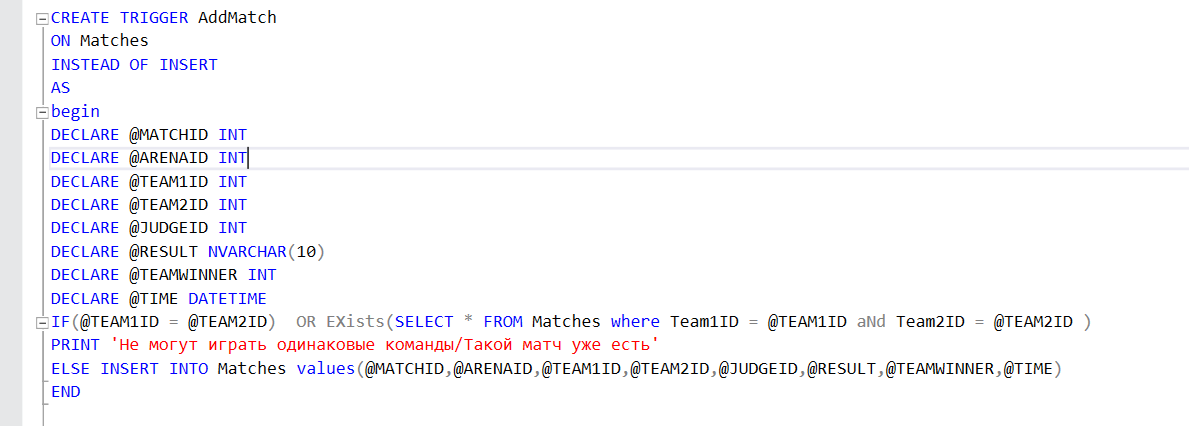


Рисунок №15

Данный триггер срабатывает при добавлении нового матча, не давая скрестить одну команду с ней же в матче

Триггер №3 (рисунок №16)

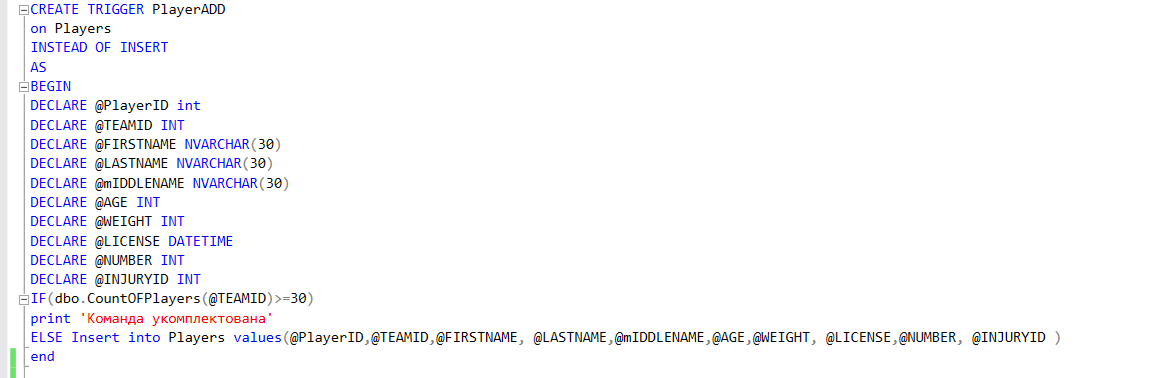


Рисунок № 16

Данный триггер не позволяет добавить в команду человека, если в команда укомплектована с учетом травм

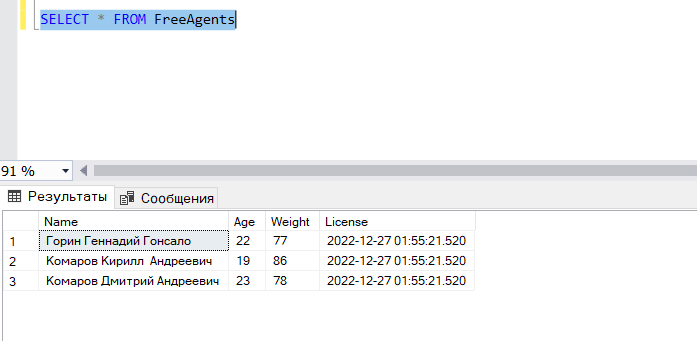
Теория:

**Триггер** **в** MSSQL — **это** определяемая пользователем **SQL**-команда, которая автоматически вызывается во время операций INSERT, delete или UPDATE. Код **триггера** связан с таблицей и уничтожается после удаления таблицы. Вы можете определить время действия **триггера** и указать, когда его нужно активировать – до или после определенного события базы данных.

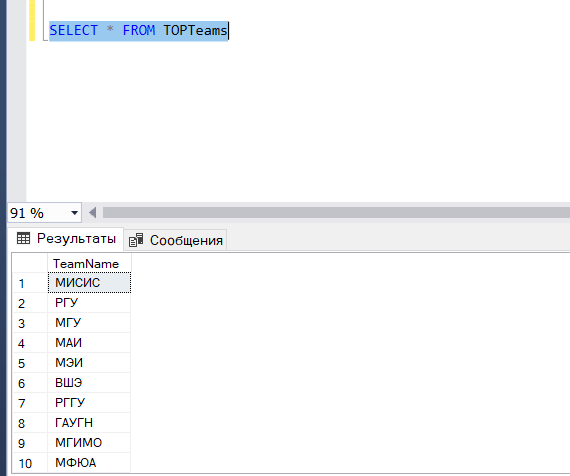
**Пример работы с БД с использованием созданных объектов**

Представления:

1. Данное представление помогает увидеть всех свободных и доступных агентов на рынке

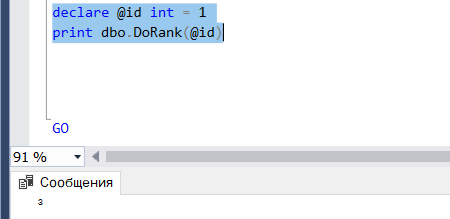


1. Список 10 лучших команд лиги.

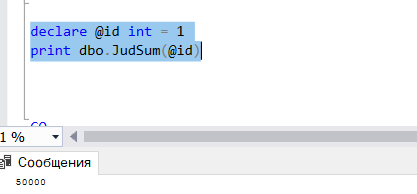


Функции

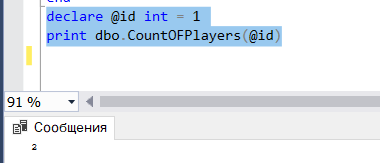
1. Функция помогает узнать количество очков, заработанных командой



1. Показывает гонорар судьи за лигу

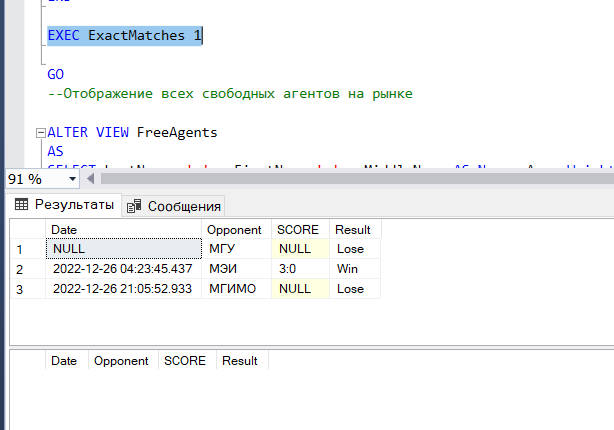


1. Возвращает фактическое количество игроков доступное для игры

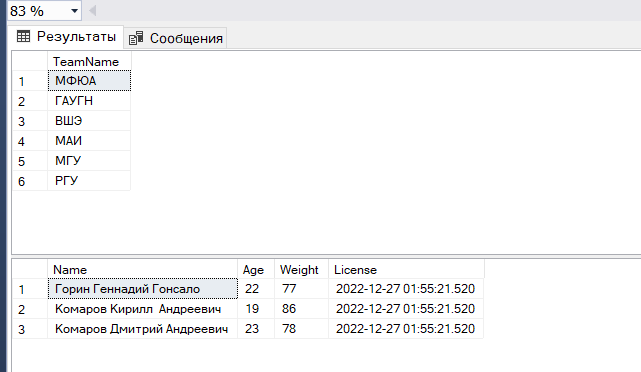


Хранимые процедуры

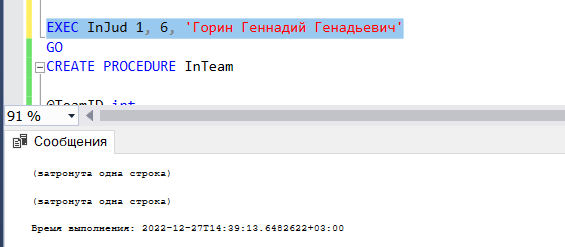
1. Процедура показывает расписание всех матчей команды: Сначала домашние, потом гостевые



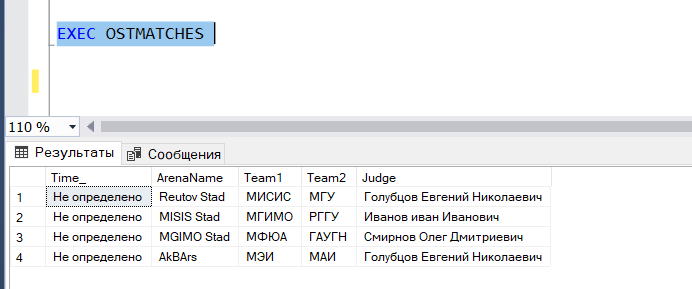
1. Процедура говорит о состоянии трансферного рынка, также показывая все команды, нуждающиеся в усилении



1. Процедура помещает нового судью

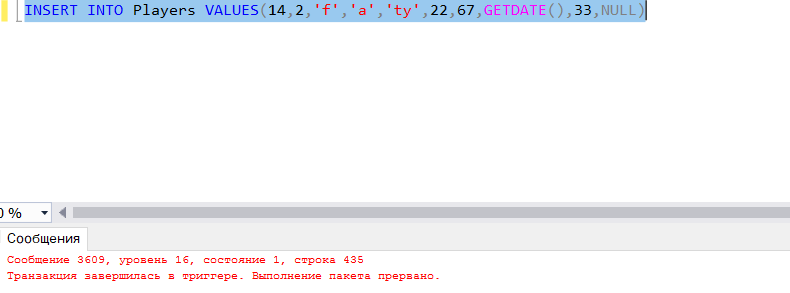


1. Оставшиеся матчи процедура

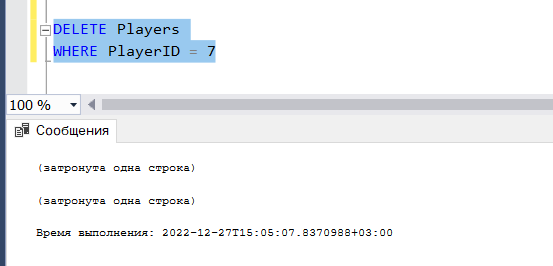


Триггеры

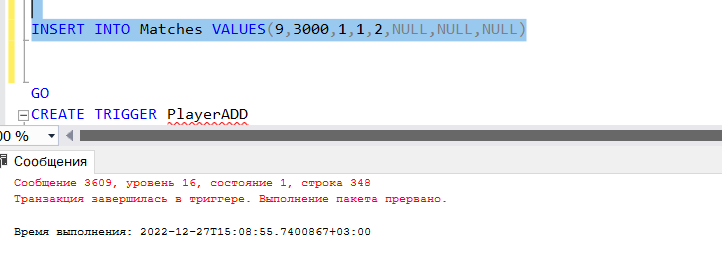
1. Триггер действует на таблицу Players при добавлении, недопуская переполнения команд



1. вместо удаления забирает лицензию у игрока



1. Триггер не дает играть одной команде самой с собой

****

1. **Используемая литература:**

1). Триггеры, виды триггеров и работа с ними - <https://metanit.com/sql/sqlserver/12.1.php>

2). Процедуры - <https://info-comp.ru/programmirovanie/636-stored-procedures-in-t-sql.html>

3). Типы функций - <https://info-comp.ru/function-types-in-t-sql>

4). Представления - <https://info-comp.ru/view-in-t-sql>

5). Классификация номеров в отеле - <http://www.idost.ru/hotels/room_types>