**Практическая работа № 8. Реализация хранимых процедур.**

**Программное обеспечение: Windows 8 и выше, Microsoft Office Word 2016 и выше, Microsoft SQL Server 2017 и выше, Microsoft SQL Server Management Studio 2017.**

**Цель работы:**

**На основании разработанной структуры базы данных, реализовать хранимые процедуры для манипулирования данными в таблицах базы данных.**

**Время на выполнение практической работы:** 4 часа.

**Получение навыков:**

* **В разработке хранимых процедур;**
* **В применении блоков перехватов ошибок;**
* **В применении условных операторов;**
* **В применении циклических операторов;**
* **В применении скалярных, агрегатных и строковых функций;**
* **В применении механизмов вывода сообщений.**

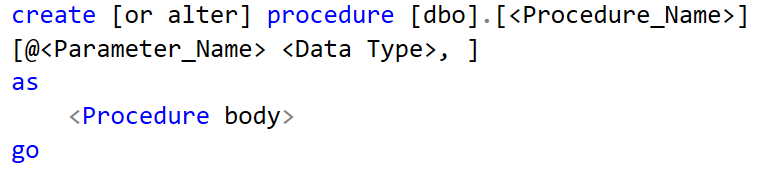
**Теоретические сведения к выполнению практической работы:**

**Хранимая процедура** – вид метаданных, который позволяет выполнять внутри себя ряд запросов на манипулирование данными.

Главные достоинства:

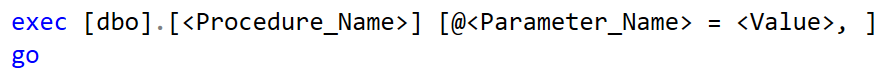
* По сравнению с прямыми запросами на манипулирование данными, хранимые процедуры повышают защиту сущностей и атрибутов, за счёт абстрактного названия объектов и параметров;
* Позволяет один запрос использовать неоднократно;
* Позволяет вложить дополнительные механизмы: валидация данных, автоматическое формирование данных.

Хранимая процедура – конструкции.

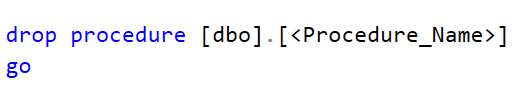


**@<Patameter\_Name>** - входной типизированный параметр в хранимую процедуру, выполняет роль точки входа данных в процедуру. Тип данных должен строго совпадать с типом данных столбца таблицы, на основании которой, разрабатывается хранимая процедура.

Вызов процедуры:



Удаление объекта.

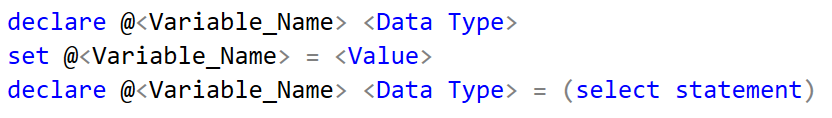


**Переменная** - область памяти, сформированная под конкретный шаблон типа данных.

Объявление:

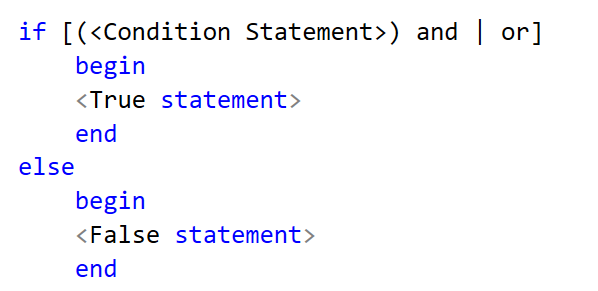
**1 Вариант** – объявление переменной, с типом данных, и заполнение, переменной через команду Set;

**2 Вариант** – объявление переменной с присвоенным значением, через конструкцию Select.



Условный оператор.

* Условия указываются в круглых скобках;
* Логические операторы и (and) или (or);
* Блок Else не обязателен;
* Если необходимо в одном из блоков указать более одной команды указываются операторные скобки Begin и End.

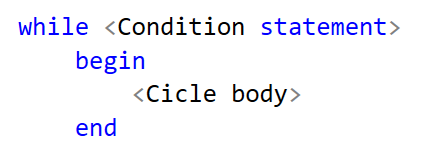


Оператор цикла.

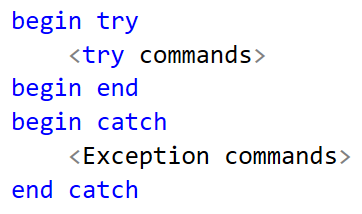
**Цикл** – конструкция позволяет выполнять несколько раз, набор команд.

**While** – цикл с предусловием, выполняется до тех пор пока условие выполнение возвращает истинное значение.

Операторные скобки Begin и End обязательны.



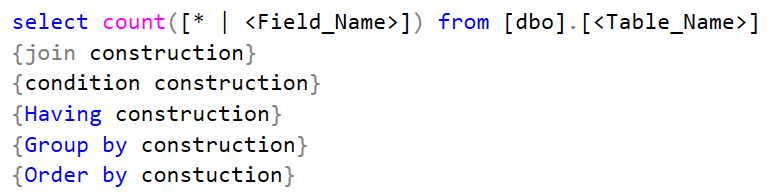
Перехват исключений.



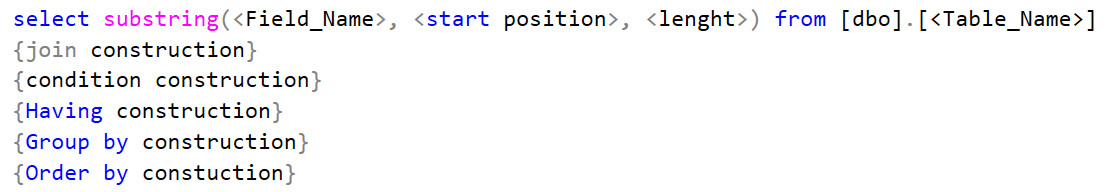
* В блоке try указываются, проверяемые команды;
* Блок Catch перехватывает исключения, следующего типа: ошибки кода, ограничения – check, unique, foreign key.

Дополнительные команды.

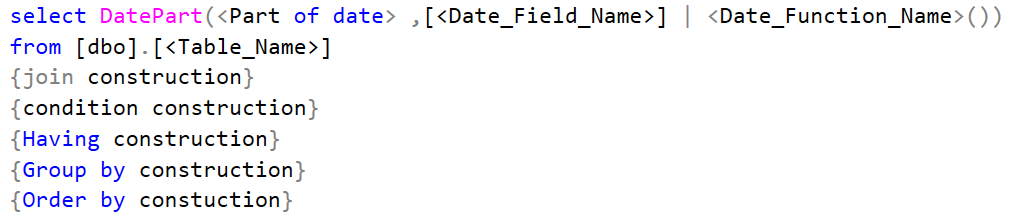
**Count** – скалярная функция возвращает, количество записей в таблице.



**Substring** – агрегатная, строковая функция, возвращает подстроку из строки, где первый формальный параметр - текстовое поле или строковое значение, второй формальный параметр – стартовый индекс строки, третий формальный параметр – длина подстроки.



**DatePart** – агрегатная функция, для работы с датой и временем, возвращает часть даты, где первый формальный параметр – значение части даты, второй формальный параметр – поле даты и времени.



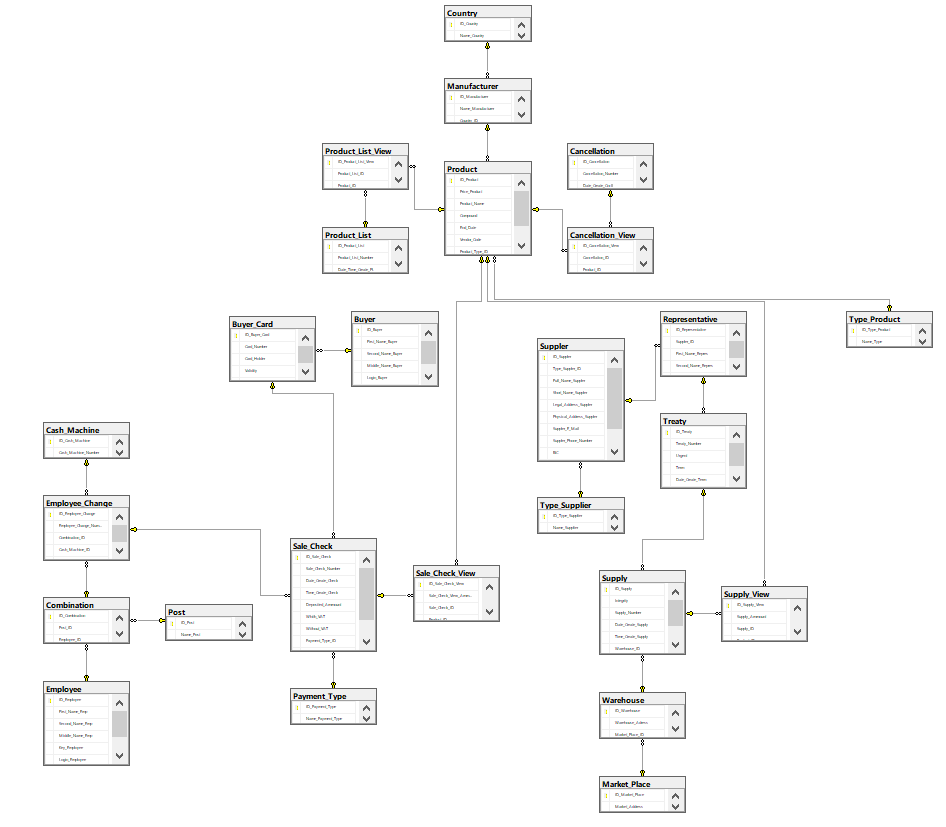
**Print** – функция, возвращает строковое сообщение.



**Этапы выполнения:**

*Для выполнения данной работы, создайте новый файл в Microsoft SQL Server Management Studio, для работы со сценариями хранимых процедур.*

1. Иллюстрация диагнраммы базы данных;



1. К каждой таблице базы данных, необходимо реализовать минимум по 3 – и хранимые процедуры, на добавление, изменение, удаление данных (при этом одна функция может включать в себя сразу несколько действий, к разным таблицам).

|  |  |
| --- | --- |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Type\_Product]  (  [ID\_Type\_Product] [int] not null identity(1,1),  [Name\_Type] [varchar] (50) not null  constraint [PK\_Type\_Product] primary key clustered  ([ID\_Type\_Product] ASC) on [PRIMARY],  constraint [UQ\_Name\_Type] unique ([Name\_Type])  )  go | create or alter procedure [dbo].[Type\_Product\_insert]  @Name\_Type [varchar] (50)  as  begin try  insert into [dbo].[Type\_Product] ([Name\_Type])  values (@Name\_Type)  end try  begin catch  print('Данный вид продукта уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Type\_Product\_update]  @ID\_Type\_Product [int], @Name\_Type [varchar] (50)  as  begin try  update [dbo].[Type\_Product] set  [Name\_Type] = @Name\_Type  where  [ID\_Type\_Product] = @ID\_Type\_Product  end try  begin catch  print('Данный вид продукта уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Type\_Product\_delete]  @ID\_Type\_Product [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Product]  where [Product\_Type\_ID] = @ID\_Type\_Product)  if (@any\_child\_record > 0)  print('Тип продукта не может удалён, так как в таблице "Продукт", есть связанные данные!');  else  delete from [dbo].[Type\_Product]  where  [ID\_Type\_Product] = @ID\_Type\_Product  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Country]  (  [ID\_Country] [int] not null identity(1,1),  [Name\_Country] [varchar] (50) not null  constraint [PK\_Country] primary key clustered  ([ID\_Country] ASC) on [PRIMARY],  constraint [UQ\_Name\_Country] unique ([Name\_Country])  )  go | create or alter procedure [dbo].[Country\_insert]  @Name\_Country [varchar] (50)  as  begin try  insert into [dbo].[Country] ([Name\_Country])  values (@Name\_Country)  end try  begin catch  print('Данная страна уже имеется в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Country\_update]  @ID\_Country [int], @Name\_Country [varchar] (50)  as  begin try  update [dbo].[Country] set  [Name\_Country] = @Name\_Country  where  [ID\_Country] = @ID\_Country  end try  begin catch  print('Данная страна уже имеется в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Country\_delete]  @ID\_Country [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Manufacturer]  where [Country\_ID] = @ID\_Country)  if (@any\_child\_record > 0 )  print('Страна не может быть удалена, так как в таблице "Производитель", есть связанные данные!"')  else  delete from [dbo].[Country]  where  [ID\_Country] = @ID\_Country  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Market\_Place]  (  [ID\_Market\_Place] [int] not null identity(1,1),  [Market\_Address] [varchar] (max) not null  constraint [PK\_Market\_Place] primary key clustered  ([ID\_Market\_Place] ASC) on [PRIMARY]  )  go | create or alter procedure [dbo].[Market\_Place\_insert]  @Market\_Address [varchar] (max)  as  declare @exsist\_record [int] = (select count(\*) from [dbo].[Market\_Place]  where [Market\_Address] = @Market\_Address)  if (@exsist\_record > 0)  print('Данная торговая точка уже есть в таблице!')  else  insert into [dbo].[Market\_Place] ([Market\_Address])  values (@Market\_Address)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Market\_Place\_update]  @ID\_Market\_Place [int], @Market\_Address [varchar] (max)  as  declare @exsist\_record [int] = (select count(\*) from [dbo].[Market\_Place]  where [Market\_Address] = @Market\_Address)  if (@exsist\_record > 0)  print('Данная торговая точка уже есть в таблице!')  else  update [dbo].[Market\_Place] set  [Market\_Address] = @Market\_Address  where  [ID\_Market\_Place] = @ID\_Market\_Place  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Market\_Place\_delete]  @ID\_Market\_Place [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Warehouse]  where [Market\_Place\_ID] = @ID\_Market\_Place)  if (@any\_child\_record > 0 )  print('Торговая точка не может быть удалена, так как в таблице "Склад", есть связанные данные!"')  else  delete from [dbo].[Market\_Place]  where  [ID\_Market\_Place] = @ID\_Market\_Place  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Type\_Supplier]  (  [ID\_Type\_Supplier] [int] not null identity(1,1),  [Name\_Supplier] [varchar] (30) not null  constraint [PK\_Type\_Supplier] primary key clustered  ([ID\_Type\_Supplier] ASC) on [PRIMARY],  constraint [UQ\_Name\_Supplier] unique ([Name\_Supplier])  )  go | create or alter procedure [dbo].[Type\_Supplier\_insert]  @Name\_Supplier [varchar] (30)  as  begin try  insert into [dbo].[Type\_Supplier] ([Name\_Supplier])  values (@Name\_Supplier)  end try  begin catch  print('Данный тип поставщика уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Type\_Supplier\_update]  @ID\_Type\_Supplier [int], @Name\_Supplier [varchar] (30)  as  begin try  update [dbo].[Type\_Supplier] set  [Name\_Supplier] = @Name\_Supplier  where  [ID\_Type\_Supplier] = @ID\_Type\_Supplier  end try  begin catch  print('Данный тип поставщика уже есть в таблице!')  end catch    go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Type\_Supplier\_delete]  @ID\_Type\_Supplier [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Suppler]  where [Type\_Suppler\_ID] = @ID\_Type\_Supplier)  if (@any\_child\_record > 0 )  print('Тип поставщика не может быть удален, так как в таблице "Поставщик", есть связанные данные!"')  else  delete from [dbo].[Type\_Supplier]  where  [ID\_Type\_Supplier] = @ID\_Type\_Supplier  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Payment\_Type]  (  [ID\_Payment\_Type] [int] not null identity(1,1),  [Name\_Payment\_Type] [varchar] (30) not null  constraint [PK\_Payment\_Type] primary key clustered  ([ID\_Payment\_Type] ASC) on [PRIMARY],  constraint [UQ\_Name\_Payment\_Type] unique ([Name\_Payment\_Type])  )  go | create or alter procedure [dbo].[Payment\_Type\_insert]  @Name\_Payment\_Type [varchar] (30)  as  begin try  insert into [dbo].[Payment\_Type] ([Name\_Payment\_Type])  values (@Name\_Payment\_Type)  end try  begin catch  print('Данный вид оплаты уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Payment\_Type\_update]  @ID\_Payment\_Type [int], @Name\_Payment\_Type [varchar] (30)  as  begin try  update [dbo].[Payment\_Type] set  [Name\_Payment\_Type] = @Name\_Payment\_Type  where  [ID\_Payment\_Type] = @ID\_Payment\_Type  end try  begin catch  print('Данный вид оплаты уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Payment\_Type\_delete]  @ID\_Payment\_Type [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Sale\_Check]  where [Payment\_Type\_ID] = @ID\_Payment\_Type)  if (@any\_child\_record > 0 )  print('Тип оплаты не может быть удален, так как в таблице "Товарный чек", есть связанные данные!"')  else  delete from [dbo].[Payment\_Type]  where  [ID\_Payment\_Type] = @ID\_Payment\_Type  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Post]  (  [ID\_Post] [int] not null identity(1,1),  [Name\_Post] [varchar] (50) not null  constraint [PK\_Post] primary key clustered  ([ID\_Post] ASC) on [PRIMARY],  constraint [UQ\_Name\_Post] unique ([Name\_Post])  )  go | create or alter procedure [dbo].[Post\_insert]  @Name\_Post [varchar] (50)  as  begin try  insert into [dbo].[Post] ([Name\_Post])  values (@Name\_Post)  end try  begin catch  print('Данное название должности уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Post\_update]  @ID\_Post [int], @Name\_Post [varchar] (50)  as  begin try  update [dbo].[Post] set  [Name\_Post] = @Name\_Post  where  [ID\_Post] = @ID\_Post  end try  begin catch  print('Данное название должности уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Post\_delete]  @ID\_Post [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Combination]  where [Post\_ID] = @ID\_Post)  if (@any\_child\_record > 0 )  print('Должность не может быть удалена, так как в таблице "Совместительство", есть связанные данные!"')  else  delete from [dbo].[Post]  where  [ID\_Post] = @ID\_Post  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Employee]  (  [ID\_Employee] [int] not null identity(1,1),  [First\_Name\_Emp] [varchar] (30) not null,  [Second\_Name\_Emp] [varchar] (30) not null,  [Middle\_Name\_Emp] [varchar] (30) null default ('-'),  [Key\_Employee] [varchar] (10) not null,  [Login\_Employee] [varchar] (32) not null,  [Password\_Employee] [varchar] (32) not null  constraint [PK\_Employee] primary key clustered  ([ID\_Employee] ASC) on [PRIMARY],  constraint [UQ\_Login\_Employee] unique ([Login\_Employee]),  constraint [CH\_Key\_Employee] check (len([Key\_Employee]) = 10),  constraint [CH\_Key\_Employee\_Letter\_Upper] check ([Key\_Employee] like ('%[A-Z]%')),  constraint [CH\_Key\_Employee\_Letter\_Lower] check ([Key\_Employee] like ('%[a-z]%')),  constraint [CH\_Key\_Employee\_Symbols] check ([Key\_Employee] like ('%[!@#$%^&\*()]%')),  constraint [CH\_Login\_Employee] check (len([Login\_Employee])>=8),  constraint [CH\_Password\_Employee\_Upper] check ([Password\_Employee] like ('%[A-Z]%')),  constraint [CH\_Password\_Employee\_Letter\_Lower] check ([Password\_Employee] like ('%[a-z]%')),  constraint [CH\_Password\_Employee\_Symbols] check ([Password\_Employee] like ('%[!@#$%^&\*()]%'))  )  go  alter table [dbo].[Employee] drop constraint [CH\_Key\_Employee]  go  alter table [dbo].[Employee] drop constraint [CH\_Key\_Employee\_Letter\_Upper]  go  alter table [dbo].[Employee] drop constraint [CH\_Key\_Employee\_Letter\_Lower]  go  alter table [dbo].[Employee] drop constraint [CH\_Key\_Employee\_Symbols]  go  alter table [dbo].[Employee] alter column [Key\_Employee] [varchar] (10) null  go  alter table [dbo].[Employee] add default('-') for [Key\_Employee]  go | create or alter procedure [dbo].[Employee\_insert]  @First\_Name\_Emp [varchar] (30), @Second\_Name\_Emp [varchar] (30), @Middle\_Name\_Emp [varchar] (30),  @Key\_Employee [varchar] (10), @Login\_Employee [varchar] (32), @Password\_Employee [varchar] (32)  as  begin try  declare @exsit\_buyer [int] = (select count(\*) from [dbo].[Buyer] where  [Login\_Buyer] = @Login\_Employee)  if (@exsit\_buyer > 0)  print('Пользователь с указанным логином, уже есть в системе!')  else  begin  declare @exist\_employee [int] = (select count(\*) from [dbo].[Employee]  where [First\_Name\_Emp] = @First\_Name\_Emp and [Second\_Name\_Emp] = @Second\_Name\_Emp and  [Middle\_Name\_Emp] = @Middle\_Name\_Emp and [Login\_Employee] = @Login\_Employee)  if (@exist\_employee > 0)  print('Пользователь с указанными ФИО и логином, уже есть в системе!')  else  begin  declare @New\_key [varchar] (10) = ''  if (@Key\_Employee <> '') or (@Key\_Employee is not null)  set @New\_key = @Key\_Employee  else  set @New\_key = '-'  if (@Middle\_Name\_Emp = null)  insert into [dbo].[Employee] ([First\_Name\_Emp], [Second\_Name\_Emp],[Key\_Employee],  [Login\_Employee], [Password\_Employee])  values (@First\_Name\_Emp, @Second\_Name\_Emp, @New\_key, @Login\_Employee, @Password\_Employee)  else  insert into [dbo].[Employee] ([First\_Name\_Emp], [Second\_Name\_Emp], [Middle\_Name\_Emp], [Key\_Employee],  [Login\_Employee], [Password\_Employee])  values (@First\_Name\_Emp, @Second\_Name\_Emp, @Middle\_Name\_Emp, @Key\_Employee, @Login\_Employee, @Password\_Employee)  end  end  end try  begin catch  print('Пользователь с указанным логином, уже есть в системе!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Employee\_update]  @ID\_Employee [int], @First\_Name\_Emp [varchar] (30), @Second\_Name\_Emp [varchar] (30), @Middle\_Name\_Emp [varchar] (30),  @Key\_Employee [varchar] (10), @Login\_Employee [varchar] (32), @Password\_Employee [varchar] (32)  as  begin try  declare @exsit\_buyer [int] = (select count(\*) from [dbo].[Buyer] where  [Login\_Buyer] = @Login\_Employee)  if (@exsit\_buyer > 0)  print('Пользователь с указанным логином, уже есть в системе!')  else  begin  declare @exist\_employee [int] = (select count(\*) from [dbo].[Employee]  where [First\_Name\_Emp] = @First\_Name\_Emp and [Second\_Name\_Emp] = @Second\_Name\_Emp and  [Middle\_Name\_Emp] = @Middle\_Name\_Emp and [Login\_Employee] = @Login\_Employee)  if (@exist\_employee > 0)  print('Пользователь с указанными ФИО и логином, уже есть в системе!')  else  begin  declare @New\_key [varchar] (10) = ''  if (@Key\_Employee = '') or (@Key\_Employee is null)  set @New\_key = '-'  else  set @New\_key = @Key\_Employee  if (@Middle\_Name\_Emp = null)  update [dbo].[Employee] set  [First\_Name\_Emp] = @First\_Name\_Emp,  [Second\_Name\_Emp] = @Second\_Name\_Emp,  [Key\_Employee] = @New\_key,  [Login\_Employee] = @Login\_Employee,  [Password\_Employee] = @Password\_Employee  where  [ID\_Employee] = @ID\_Employee  else  update [dbo].[Employee] set  [First\_Name\_Emp] = @First\_Name\_Emp,  [Second\_Name\_Emp] = @Second\_Name\_Emp,  [Middle\_Name\_Emp] = @Middle\_Name\_Emp,  [Key\_Employee] = @New\_key,  [Login\_Employee] = @Login\_Employee,  [Password\_Employee] =@Password\_Employee  where  [ID\_Employee] = @ID\_Employee  end  end  end try  begin catch  print('Пользователь с указанным логином, уже есть в системе!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Employee\_delete]  @ID\_Employee [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Combination]  where [Employee\_ID] = @ID\_Employee)  if (@any\_child\_record > 0 )  print('Сотрудник не может быть удален, так как в таблице "Совместительство", есть связанные данные!"')  else  delete from [dbo].[Employee]  where  [ID\_Employee] = @ID\_Employee  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Cash\_Machine]  (  [ID\_Cash\_Machine] [int] not null identity(1,1),  [Cash\_Machine\_Number] [varchar] (11) not null  constraint [PK\_Cash\_Machine] primary key clustered  ([ID\_Cash\_Machine] ASC) on [PRIMARY],  constraint [UQ\_Cash\_Machine\_Number] unique ([Cash\_Machine\_Number]),  constraint [CH\_Cash\_Machine\_Number] check ([Cash\_Machine\_Number] like('КА-[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'))  )  go | create or alter procedure [dbo].[Cash\_Machine\_insert]  @Cash\_Machine\_Number [varchar] (11)  as  begin try  if (@Cash\_Machine\_Number = 'KA-\_\_\_\_\_\_\_\_') or (@Cash\_Machine\_Number = 'KA-') or (@Cash\_Machine\_Number = null)  or (@Cash\_Machine\_Number = ' ')  begin  declare @New\_Number [varchar] (8) = ''  declare @Part\_Number [varchar] (8) = (select convert([varchar](8),count(\*)+1) from [dbo].[Cash\_Machine])  declare @Count\_Value [int] = (select len(@Part\_Number))  declare @step [int] = 8 - @Count\_Value  while @step > 0  begin  set @New\_Number = @New\_Number+'0'  set @step = @step - 1  end  insert into [dbo].[Cash\_Machine] ([Cash\_Machine\_Number])  values ('КА-'+@New\_Number+@Part\_Number)  end  else  insert into [dbo].[Cash\_Machine] ([Cash\_Machine\_Number])  values (@Cash\_Machine\_Number)  end try  begin catch  print('Данный кассовый аппарат уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Cash\_Machine\_update]  @ID\_Cash\_Machine [int], @Cash\_Machine\_Number [varchar] (11)  as  begin try  if (@Cash\_Machine\_Number = 'KA-\_\_\_\_\_\_\_\_') or (@Cash\_Machine\_Number = 'KA-') or (@Cash\_Machine\_Number = null)  or (@Cash\_Machine\_Number = '')  begin  declare @New\_Number [varchar] (8) = ''  declare @Part\_Number [varchar] (8) = (select convert([varchar](8),count(\*)+1) from [dbo].[Cash\_Machine])  declare @Count\_Value [int] = (select len(@Part\_Number))  declare @step [int] = 8 - @Count\_Value  while @step > 0  begin  set @New\_Number = @New\_Number+'0'  set @step = @step - 1  end  update [dbo].[Cash\_Machine] set  [Cash\_Machine\_Number] = 'КА-'+@New\_Number+@Part\_Number  where  [ID\_Cash\_Machine] = @ID\_Cash\_Machine  end  else  update [dbo].[Cash\_Machine] set  [Cash\_Machine\_Number] = @Cash\_Machine\_Number  where  [ID\_Cash\_Machine] = @ID\_Cash\_Machine  end try  begin catch  print('Данный кассовый аппарат уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Cash\_Machine\_delete]  @ID\_Cash\_Machine [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Employee\_Change]  where [Cash\_Machine\_ID] = @ID\_Cash\_Machine)  if (@any\_child\_record > 0 )  print('Кассовый аппарат не может быть удален, так как в таблице "Смена сотрудника", есть связанные данные!"')  else  delete from [dbo].[Cash\_Machine]  where  [ID\_Cash\_Machine] = @ID\_Cash\_Machine  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Buyer]  (  [ID\_Buyer] [int] not null identity(1,1),  [First\_Name\_Buyer] [varchar] (30) not null,  [Second\_Name\_Buyer] [varchar] (30) not null,  [Middle\_Name\_Buyer] [varchar] (30) null default ('-'),  [Login\_Buyer] [varchar] (32) not null,  [Password\_Buyer] [varchar] (32) not null  constraint [PK\_Buyer] primary key clustered  ([ID\_Buyer] ASC) on [PRIMARY],  constraint [UQ\_Login\_Buyer] unique ([Login\_Buyer]),  constraint [CH\_Login\_Buyer] check (len([Login\_Buyer])>=8),  constraint [CH\_Password\_Buyer\_Upper] check ([Password\_Buyer] like ('%[A-Z]%')),  constraint [CH\_Password\_Buyer\_Letter\_Lower] check ([Password\_Buyer] like ('%[a-z]%')),  constraint [CH\_Password\_Buyer\_Symbols] check ([Password\_Buyer] like ('%[!@#$%^&\*()]%'))  )  go | create or alter procedure [dbo].[Buyer\_insert]  @First\_Name\_Buyer [varchar] (30), @Second\_Name\_Buyer [varchar] (30), @Middle\_Name\_Buyer [varchar] (30),  @Login\_Buyer [varchar] (32), @Password\_Buyer [varchar] (32)  as  begin try  declare @excist\_employee [int] = (select count(\*) from [dbo].[Employee]  where [Login\_Employee] = @Login\_Buyer)  if (@excist\_employee > 0)  print('Пользователь с указанным логином, уже есть в системе!')  else  begin  declare @exist\_buyer [int] = (select count(\*) from [dbo].[Buyer]  where [First\_Name\_Buyer] = @First\_Name\_Buyer and [Second\_Name\_Buyer] = @Second\_Name\_Buyer and  [Middle\_Name\_Buyer] = @Middle\_Name\_Buyer and [Login\_Buyer] = @Login\_Buyer)  if (@exist\_buyer > 0)  print('Пользователь с указанными ФИО и логином, уже есть в системе!')  else  if (@Middle\_Name\_Buyer = null)  insert into [dbo].[Buyer] ([First\_Name\_Buyer], [Second\_Name\_Buyer],  [Login\_Buyer], [Password\_Buyer])  values (@First\_Name\_Buyer, @Second\_Name\_Buyer, @Login\_Buyer, @Password\_Buyer)  else  insert into [dbo].[Buyer] ([First\_Name\_Buyer], [Second\_Name\_Buyer], [Middle\_Name\_Buyer],  [Login\_Buyer], [Password\_Buyer])  values (@First\_Name\_Buyer, @Second\_Name\_Buyer, @Middle\_Name\_Buyer, @Login\_Buyer, @Password\_Buyer)  end  end try  begin catch  print('Пользователь с указанным логином, уже есть в системе!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Buyer\_update]  @ID\_Buyer [int], @First\_Name\_Buyer [varchar] (30), @Second\_Name\_Buyer [varchar] (30), @Middle\_Name\_Buyer [varchar] (30),  @Login\_Buyer [varchar] (32), @Password\_Buyer [varchar] (32)  as  begin try  declare @excist\_employee [int] = (select count(\*) from [dbo].[Employee]  where [Login\_Employee] = @Login\_Buyer)  if (@excist\_employee > 0)  print('Пользователь с указанным логином, уже есть в системе!')  else  begin  declare @exist\_buyer [int] = (select count(\*) from [dbo].[Buyer]  where [First\_Name\_Buyer] = @First\_Name\_Buyer and [Second\_Name\_Buyer] = @Second\_Name\_Buyer and  [Middle\_Name\_Buyer] = @Middle\_Name\_Buyer and [Login\_Buyer] = @Login\_Buyer)  if (@exist\_buyer > 0)  print('Пользователь с указанными ФИО и логином, уже есть в системе!')  else  if (@Middle\_Name\_Buyer = null)  update [dbo].[Buyer] set  [First\_Name\_Buyer] = @First\_Name\_Buyer,  [Second\_Name\_Buyer] = @Second\_Name\_Buyer,  [Login\_Buyer] = @Login\_Buyer,  [Password\_Buyer] = @Password\_Buyer  where  [ID\_Buyer] = @ID\_Buyer  else  update [dbo].[Buyer] set  [First\_Name\_Buyer] = @First\_Name\_Buyer,  [Second\_Name\_Buyer] = @Second\_Name\_Buyer,  [Middle\_Name\_Buyer] = @Middle\_Name\_Buyer,  [Login\_Buyer] = @Login\_Buyer,  [Password\_Buyer] = @Password\_Buyer  where  [ID\_Buyer] = @ID\_Buyer  end  end try  begin catch  print('Пользователь с указанным логином, уже есть в системе!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Buyer\_delete]  @ID\_Buyer [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Buyer\_Card]  where [Buyer\_ID] = @ID\_Buyer)  if (@any\_child\_record > 0 )  print('Покупатель не может быть удален, так как в таблице "Банковская карта", есть связанные данные!"')  else  delete from [dbo].[Buyer]  where  [ID\_Buyer] = @ID\_Buyer  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Product\_List]  (  [ID\_Product\_List] [int] not null identity(1,1),  [Product\_List\_Number] [varchar] (13) not null,  [Date\_Time\_Create\_PL] [datetime] null default(getdate())  constraint [PK\_Product\_List] primary key clustered  ([ID\_Product\_List] ASC) on [PRIMARY],  constraint [CH\_Product\_List\_Number] check ([Product\_List\_Number] like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]-[0-9][0-9]')  )  go  alter table [dbo].[Product\_List] add constraint [UQ\_Product\_List\_Number] unique ([Product\_List\_Number])  go | create or alter procedure [dbo].[Product\_List\_insert]  @Product\_List\_Number [varchar] (13)  as  begin try  if (@Product\_List\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_') or (@Product\_List\_Number = null) or (@Product\_List\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select convert([varchar] (10), count(\*)+1) from [dbo].[Product\_List])  declare @new\_count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @new\_count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  insert into [dbo].[Product\_List] ([Product\_List\_Number])  values (@new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2))  end  else  insert into [dbo].[Product\_List] ([Product\_List\_Number])  values (@Product\_List\_Number)  end try  begin catch  print('Данный номер сметы уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Product\_List\_update]  @ID\_Product\_List [int], @Product\_List\_Number [varchar] (13)  as  begin try  if (@Product\_List\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_') or (@Product\_List\_Number = null) or (@Product\_List\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select convert([varchar] (10), count(\*)+1) from [dbo].[Product\_List])  declare @new\_count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @new\_count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  update [dbo].[Product\_List] set  [Product\_List\_Number] = @new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)  where  [ID\_Product\_List] = @ID\_Product\_List  end  else  update [dbo].[Product\_List] set  [Product\_List\_Number] = @Product\_List\_Number  where  [ID\_Product\_List] = @ID\_Product\_List  end try  begin catch  print('Данный номер сметы уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Product\_List\_delete]  @ID\_Product\_List [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Product\_List\_View]  where [Product\_List\_ID] = @ID\_Product\_List)  if (@any\_child\_record > 0 )  print('Смета не может быть удалена, так как в таблице "Состав сметы", есть связанные данные!"')  else  delete from [dbo].[Product\_List]  where  [ID\_Product\_List] = @ID\_Product\_List  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Cancellation]  (  [ID\_Cancellation] [int] not null identity(1,1),  [Cancellation\_Number] [varchar] (16) not null,  [Date\_Create\_Cncll] [date] null default (getdate())  constraint [PK\_Cancellation] primary key clustered  ([ID\_Cancellation] ASC) on [PRIMARY],  constraint [CH\_Cancellation\_Number] check ([Cancellation\_Number] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]-[0-9][0-9]-ОП'))  )  go | create or alter procedure [dbo].[Cancellation\_insert]  @Cancellation\_Number [varchar] (16)  as  begin try  if (@Cancellation\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-\_\_') or (@Cancellation\_Number = null) or (@Cancellation\_Number = '')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select convert([varchar] (10), COUNT(\*)+1) from [dbo].[Cancellation])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  insert into [dbo].[Cancellation] ([Cancellation\_Number])  values (@new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-ОП')  end  else  insert into [dbo].[Cancellation] ([Cancellation\_Number])  values (@Cancellation\_Number)  end try  begin catch  print('Данный номер акта отказа уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Cancellation\_update]  @ID\_Cancellation [int], @Cancellation\_Number [varchar] (16)  as  begin try  if (@Cancellation\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-\_\_') or (@Cancellation\_Number = null) or (@Cancellation\_Number = '')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select convert([varchar] (10), COUNT(\*)+1) from [dbo].[Cancellation])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  update [dbo].[Cancellation] set  [Cancellation\_Number] = @new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-ОП'  where  [ID\_Cancellation] = @ID\_Cancellation  end  else  update [dbo].[Cancellation] set  [Cancellation\_Number] = @Cancellation\_Number  where  [ID\_Cancellation] = @ID\_Cancellation  end try  begin catch  print('Данный номер акта отказа уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Cancellation\_delete]  @ID\_Cancellation [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Cancellation\_View]  where [Cancellation\_ID] = @ID\_Cancellation)  if (@any\_child\_record > 0 )  print('Акт о возврате не может быть удален, так как в таблице "Состав акта возврата", есть связанные данные!"')  else  delete from [dbo].[Cancellation]  where  [ID\_Cancellation] = @ID\_Cancellation  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Manufacturer]  (  [ID\_Manufacturer] [int] not null identity(1,1),  [Name\_Manufacturer] [varchar] (50) not null,  [Country\_ID] [int] not null  constraint [PK\_Manufacturer] primary key clustered  ([ID\_Manufacturer] ASC) on [PRIMARY],  constraint [UQ\_Name\_Manufacturer] unique ([Name\_Manufacturer]),  constraint [FK\_Country\_Manufacturer] foreign key ([Country\_ID])  references [dbo].[Country] ([ID\_Country])  )  go | create or alter procedure [dbo].[Manufacturer\_insert]  @Name\_Manufacturer [varchar] (50), @Country\_ID [int]  as  begin try  declare @exist\_record [int] = (select count(\*) from [dbo].[Manufacturer]  where [Name\_Manufacturer] = @Name\_Manufacturer and [Country\_ID] = @Country\_ID)  if (@exist\_record > 0)  print('Указанный производитель уже есть у данной страны!')  else  insert into [dbo].[Manufacturer] ([Name\_Manufacturer], [Country\_ID])  values (@Name\_Manufacturer, @Country\_ID)  end try  begin catch  print('Указанный производитель уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Manufacturer\_update]  @ID\_Manufacturer [int], @Name\_Manufacturer [varchar] (50), @Country\_ID [int]  as  begin try  declare @exist\_record [int] = (select count(\*) from [dbo].[Manufacturer]  where [Name\_Manufacturer] = @Name\_Manufacturer and [Country\_ID] = @Country\_ID)  if (@exist\_record > 0)  print('Указанный производитель уже есть у данной страны!')  else  update [dbo].[Manufacturer] set  [Name\_Manufacturer] = @Name\_Manufacturer,  [Country\_ID] = @Country\_ID  where  [ID\_Manufacturer] = @ID\_Manufacturer  end try  begin catch  print('Указанный производитель уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Manufacturer\_delete]  @ID\_Manufacturer [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Product]  where [Manufacturer\_ID] = @ID\_Manufacturer)  if (@any\_child\_record > 0 )  print('Производитель не может быть удален, так как в таблице "Продукт", есть связанные данные!"')  else  delete from [dbo].[Manufacturer]  where  [ID\_Manufacturer] = @ID\_Manufacturer  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Warehouse]  (  [ID\_Warehouse] [int] not null identity(1,1),  [Warehouse\_Adress] [varchar] (max) not null,  [Market\_Place\_ID] [int] not null  constraint [PK\_Warehouse] primary key clustered  ([ID\_Warehouse] ASC) on [PRIMARY],  constraint [FK\_Market\_Place\_Warehouse] foreign key ([Market\_Place\_ID])  references [dbo].[Market\_Place] ([ID\_Market\_Place])  )  go | create or alter procedure [dbo].[Warehouse\_insert]  @Warehouse\_Adress [varchar] (max), @Market\_Place\_ID [int]  as  declare @exsist\_record [int] = (select count(\*) from [dbo].[Warehouse]  where [Warehouse\_Adress] = @Warehouse\_Adress and [Market\_Place\_ID] = @Market\_Place\_ID)  if (@exsist\_record > 0)  print('Данный склад у торговой точки уже есть в таблице!')  else  insert into [dbo].[Warehouse] ([Warehouse\_Adress], [Market\_Place\_ID])  values (@Warehouse\_Adress, @Market\_Place\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Warehouse\_update]  @ID\_Warehouse [int], @Warehouse\_Adress [varchar] (max), @Market\_Place\_ID [int]  as  declare @exsist\_record [int] = (select count(\*) from [dbo].[Warehouse]  where [Warehouse\_Adress] = @Warehouse\_Adress and [Market\_Place\_ID] = @Market\_Place\_ID)  if (@exsist\_record > 0)  print('Данный склад у торговой точки уже есть в таблице!')  else  update [dbo].[Warehouse] set  [Warehouse\_Adress] = @Warehouse\_Adress,  [Market\_Place\_ID] = @Market\_Place\_ID  where  [ID\_Warehouse] = @ID\_Warehouse  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Warehouse\_delete]  @ID\_Warehouse [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Supply]  where [Warehouse\_ID] = @ID\_Warehouse)  if (@any\_child\_record > 0 )  print('Склад не может быть удален, так как в таблице "Поставка", есть связанные данные!"')  else  delete from [dbo].[Warehouse]  where  [ID\_Warehouse] = @ID\_Warehouse  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Suppler]  (  [ID\_Suppler] [int] not null identity(1,1),  [Type\_Suppler\_ID] [int] not null,  [Full\_Name\_Suppler] [varchar] (max) not null,  [Short\_Name\_Suppler] [varchar] (50) not null,  [Legal\_Address\_Suppler] [varchar] (max) not null,  [Physical\_Address\_Suppler] [varchar] (max) not null,  [Suppler\_E\_Mail] [varchar] (max) not null,  [Suppler\_Phone\_Number] [varchar] (17) not null,  [BIC] [varchar] (8) not null,  [ITN] [varchar] (12) not null,  [ARCoEaO] [varchar] (10) not null  constraint [PK\_Suppler] primary key clustered  ([ID\_Suppler] ASC) on [PRIMARY],  constraint [UQ\_Suppler\_Phone\_Number] unique ([Suppler\_Phone\_Number]),  constraint [UQ\_ITN] unique ([ITN]),  constraint [UQ\_ARCoEaO] unique ([ARCoEaO]),  constraint [CH\_Suppler\_E\_Mail] check ([Suppler\_E\_Mail] like ('%@%.%')),  constraint [CH\_Suppler\_Phone\_Number] check ([Suppler\_Phone\_Number] like '+7([0-9][0-9][0-9])[0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9]'),  constraint [CH\_BIC] check ([BIC] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_ITN] check ([ITN] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_ARCoEaO] check ([ARCoEaO] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')),  constraint [FK\_Type\_Suppler\_Suppler] foreign key ([Type\_Suppler\_ID])  references [dbo].[Type\_Supplier] ([ID\_Type\_Supplier])  )  go | create or alter procedure [dbo].[Suppler\_insert]  @Type\_Suppler\_ID [int], @Full\_Name\_Suppler [varchar] (max), @Short\_Name\_Suppler [varchar] (50),  @Legal\_Address\_Suppler [varchar] (max), @Physical\_Address\_Suppler [varchar] (max), @Suppler\_E\_Mail [varchar] (max),  @Suppler\_Phone\_Number [varchar] (17), @BIC [varchar] (8), @ITN [varchar] (12), @ARCoEaO [varchar] (10)  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Suppler]  where [Type\_Suppler\_ID] = @Type\_Suppler\_ID and [Full\_Name\_Suppler] = @Full\_Name\_Suppler and [BIC] = @BIC  and [Suppler\_E\_Mail] = @Suppler\_E\_Mail)  if (@exist\_record > 0)  print('Указанный тип организации, название, БИК и E-Mail уже есть в таблице!')  else  insert into [dbo].[Suppler] ([Type\_Suppler\_ID], [Full\_Name\_Suppler], [Short\_Name\_Suppler], [Legal\_Address\_Suppler],  [Physical\_Address\_Suppler], [Suppler\_E\_Mail], [Suppler\_Phone\_Number], [BIC], [ITN], [ARCoEaO])  values (@Type\_Suppler\_ID, @Full\_Name\_Suppler, @Short\_Name\_Suppler, @Legal\_Address\_Suppler, @Physical\_Address\_Suppler,  @Suppler\_E\_Mail, @Suppler\_Phone\_Number, @BIC, @ITN, @ARCoEaO)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Suppler\_update]  @ID\_Suppler [int], @Type\_Suppler\_ID [int], @Full\_Name\_Suppler [varchar] (max), @Short\_Name\_Suppler [varchar] (50),  @Legal\_Address\_Suppler [varchar] (max), @Physical\_Address\_Suppler [varchar] (max), @Suppler\_E\_Mail [varchar] (max),  @Suppler\_Phone\_Number [varchar] (17), @BIC [varchar] (8), @ITN [varchar] (12), @ARCoEaO [varchar] (10)  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Suppler]  where [Type\_Suppler\_ID] = @Type\_Suppler\_ID and [Full\_Name\_Suppler] = @Full\_Name\_Suppler and [BIC] = @BIC  and [Suppler\_E\_Mail] = @Suppler\_E\_Mail)  if (@exist\_record > 0)  print('Указанный тип организации, название, БИК и E-Mail уже есть в таблице!')  else  update [dbo].[Suppler] set  [Type\_Suppler\_ID] = @Type\_Suppler\_ID,  [Full\_Name\_Suppler] = @Full\_Name\_Suppler,  [Short\_Name\_Suppler] = @Short\_Name\_Suppler,  [Legal\_Address\_Suppler] = @Legal\_Address\_Suppler,  [Physical\_Address\_Suppler] = @Physical\_Address\_Suppler,  [Suppler\_E\_Mail] = @Suppler\_E\_Mail,  [Suppler\_Phone\_Number] = @Suppler\_Phone\_Number,  [BIC] = @BIC,  [ITN] = @ITN,  [ARCoEaO] = @ARCoEaO  where  [ID\_Suppler] = @ID\_Suppler  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Suppler\_delete]  @ID\_Suppler [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Representative]  where [Suppler\_ID] = @ID\_Suppler)  if (@any\_child\_record > 0 )  print('Поставщик не может быть удален, так как в таблице "Представитель", есть связанные данные!"')  else  delete from [dbo].[Suppler]  where  [ID\_Suppler] = @ID\_Suppler  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Representative]  (  [ID\_Representative] [int] not null identity(1,1),  [Suppler\_ID] [int] not null,  [First\_Name\_Repres] [varchar] (30) not null,  [Second\_Name\_Repres] [varchar] (30) not null,  [Middle\_Name\_Repres] [varchar] (30) null default('-')  constraint [PK\_Representative] primary key clustered  ([ID\_Representative] ASC) on [PRIMARY],  constraint [FK\_Suppler\_Representative] foreign key ([Suppler\_ID])  references [dbo].[Suppler] ([ID\_Suppler])  )  go | create or alter procedure [dbo].[Representative\_insert]  @Suppler\_ID [int], @First\_Name\_Repres [varchar] (30), @Second\_Name\_Repres [varchar] (30),  @Middle\_Name\_Repres [varchar] (30)  as  if (@Middle\_Name\_Repres = null)  insert into [dbo].[Representative] ([Suppler\_ID], [First\_Name\_Repres], [Second\_Name\_Repres])  values (@Suppler\_ID, @First\_Name\_Repres, @Second\_Name\_Repres)  else  insert into [dbo].[Representative] ([Suppler\_ID], [First\_Name\_Repres], [Second\_Name\_Repres], [Middle\_Name\_Repres])  values (@Suppler\_ID, @First\_Name\_Repres, @Second\_Name\_Repres, @Middle\_Name\_Repres)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Representative\_update]  @ID\_Representative [int], @Suppler\_ID [int], @First\_Name\_Repres [varchar] (30), @Second\_Name\_Repres [varchar] (30),  @Middle\_Name\_Repres [varchar] (30)  as  if (@Middle\_Name\_Repres = null)  update [dbo].[Representative] set  [Suppler\_ID] = @Suppler\_ID,  [First\_Name\_Repres] = @First\_Name\_Repres,  [Second\_Name\_Repres] = @Second\_Name\_Repres  where  [ID\_Representative] = @ID\_Representative  else  update [dbo].[Representative] set  [Suppler\_ID] = @Suppler\_ID,  [First\_Name\_Repres] = @First\_Name\_Repres,  [Second\_Name\_Repres] = @Second\_Name\_Repres,  [Middle\_Name\_Repres] = @Middle\_Name\_Repres  where  [ID\_Representative] = @ID\_Representative  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Representative\_delete]  @ID\_Representative [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Treaty]  where [Representative\_ID] = @ID\_Representative)  if (@any\_child\_record > 0 )  print('Представитель не может быть удален, так как в таблице "Договор", есть связанные данные!"')  else  delete from [dbo].[Representative]  where  [ID\_Representative] = @ID\_Representative  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Treaty]  (  [ID\_Treaty] [int] not null identity(1,1),  [Treaty\_Number] [varchar] (15) not null,  [Urgent] [varchar] (3) not null,  [Term] [varchar] (15) not null,  [Date\_Create\_Term] [date] null default(getdate()),  [Representative\_ID] [int] not null  constraint [PK\_Treaty] primary key clustered  ([ID\_Treaty] ASC) on [PRIMARY],  constraint [UQ\_Treaty\_Number] unique ([Treaty\_Number]),  constraint [CH\_Treaty\_Number] check ([Treaty\_Number] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]-[0-9][0-9]-Д')),  constraint [CH\_Urgent] check ([Urgent] in ('Да','Нет')),  constraint [CH\_Term] check ([Term] like ('[0-9][0-9] лет') or [Term] like ('[0-9][0-9][0-9] месяцев')),  constraint [FK\_Representative\_Treaty] foreign key ([Representative\_ID])  references [dbo].[Representative] ([ID\_Representative])  )  go | create or alter procedure [dbo].[Treaty\_insert]  @Treaty\_Number [varchar] (15), @Urgent [varchar] (3), @Term [varchar] (15), @Representative\_ID [int]  as  begin try  if (@Treaty\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-Д') or (@Treaty\_Number = null) or (@Treaty\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Treaty])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number+'0'  set @step = @step - 1  end  insert into [dbo].[Treaty] ([Treaty\_Number], [Urgent], [Term], [Representative\_ID])  values (@new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-Д', @Urgent, @Term, @Representative\_ID)  end  else  insert into [dbo].[Treaty] ([Treaty\_Number], [Urgent], [Term], [Representative\_ID])  values (@Treaty\_Number, @Urgent, @Term, @Representative\_ID)  end try  begin catch  print('Данный номер договора уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Treaty\_update]  @ID\_Treaty [int], @Treaty\_Number [varchar] (15), @Urgent [varchar] (3), @Term [varchar] (15), @Representative\_ID [int]  as  begin try  if (@Treaty\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-Д') or (@Treaty\_Number = null) or (@Treaty\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Treaty])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number+'0'  set @step = @step - 1  end  update [dbo].[Treaty] set  [Treaty\_Number] = @Treaty\_Number,  [Urgent] = @Urgent,  [Term] = @Term,  [Representative\_ID] =@new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-Д'  where  [ID\_Treaty] = @ID\_Treaty  end  else  update [dbo].[Treaty] set  [Treaty\_Number] = @Treaty\_Number,  [Urgent] = @Urgent,  [Term] = @Term,  [Representative\_ID] =@Representative\_ID  where  [ID\_Treaty] = @ID\_Treaty  end try  begin catch  print('Данный номер договора уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Treaty\_delete]  @ID\_Treaty [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Supply]  where [Treaty\_ID] = @ID\_Treaty)  if (@any\_child\_record > 0 )  print('Договор не может быть удален, так как в таблице "Поставка", есть связанные данные!"')  else  delete from [dbo].[Treaty]  where  [ID\_Treaty] = @ID\_Treaty  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Buyer\_Card]  (  [ID\_Buyer\_Card] [int] not null identity(1,1),  [Card\_Number] [varchar] (19) not null,  [Card\_Holder] [varchar] (60) not null,  [Validity] [varchar] (5) not null,  [Buyer\_ID] [int] not null  constraint [PK\_Buyer\_Card] primary key clustered  ([ID\_Buyer\_Card] ASC) on [PRIMARY],  constraint [UQ\_Card\_Number] unique ([Card\_Number]),  constraint [CH\_Card\_Number] check ([Card\_Number] like '[0-9][0-9][0-9][0-9] [0-9][0-9][0-9][0-9] [0-9][0-9][0-9][0-9] [0-9][0-9][0-9][0-9]'),  constraint [CH\_Validity] check ([Validity] like ('[0-9][0-9]/[0-9][0-9]')),  constraint [FK\_Buyer\_Buyer\_Card] foreign key ([Buyer\_ID])  references [dbo].[Buyer] ([ID\_Buyer])  )  go | create or alter procedure [dbo].[Buyer\_Card\_insert]  @Card\_Number [varchar] (19), @Card\_Holder [varchar] (60), @Validity [varchar] (5),  @Buyer\_ID [int]  as  begin try  declare @exist\_record [int] = (select count(\*) from [dbo].[Buyer\_Card]  where [Card\_Number] = @Card\_Number and [Buyer\_ID] = @Buyer\_ID)  if (@exist\_record > 0)  print('У данного клиента уже есть указанная банковская карта!')  else  insert into [dbo].[Buyer\_Card] ([Card\_Number], [Card\_Holder], [Validity], [Buyer\_ID])  values (@Card\_Number, @Card\_Holder, @Validity, @Buyer\_ID)  end try  begin catch  print('Указанная бакоковская карта уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Buyer\_Card\_update]  @ID\_Buyer\_Card [int], @Card\_Number [varchar] (19), @Card\_Holder [varchar] (60), @Validity [varchar] (5),  @Buyer\_ID [int]  as  begin try  declare @exist\_record [int] = (select count(\*) from [dbo].[Buyer\_Card]  where [Card\_Number] = @Card\_Number and [Buyer\_ID] = @Buyer\_ID)  if (@exist\_record > 0)  print('У данного клиента уже есть указанная банковская карта!')  else  update [dbo].[Buyer\_Card] set  [Card\_Number] = @Card\_Number,  [Card\_Holder] = @Card\_Holder,  [Validity] = @Validity,  [Buyer\_ID] = @Buyer\_ID  where  [ID\_Buyer\_Card] = @ID\_Buyer\_Card  end try  begin catch  print('Указанная бакоковская карта уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Buyer\_Card\_delete]  @ID\_Buyer\_Card [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Sale\_Check]  where [Buyer\_Card\_ID] = @ID\_Buyer\_Card)  if (@any\_child\_record > 0 )  print('Карта клиента не может быть удалена, так как в таблице "Товарный чек", есть связанные данные!"')  else  delete from [dbo].[Buyer\_Card]  where  [ID\_Buyer\_Card] = @ID\_Buyer\_Card  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Combination]  (  [ID\_Combination] [int] not null identity(1,1),  [Post\_ID] [int] not null,  [Employee\_ID] [int] not null  constraint [PK\_Combination] primary key clustered  ([ID\_Combination] ASC) on [PRIMARY],  constraint [FK\_Post\_Combination] foreign key ([Post\_ID])  references [dbo].[Post] ([ID\_Post]),  constraint [FK\_Employee\_Combination] foreign key ([Employee\_ID])  references [dbo].[Employee] ([ID\_Employee])  )  go | create or alter procedure [dbo].[Combination\_insert]  @Post\_ID [int], @Employee\_ID [int]  as  declare @exist\_rectord [int] = (select count(\*) from [dbo].[Combination]  where [Post\_ID] = @Post\_ID and [Employee\_ID] = @Employee\_ID)  if (@exist\_rectord > 0 )  print('Указанная должность уже имеется у сотрудника!')  else  insert into [dbo].[Combination] ([Post\_ID], [Employee\_ID])  values (@Post\_ID, @Employee\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Combination\_update]  @ID\_Combination [int], @Post\_ID [int], @Employee\_ID [int]  as  declare @exist\_rectord [int] = (select count(\*) from [dbo].[Combination]  where [Post\_ID] = @Post\_ID and [Employee\_ID] = @Employee\_ID)  if (@exist\_rectord > 0 )  print('Указанная должность уже имеется у сотрудника!')  else  update [dbo].[Combination] set  [Post\_ID] = @Post\_ID,  [Employee\_ID] = @Employee\_ID  where  [ID\_Combination] = @ID\_Combination  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Combination\_delete]  @ID\_Combination [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Employee\_Change]  where [Combination\_ID] = @ID\_Combination)  if (@any\_child\_record > 0 )  print('Совместительство не может быть удалено, так как в таблице "Смена сотрудника", есть связанные данные!"')  else  delete from [dbo].[Combination]  where  [ID\_Combination] = @ID\_Combination  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Employee\_Change]  (  [ID\_Employee\_Change] [int] not null identity(1,1),  [Employee\_Change\_Number] [varchar] (11) not null,  [Combination\_ID] [int] not null,  [Cash\_Machine\_ID] [int] not null,  [Date\_Time\_Create\_Emp\_Chg] [datetime] null default(getdate())  constraint [PK\_Employee\_Change] primary key clustered  ([ID\_Employee\_Change] ASC) on [PRIMARY],  constraint [FK\_Combination\_Employee\_Change] foreign key ([Combination\_ID])  references [dbo].[Combination] ([ID\_Combination]),  constraint [FK\_Cash\_Machine\_Employee\_Change] foreign key ([Cash\_Machine\_ID])  references [dbo].[Cash\_Machine] ([ID\_Cash\_Machine])  )  go  alter table [dbo].[Employee\_Change] add constraint [CH\_Employee\_Change\_Number]  check ([Employee\_Change\_Number] like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')  go  alter table [dbo].[Employee\_Change] add constraint [UQ\_Employee\_Change\_Number]  unique ([Employee\_Change\_Number])  go | create or alter procedure [dbo].[Employee\_Change\_insert]  @Combination\_ID [int], @Cash\_Machine\_ID [int]  as  begin try  declare @new\_Number [varchar] (11) = ''  declare @Part\_Number [varchar] (11) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Treaty])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 11 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  insert into [dbo].[Employee\_Change] ([Employee\_Change\_Number], [Combination\_ID], [Cash\_Machine\_ID])  values (@new\_Number+@Part\_Number, @Combination\_ID, @Cash\_Machine\_ID)  end try  begin catch  print('Данный номер смены уже имеется в системе')  end catch  go |
| Хранимая процедура на изменение данных |
| Не является рациональным решением, так как данные о смене менять запрещено. |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Employee\_Change\_delete]  @ID\_Employee\_Change [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Sale\_Check]  where [Employee\_Change\_ID] = @ID\_Employee\_Change)  if (@any\_child\_record > 0 )  print('Смена не может быть удалена, так как в таблице "Товарный чек", есть связанные данные!"')  else  delete from [dbo].[Employee\_Change]  where  [ID\_Employee\_Change] = @ID\_Employee\_Change  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Product]  (  [ID\_Product] [int] not null identity(1,1),  [Price\_Product] [decimal] (38,2) null default (0.0),  [Product\_Name] [varchar] (50) not null,  [Compound] [varchar] (max) not null,  [End\_Date] [date] not null,  [Vendor\_Code] [varchar] (13) not null,  [Product\_Type\_ID] [int] not null,  [Manufacturer\_ID] [int] not null  constraint [PK\_Product] primary key clustered  ([ID\_Product] ASC) on [PRIMARY],  constraint [UQ\_Vendor\_Code] unique ([Vendor\_Code]),  constraint [CH\_Vendor\_Code] check ([Vendor\_Code] like '[A-Z][0-9][0-9][0-9][0-9][0-9][0-9]-[0-9][0-9][0-9][0-9][0-9]'),  constraint [CH\_Price\_Product] check ([Price\_Product] >= 0),  constraint [CH\_End\_Date] check ([End\_Date] >= Getdate()),  constraint [FK\_Product\_Type\_Product] foreign key ([Product\_Type\_ID])  references [dbo].[Type\_Product] ([ID\_Type\_Product]),  constraint [FK\_Manufacturer\_Product] foreign key ([Manufacturer\_ID])  references [dbo].[Manufacturer] ([ID\_Manufacturer])  )  go  alter table [dbo].[Product] drop constraint [CH\_End\_Date]  go  alter table [dbo].[Product] alter column [End\_Date] [varchar] (20) not null  go | create or alter procedure [dbo].[Product\_insert]  @Price\_Product [decimal], @Product\_Name [varchar] (50), @Compound [varchar] (max),  @End\_Date [date], @Vendor\_Code [varchar] (13), @Product\_Type\_ID [int], @Manufacturer\_ID [int]  as  begin try  insert into [dbo].[Product] ([Price\_Product], [Product\_Name], [Compound], [End\_Date], [Vendor\_Code],  [Product\_Type\_ID], [Manufacturer\_ID])  values (@Price\_Product, @Product\_Name, @Compound, @End\_Date, @Vendor\_Code, @Product\_Type\_ID, @Manufacturer\_ID)  end try  begin catch  print('Ошибка при вводе данных о товаре!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Product\_update]  @ID\_Product [int], @Price\_Product [decimal], @Product\_Name [varchar] (50), @Compound [varchar] (max),  @End\_Date [date], @Vendor\_Code [varchar] (13), @Product\_Type\_ID [int], @Manufacturer\_ID [int]  as  begin try  update [dbo].[Product] set  [Price\_Product] = @Price\_Product,  [Product\_Name] = @Product\_Name,  [Compound] = @Compound,  [End\_Date] = @End\_Date,  [Vendor\_Code] = @Vendor\_Code,  [Product\_Type\_ID] = @Product\_Type\_ID,  [Manufacturer\_ID] = @Manufacturer\_ID  where  [ID\_Product] = @ID\_Product  end try  begin catch  print('Ошибка при вводе данных о товаре!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Product\_delete]  @ID\_Product [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Product]  inner join [dbo].[Cancellation\_View] on [ID\_Product] = [dbo].[Cancellation\_View].[Product\_ID]  inner join [dbo].[Product\_List\_View] on [ID\_Product] = [dbo].[Product\_List\_View].[Product\_ID]  inner join [dbo].[Sale\_Check\_View] on [ID\_Product] = [dbo].[Sale\_Check\_View].[Product\_ID]  inner join [dbo].[Supply\_View] on [ID\_Product] = [dbo].[Supply\_View].[Product\_ID]  where [dbo].[Cancellation\_View].[Product\_ID] = @ID\_Product or [dbo].[Product\_List\_View].[Product\_ID] = @ID\_Product  or [dbo].[Sale\_Check\_View].[Product\_ID] = @ID\_Product or [dbo].[Supply\_View].[Product\_ID] = @ID\_Product)  if (@any\_child\_record > 0 )  print('Продукт не может быть удален, так как в одной из таблиц: "Состав акта возврата", "Состав сметы", "Состав чека" или "Состав поставки", есть связанные данные!"')  else  delete from [dbo].[Product]  where  [ID\_Product] = @ID\_Product  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Product\_List\_View]  (  [ID\_Product\_List\_View] [int] not null identity(1,1),  [Product\_List\_ID] [int] not null,  [Product\_ID] [int] not null  constraint [PK\_Product\_List\_View] primary key clustered  ([ID\_Product\_List\_View] ASC) on [PRIMARY],  constraint [FK\_Product\_List\_Product\_List\_View] foreign key ([Product\_List\_ID])  references [dbo].[Product\_List] ([ID\_Product\_List]),  constraint [FK\_Product\_Product\_List\_View] foreign key ([Product\_ID])  references [dbo].[Product] ([ID\_Product])  )  go | create or alter procedure [dbo].[Product\_List\_View\_insert]  @Product\_List\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Product\_List\_View]  where [Product\_List\_ID] = @Product\_List\_ID and [Product\_ID] = @Product\_ID)  if (@exist\_record > 0)  print('Указанный товар уже есть в заявке!')  else  insert into [dbo].[Product\_List\_View] ([Product\_List\_ID], [Product\_ID])  values (@Product\_List\_ID, @Product\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Product\_List\_View\_update]  @ID\_Product\_List\_View [int], @Product\_List\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Product\_List\_View]  where [Product\_List\_ID] = @Product\_List\_ID and [Product\_ID] = @Product\_ID)  if (@exist\_record > 0)  print('Указанный товар уже есть в заявке!')  else  update [dbo].[Product\_List\_View] set  [Product\_List\_ID] = @Product\_List\_ID,  [Product\_ID] = @Product\_ID  where  [ID\_Product\_List\_View] = @ID\_Product\_List\_View  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Product\_List\_View\_delete]  @ID\_Product\_List\_View [int]  as  delete from [dbo].[Product\_List\_View]  where  [ID\_Product\_List\_View] = @ID\_Product\_List\_View  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Cancellation\_View]  (  [ID\_Cancellation\_View] [int] not null identity(1,1),  [Cancellation\_ID] [int] not null,  [Product\_ID] [int] not null  constraint [PK\_Cancellation\_View] primary key clustered  ([ID\_Cancellation\_View] ASC) on [PRIMARY],  constraint [FK\_Cancellation\_Cancellation\_View] foreign key ([Cancellation\_ID])  references [dbo].[Cancellation] ([ID\_Cancellation]),  constraint [FK\_Product\_Cancellation\_View] foreign key ([Product\_ID])  references [dbo].[Product] ([ID\_Product])  )  go | create or alter procedure [dbo].[Cancellation\_View\_insert]  @Cancellation\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Cancellation\_View]  where [Cancellation\_ID] = @Cancellation\_ID and [Product\_ID] = @Product\_ID)  if (@exist\_record > 0)  print('В указанном акте уже имеется данный товар!')  else  insert into [dbo].[Cancellation\_View] ([Cancellation\_ID], [Product\_ID])  values (@Cancellation\_ID, @Product\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Cancellation\_View\_update]  @ID\_Cancellation\_View [int], @Cancellation\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Cancellation\_View]  where [Cancellation\_ID] = @Cancellation\_ID and [Product\_ID] = @Product\_ID)  if (@exist\_record > 0)  print('В указанном акте уже имеется данный товар!')  else  update [dbo].[Cancellation\_View] set  [Cancellation\_ID] = @Cancellation\_ID,  [Product\_ID] = @Product\_ID  where  [ID\_Cancellation\_View] = @ID\_Cancellation\_View  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Cancellation\_View\_delete]  @ID\_Cancellation\_View [int]  as  delete from [dbo].[Cancellation\_View]  where  [ID\_Cancellation\_View] = @ID\_Cancellation\_View  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Supply]  (  [ID\_Supply] [int] not null identity(1,1),  [Integrity] [varchar] (30) not null,  [Supply\_Number] [varchar] (15) not null,  [Date\_Create\_Supply] [date] null default (getdate()),  [Time\_Create\_Supply] [time] null default (getdate()),  [Treaty\_ID] [int] not null,  [Warehouse\_ID] [int] not null  constraint [PK\_Supply] primary key clustered  ([ID\_Supply] ASC) on [PRIMARY],  constraint [UQ\_Supply\_Number] unique ([Supply\_Number]),  constraint [CH\_Integrity] check ([Integrity] in ('Нормально','Нарушенное')),  constraint [CH\_Supply\_Number] check ([Supply\_Number] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]-[0-9][0-9]-П')),  constraint [FK\_Warehouse\_Supply] foreign key ([Warehouse\_ID])  references [dbo].[Warehouse] ([ID\_Warehouse]),  constraint [FK\_Treaty\_Supply] foreign key ([Treaty\_ID])  references [dbo].[Treaty] ([ID\_Treaty])  )  go | create or alter procedure [dbo].[Supply\_insert]  @Integrity [varchar] (30), @Supply\_Number [varchar] (15), @Treaty\_ID [int], @Warehouse\_ID [int]  as  begin try  if (@Supply\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-\_') or (@Supply\_Number = null) or (@Supply\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Supply])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number+'0'  set @step = @step - 1  end  insert into [dbo].[Supply] ([Integrity], [Supply\_Number], [Treaty\_ID], [Warehouse\_ID])  values (@Integrity, @new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-П', @Treaty\_ID, @Warehouse\_ID)  end  else  insert into [dbo].[Supply] ([Integrity], [Supply\_Number], [Treaty\_ID], [Warehouse\_ID])  values (@Integrity, @Supply\_Number, @Treaty\_ID, @Warehouse\_ID)  end try  begin catch  print('Указанный номер поставки уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Supply\_update]  @ID\_Supply [int], @Integrity [varchar] (30), @Supply\_Number [varchar] (15), @Treaty\_ID [int], @Warehouse\_ID [int]  as  begin try  if (@Supply\_Number = '\_\_\_\_\_\_\_\_\_\_-\_\_-\_') or (@Supply\_Number = null) or (@Supply\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Supply])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number+'0'  set @step = @step - 1  end  update [dbo].[Supply] set  [Integrity] = @Integrity,  [Supply\_Number] = @new\_Number+@Part\_Number+'-'+SUBSTRING(CONVERT([varchar](4),DATEPART(YEAR,getdate())),3,2)+'-П',  [Treaty\_ID] = @Treaty\_ID,  [Warehouse\_ID] = @Warehouse\_ID  where  [ID\_Supply] = @ID\_Supply  end  else  update [dbo].[Supply] set  [Integrity] = @Integrity,  [Supply\_Number] = @Supply\_Number,  [Treaty\_ID] = @Treaty\_ID,  [Warehouse\_ID] = @Warehouse\_ID  where  [ID\_Supply] = @ID\_Supply  end try  begin catch  print('Указанный номер поставки уже есть в таблице!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Supply\_delete]  @ID\_Supply [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Supply\_View]  where [Supply\_ID] = @ID\_Supply)  if (@any\_child\_record > 0 )  print('Поставка не может быть удалена, так как в таблице "Состав поставки", есть связанные данные!"')  else  delete from [dbo].[Supply]  where  [ID\_Supply] = @ID\_Supply  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Supply\_View]  (  [ID\_Supply\_View] [int] not null identity(1,1),  [Supply\_Ammount] [int] not null,  [Supply\_ID] [int] not null,  [Product\_ID] [int] not null  constraint [PK\_Supply\_View] primary key clustered  ([ID\_Supply\_View] ASC) on [PRIMARY],  constraint [CH\_Supply\_Ammount] check ([Supply\_Ammount] > 0),  constraint [FK\_Supply\_Supply\_View] foreign key ([Supply\_ID])  references [dbo].[Supply] ([ID\_Supply]),  constraint [FK\_Product\_Supply\_View] foreign key ([Product\_ID])  references [dbo].[Product] ([ID\_Product])  )  go | create or alter procedure [dbo].[Supply\_View\_insert]  @Supply\_Ammount [int], @Supply\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Supply\_View]  where [Supply\_ID] = @Supply\_ID and [Product\_ID] = @Product\_ID and [Supply\_Ammount] = @Supply\_Ammount)  if (@exist\_record > 0)  print('Указанный товар уже есть в смете!')  else  insert into [dbo].[Supply\_View] ([Supply\_Ammount], [Supply\_ID], [Product\_ID])  values (@Supply\_Ammount, @Supply\_ID, @Product\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Supply\_View\_update]  @ID\_Supply\_View [int], @Supply\_Ammount [int], @Supply\_ID [int], @Product\_ID [int]  as  declare @exist\_record [int] = (select count(\*) from [dbo].[Supply\_View]  where [Supply\_ID] = @Supply\_ID and [Product\_ID] = @Product\_ID and [Supply\_Ammount] = @Supply\_Ammount)  if (@exist\_record > 0)  print('Указанный товар уже есть в смете!')  else  update [dbo].[Supply\_View] set  [Supply\_Ammount] = @Supply\_Ammount,  [Supply\_ID] = @Supply\_ID,  [Product\_ID] = @Product\_ID  where  [ID\_Supply\_View] = @ID\_Supply\_View  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Supply\_View\_delete]  @ID\_Supply\_View [int]  as  delete from [dbo].[Supply\_View]  where  [ID\_Supply\_View] = @ID\_Supply\_View  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Sale\_Check]  (  [ID\_Sale\_Check] [int] not null identity(1,1),  [Sale\_Check\_Number] [varchar] (13) not null,  [Date\_Create\_Check] [date] null default(getdate()),  [Time\_Create\_Check] [time] null default(getdate()),  [Deposited\_Ammount] [decimal] (38,2) not null,  [Whith\_VAT] [decimal] (38,2) not null,  [Without\_VAT] [decimal] (38,2) not null,  [Payment\_Type\_ID] [int] not null,  [Employee\_Change\_ID] [int] not null,  [Buyer\_Card\_ID] [int] not null  constraint [PK\_Sale\_Check] primary key clustered  ([ID\_Sale\_Check] ASC) on [PRIMARY],  constraint [UQ\_Sale\_Check\_Number] unique ([Sale\_Check\_Number]),  constraint [CH\_Sale\_Check\_Number] check ([Sale\_Check\_Number] like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]-ФК'),  constraint [CH\_Deposited\_Ammount] check ([Deposited\_Ammount] > 0),  constraint [CH\_Whith\_VAT] check ([Whith\_VAT] > 0),  constraint [CH\_Without\_VAT] check ([Without\_VAT] > 0),  constraint [FK\_Payment\_Type\_Sale\_Check] foreign key ([Payment\_Type\_ID])  references [dbo].[Payment\_Type] ([ID\_Payment\_Type]),  constraint [FK\_Employee\_Change\_Sale\_Check] foreign key ([Employee\_Change\_ID])  references [dbo].[Employee\_Change] ([ID\_Employee\_Change]),  constraint [FK\_Buyer\_Card\_Sale\_Check] foreign key ([Buyer\_Card\_ID])  references [dbo].[Buyer\_Card] ([ID\_Buyer\_Card])  )  go | create or alter procedure [dbo].[Sale\_Check\_insert]  @Sale\_Check\_Number [varchar], @Deposited\_Ammount [decimal] (38,2), @Whith\_VAT [decimal] (38,2),  @Without\_VAT [decimal] (38,2), @Payment\_Type\_ID [int],  @Employee\_Change\_ID [int], @Buyer\_Card\_ID [int]  as  begin try  if (@Sale\_Check\_Number = '\_\_\_\_\_\_\_\_\_-\_\_') or (@Sale\_Check\_Number = null) or (@Sale\_Check\_Number = ' ')  begin  declare @new\_Number [varchar] (10) = ''  declare @Part\_Number [varchar] (10) = (select CONVERT([varchar](10),count(\*)+1) from [dbo].[Sale\_Check])  declare @New\_Count [int] = (select len(@Part\_Number))  declare @step [int] = 10 - @New\_Count  while @step > 0  begin  set @new\_Number = @new\_Number + '0'  set @step = @step - 1  end  insert into [dbo].[Sale\_Check] ([Sale\_Check\_Number], [Deposited\_Ammount], [Whith\_VAT], [Without\_VAT], [Payment\_Type\_ID],  [Employee\_Change\_ID], [Buyer\_Card\_ID])  values (@new\_Number+@Part\_Number+'-ФК', @Deposited\_Ammount, @Whith\_VAT, @Without\_VAT, @Payment\_Type\_ID, @Employee\_Change\_ID,  @Buyer\_Card\_ID)  end  else  insert into [dbo].[Sale\_Check] ([Sale\_Check\_Number], [Deposited\_Ammount], [Whith\_VAT], [Without\_VAT], [Payment\_Type\_ID],  [Employee\_Change\_ID], [Buyer\_Card\_ID])  values (@Sale\_Check\_Number, @Deposited\_Ammount, @Whith\_VAT, @Without\_VAT, @Payment\_Type\_ID, @Employee\_Change\_ID,  @Buyer\_Card\_ID)  end try  begin catch  print('Указанный номер чека уже есть в таблице!')  end catch  go |
| Хранимая процедура на изменение данных |
| Не является рациональным решением, так как нельзя изменять данные о существующих чеках. |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Sale\_Check\_delete]  @ID\_Sale\_Check [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Sale\_Check\_View]  where [Sale\_Check\_ID] = @ID\_Sale\_Check)  if (@any\_child\_record > 0 )  print('Товарный чек не может быть удален, так как в таблице "Состав товарного чека", есть связанные данные!"')  else  delete from [dbo].[Sale\_Check]  where  [ID\_Sale\_Check] = @ID\_Sale\_Check  go |
| Скрипт таблицы | Хранимая процедура на добавление и изменение данных данных |
| create table [dbo].[Sale\_Check\_View]  (  [ID\_Sale\_Check\_View] [int] not null identity(1,1),  [Sale\_Check\_View\_Ammount] [int] not null,  [Sale\_Check\_ID] [int] not null,  [Product\_ID] [int] not null  constraint [PK\_Sale\_Check\_View] primary key clustered  ([ID\_Sale\_Check\_View] ASC) on [PRIMARY],  constraint [CH\_Sale\_Check\_View\_Ammount] check ([Sale\_Check\_View\_Ammount] > 0),  constraint [FK\_Sale\_Check\_Sale\_Check\_View] foreign key ([Sale\_Check\_ID])  references [dbo].[Sale\_Check] ([ID\_Sale\_Check]),  constraint [FK\_Product\_Sale\_Check\_View] foreign key ([Product\_ID])  references [dbo].[Product] ([ID\_Product])  )  go | create or alter procedure [dbo].[Sale\_Check\_View\_Work]  @Sale\_Check\_ID [int], @Product\_ID [int]  as  declare @ID\_sale\_check\_view [int] = (select [ID\_Sale\_Check\_View] from [dbo].[Sale\_Check\_View]  where [Sale\_Check\_ID] = @Sale\_Check\_ID and [Product\_ID] = @Product\_ID)  if (@ID\_sale\_check\_view = null)  insert into [dbo].[Sale\_Check\_View] ([Sale\_Check\_View\_Ammount], [Sale\_Check\_ID], [Product\_ID])  values (1, @Sale\_Check\_ID, @Product\_ID)  else  update [dbo].[Sale\_Check\_View] set  [Sale\_Check\_View\_Ammount] = [Sale\_Check\_View\_Ammount] + 1,  [Sale\_Check\_ID] = @Sale\_Check\_ID,  [Product\_ID] = @Product\_ID  where  [ID\_Sale\_Check\_View] = @ID\_sale\_check\_view  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Sale\_Check\_View\_delete]  @ID\_Sale\_Check\_View [int]  as  delete from [dbo].[Sale\_Check\_View]  where  [ID\_Sale\_Check\_View] = @ID\_Sale\_Check\_View  go |

1. Иллюстрация обозревателя объекта

