|  |  |
| --- | --- |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Country] (  [ID\_Country] [int] not null identity(1,1),  [Country\_Name] [varchar] (50) not null  constraint [PK\_Country] primary key clustered  ([ID\_Country] ASC) on [PRIMARY],  constraint [UQ\_Country\_Name] unique ([Country\_Name]) ) go | create or alter procedure [dbo].*[Country\_insert]* @Country\_Name [varchar] (50) as  begin try  insert into [dbo].[Country] ([Country\_Name])  values (@Country\_Name)  end try  begin catch  print('Не удалось вставить название страны')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Country\_update]* @ID\_Country [int], @Country\_Name [varchar] (50) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Country]  where [Country\_Name] = @Country\_Name)  if (@exist\_record > 0)  print('Данная страна уже есть в таблице!')  else  update [dbo].[Country] set  [Country\_Name] = @Country\_Name  where [ID\_Country] = @ID\_Country 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 Manufacturer\_Country\_ID = @ID\_Country)  if (@any\_child\_record > 0)  print('Страна не может быть удалена, т.к. используеся в таблице "Производитель"')  else  delete from [dbo].[Country] where ID\_Country = @ID\_Country go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Manufacturer] (  [ID\_Manufacturer] [int] not null identity(1,1),  [Manufacturer\_Name] [varchar] (50) not null,  [Manufacturer\_Country\_ID] [int] not null  constraint [PK\_Manufacturer] primary key clustered  ([ID\_Manufacturer] ASC) on [PRIMARY],  constraint [UQ\_Manufacturer\_Name] unique ([Manufacturer\_Name]),  constraint [FK\_Country\_Manufacturer] foreign key ([Manufacturer\_Country\_ID])  references [dbo].[Country] ([ID\_Country]) ) go | create or alter procedure [dbo].*[Manufacturer\_insert]* @Manufacturer\_Name [varchar] (50), @Manufacturer\_Country\_ID int as  begin try  insert into [dbo].[Manufacturer] ([Manufacturer\_Name], [Manufacturer\_Country\_ID])  values (@Manufacturer\_Name, @Manufacturer\_Country\_ID)  end try  begin catch  print('Не удалось вставить производителя')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Manufacturer\_update]* @ID\_Manufacturer int, @Manufacturer\_Name [varchar] (50), @Manufacturer\_Country\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Manufacturer]  where [Manufacturer\_Name] = @Manufacturer\_Name or [Manufacturer\_Country\_ID] = @Manufacturer\_Country\_ID)  if (@exist\_record > 0)  print('Производитель с такими данными уже есть в таблице!')  else  update [dbo].[Manufacturer] set  [Manufacturer\_Name] = @Manufacturer\_Name,  [Manufacturer\_Country\_ID] = @Manufacturer\_Country\_ID  where [ID\_Manufacturer] = @ID\_Manufacturer 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].[Goods] where Manufacturer\_ID = @ID\_Manufacturer)  if (@any\_child\_record > 0)  print('Производитель не может быть удален, т.к. используеся в таблице "Товар"')  else  delete from [dbo].[Manufacturer] where ID\_Manufacturer = @ID\_Manufacturer go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Buyer] (  [ID\_Buyer] [int] not null identity(1,1),  [Buyer\_Login] [varchar] (32) not null,  [Buyer\_Password] [varchar] (32) not null,  [Buyer\_Phone\_Number] [varchar] (17) not null,  [Buyer\_Address] [varchar] (max) not null,  constraint [PK\_Buyer] primary key clustered  ([ID\_Buyer] ASC) on [PRIMARY],  constraint [UQ\_Buyer\_Login] unique ([Buyer\_Login]),  constraint [UQ\_Buyer\_Phone\_Number] unique ([Buyer\_Phone\_Number]),  constraint [CH\_Buyer\_Login] check (*len*([Buyer\_Login])>=8),  constraint [CH\_Buyer\_Password\_Upper] check ([Buyer\_Password] like ('%[A-Z]%')),  constraint [CH\_Buyer\_Password\_Lower] check ([Buyer\_Password] like ('%[a-z]%')),  constraint [CH\_Buyer\_Password\_Symbols] check ([Buyer\_Password] like ('%[!@#$%^&\*()]%')),  constraint [CH\_Buyer\_Phone\_Number] check ([Buyer\_Phone\_Number] like '+7([0-9][0-9][0-9])[0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9]')  ) go | create or alter procedure [dbo].*[Buyer\_insert]* @Buyer\_Login varchar(32), @Buyer\_Password varchar(32), @Buyer\_Phone\_Number varchar(17), @Buyer\_Address varchar(max) as  begin try  insert into [dbo].[Buyer] ([Buyer\_Login], [Buyer\_Password], [Buyer\_Phone\_Number], [Buyer\_Address])  values (@Buyer\_Login, @Buyer\_Password, @Buyer\_Phone\_Number, @Buyer\_Address)  end try  begin catch  print('Не удалось вставить покупателя')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Buyer\_update]* @ID\_Buyer int, @Buyer\_Login varchar(32), @Buyer\_Password varchar(32), @Buyer\_Phone\_Number varchar(17), @Buyer\_Address varchar(max) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Buyer]  where [Buyer\_Login] = @Buyer\_Login or [Buyer\_Phone\_Number] = @Buyer\_Phone\_Number)  if (@exist\_record > 0)  print('Покупатель с таким телефоном или логином уже есть в таблице!')  else  update [dbo].[Buyer] set  [Buyer\_Login] = @Buyer\_Login,  [Buyer\_Password] = @Buyer\_Password,  [Buyer\_Phone\_Number] = @Buyer\_Phone\_Number,  [Buyer\_Address] = @Buyer\_Address  where [ID\_Buyer] = @ID\_Buyer 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) + (select *count*(\*)  from [dbo].[Passport] where Buyer\_ID = @ID\_Buyer)  if (@any\_child\_record > 0)  print('Покупатель не может быть удален, т.к. используеся в таблице "Паспорт" или "Карта"')  else  delete from [dbo].[Buyer] where ID\_Buyer = @ID\_Buyer 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,  [Card\_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\_Card\_Holder] check ([Card\_Holder] = *UPPER*([Card\_Holder])),  constraint [CH\_Card\_Validity] check ([Card\_Validity] like ('[0-1][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), @Card\_Validity varchar(5), @Buyer\_ID int as  begin try  insert into [dbo].[Buyer\_Card] ([Card\_Number],[Card\_Holder],[Card\_Validity],[Buyer\_ID])  values (@Card\_Number,@Card\_Holder,@Card\_Validity,@Buyer\_ID)  end try  begin catch  print('Не удалось вставить карту покупателя')  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), @Card\_Validity varchar(5), @Buyer\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Buyer\_Card]  where [Card\_Number] = @Card\_Number)  if (@exist\_record > 0)  print('Карта с данным номером уже есть в таблице!')  else  update [dbo].[Buyer\_Card] set  [Card\_Number] = @Card\_Number,  [Card\_Holder] = @Card\_Holder,  [Card\_Validity] = @Card\_Validity,  [Buyer\_ID] = @Buyer\_ID  where [ID\_Buyer\_Card] = @ID\_Buyer\_Card 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].[Order] where Buyer\_Card\_ID = @ID\_Buyer\_Card)  if (@any\_child\_record > 0)  print('Карта покупателя не может быть удалена, т.к. используеся в таблице "Заказ"')  else  delete from [dbo].[Buyer\_Card] where ID\_Buyer\_Card = @ID\_Buyer\_Card go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Supplier] (  [ID\_Supplier] [int] not null identity(1,1),  [Supplier\_Full\_Name] [varchar] (max) not null,  [Supplier\_Short\_Name] [varchar] (50) not null,  [Supplier\_Legal\_Address] [varchar] (max) not null,  [Supplier\_Physical\_Address] [varchar] (max) not null,  [Supplier\_E\_Mail] [varchar] (max) not null,  [Supplier\_Phone\_Number] [varchar] (17) not null  constraint [PK\_Supplier] primary key clustered  ([ID\_Supplier] ASC) on [PRIMARY],  constraint [UQ\_Supplier\_Short\_Name] unique ([Supplier\_Short\_Name]),  constraint [UQ\_Supplier\_Phone\_Number] unique ([Supplier\_Phone\_Number]),  constraint [CH\_Supplier\_E\_Mail] check ([Supplier\_E\_Mail] like ('%@%.%')),  constraint [CH\_Supplier\_Phone\_Number] check ([Supplier\_Phone\_Number] like '+7([0-9][0-9][0-9])[0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9]') ) go | create or alter procedure [dbo].*[Supplier\_insert]* @Supplier\_Full\_Name varchar(max), @Supplier\_Short\_Name varchar(50), @Supplier\_Legal\_Address varchar(max), @Supplier\_Physical\_Address varchar(max), @Supplier\_E\_Mail varchar(max), @Supplier\_Phone\_Number varchar(17) as  begin try  insert into [dbo].[Supplier] ([Supplier\_Full\_Name], [Supplier\_Short\_Name],  [Supplier\_Legal\_Address], [Supplier\_Physical\_Address], [Supplier\_E\_Mail], [Supplier\_Phone\_Number])  values (@Supplier\_Full\_Name, @Supplier\_Short\_Name, @Supplier\_Legal\_Address,  @Supplier\_Physical\_Address, @Supplier\_E\_Mail, @Supplier\_Phone\_Number)return 1  end try  begin catch  print('Не удалось вставить поставщика')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Supplier\_update]* @ID\_Supplier int, @Supplier\_Full\_Name varchar(max), @Supplier\_Short\_Name varchar(50), @Supplier\_Legal\_Address varchar(max), @Supplier\_Physical\_Address varchar(max), @Supplier\_E\_Mail varchar(max), @Supplier\_Phone\_Number varchar(17) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Supplier]  where [Supplier\_Full\_Name] = @Supplier\_Full\_Name or  [Supplier\_Short\_Name] = @Supplier\_Short\_Name or  [Supplier\_E\_Mail] = @Supplier\_E\_Mail or  [Supplier\_Phone\_Number] = @Supplier\_Phone\_Number)  if (@exist\_record > 0)  print('Поставщик с такими данными уже есть в таблице!')  else  update [dbo].[Supplier] set  [Supplier\_Full\_Name] = @Supplier\_Full\_Name,  [Supplier\_Short\_Name] = @Supplier\_Short\_Name,  [Supplier\_Legal\_Address] = @Supplier\_Legal\_Address,  [Supplier\_Physical\_Address] = @Supplier\_Physical\_Address,  [Supplier\_E\_Mail] = @Supplier\_E\_Mail,  [Supplier\_Phone\_Number] = @Supplier\_Phone\_Number  where [ID\_Supplier] = @ID\_Supplier go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Supplier\_delete]* @ID\_Supplier int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Representative] where Supplier\_ID = @ID\_Supplier)  if (@any\_child\_record > 0)  print('Поставщик не может быть удален, т.к. используеся в таблице "Представитель"')  else  delete from [dbo].[Supplier] where ID\_Supplier = @ID\_Supplier go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Representative] (  [ID\_Representative] [int] not null identity(1,1),  [Supplier\_ID] [int] not null,  [Repres\_First\_Name] [varchar] (30) not null,  [Repres\_Second\_Name] [varchar] (30) not null,  [Repres\_Middle\_Name] [varchar] (30) null default('-')  constraint [PK\_Representative] primary key clustered  ([ID\_Representative] ASC) on [PRIMARY],  constraint [FK\_Supplier\_Representative] foreign key ([Supplier\_ID])  references [dbo].[Supplier] ([ID\_Supplier]) ) go | create or alter procedure [dbo].*[Representative\_insert]* @Supplier\_ID int, @Repres\_First\_Name varchar(30), @Repres\_Second\_Name varchar(30), @Repres\_Middle\_Name varchar(30) as  begin try  insert into [dbo].[Representative] ([Supplier\_ID], [Repres\_First\_Name],  [Repres\_Second\_Name], [Repres\_Middle\_Name])  values (@Supplier\_ID, @Repres\_First\_Name, @Repres\_Second\_Name,  @Repres\_Middle\_Name)  end try  begin catch  print('Не удалось вставить поставщика')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Representative\_update]* @ID\_Representative int, @Supplier\_ID int, @Repres\_First\_Name varchar(30), @Repres\_Second\_Name varchar(30), @Repres\_Middle\_Name varchar(30) as  update [dbo].[Representative] set  [Supplier\_ID] = @Supplier\_ID,  [Repres\_First\_Name] = @Repres\_First\_Name,  [Repres\_Second\_Name] = @Repres\_Second\_Name,  [Repres\_Middle\_Name] = @Repres\_Middle\_Name  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].[Contract] where Representative\_ID = @ID\_Representative)  if (@any\_child\_record > 0)  print('Представитель не может быть удален, т.к. используеся в таблице "Поставщик"')  else  delete from [dbo].[Representative] where ID\_Representative = @ID\_Representative go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Contract] (  [ID\_Contract] [int] not null identity(1,1),  [Representative\_ID] [int] not null,  [Contract\_Expiry\_Date] [date] not null,  [Contract\_Conclusion\_Date] [date] null default(*getdate*())  constraint [PK\_Contract] primary key clustered  ([ID\_Contract] ASC) on [PRIMARY],  constraint [CH\_Contract\_Expiry\_Date] check ([Contract\_Expiry\_Date] >= *CAST*(*getdate*() as date)),  constraint [CH\_Contract\_Conclusion\_Date] check ([Contract\_Conclusion\_Date] <= *CAST*(*getdate*() as date)),  constraint [FK\_Representative\_Contract] foreign key ([Representative\_ID])  references [dbo].[Representative] ([ID\_Representative]) ) go | create or alter procedure [dbo].*[Contract\_insert]* @Representative\_ID int, @Contract\_Expiry\_Date date, @Contract\_Conclusion\_Date date as  begin try  insert into [dbo].[Contract] ([Representative\_ID], [Contract\_Expiry\_Date],  [Contract\_Conclusion\_Date])  values (@Representative\_ID, @Contract\_Expiry\_Date, @Contract\_Conclusion\_Date)  return 1  end try  begin catch  print('Не удалось вставить контракт')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Contract\_update]* @ID\_Contract int, @Representative\_ID int, @Contract\_Expiry\_Date date, @Contract\_Conclusion\_Date date as  update [dbo].[Contract] set  [Representative\_ID] = @Representative\_ID,  [Contract\_Expiry\_Date] = @Contract\_Expiry\_Date,  [Contract\_Conclusion\_Date] = @Contract\_Conclusion\_Date  where [ID\_Contract] = @ID\_Contract go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Contract\_delete]* @ID\_Contract int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Requests\_For\_Supply] where Contract\_ID = @ID\_Contract)  if (@any\_child\_record > 0)  print('Контракт не может быть удален, т.к. используеся в таблице "Заявки на поставки"')  else  delete from [dbo].[Contract] where ID\_Contract = @ID\_Contract go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Characteristics\_Name] (  [ID\_Characteristic\_Name] [int] not null identity(1,1),  [Characteristic\_Name] [varchar] (30) not null  constraint [PK\_Characteristic\_Name] primary key clustered  ([ID\_Characteristic\_Name] ASC) on [PRIMARY],  constraint [UQ\_Characteristic\_Name] unique ([Characteristic\_Name]) ) go | create or alter procedure [dbo].*[Characteristics\_Name\_insert]* @Characteristic\_Name varchar(30) as  begin try  insert into [dbo].[Characteristics\_Name] ([Characteristic\_Name])  values (@Characteristic\_Name)  return 1  end try  begin catch  print('Не удалось вставить имя характеристики')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Characteristics\_Name\_update]* @ID\_Characteristic\_Name int, @Characteristic\_Name varchar(30) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Characteristics\_Name]  where [Characteristic\_Name] = @Characteristic\_Name)  if (@exist\_record > 0)  print('Такое название характеристики уже есть в таблице!')  else  update [dbo].[Characteristics\_Name] set  [Characteristic\_Name] = @Characteristic\_Name  where [ID\_Characteristic\_Name] = @ID\_Characteristic\_Name go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Characteristics\_Name\_delete]* @ID\_Characteristic\_Name int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Characteristics] where Characteristic\_Name\_ID = @ID\_Characteristic\_Name)  if (@any\_child\_record > 0)  print('Название характеристики не может быть удалено, т.к. используеся в таблице "Характеристики"')  else  delete from [dbo].[Characteristics\_Name] where ID\_Characteristic\_Name = @ID\_Characteristic\_Name go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Goods\_Type] (  [ID\_Goods\_Type] [int] not null identity(1,1),  [Goods\_Type\_Name] [varchar] (30) not null  constraint [PK\_Goods\_Type] primary key clustered  ([ID\_Goods\_Type] ASC) on [PRIMARY],  constraint [UQ\_Goods\_Type\_Name] unique ([Goods\_Type\_Name]) ) go | create or alter procedure [dbo].*[Goods\_Type\_insert]* @Goods\_Type\_Name varchar(30) as  begin try  insert into [dbo].[Goods\_Type] ([Goods\_Type\_Name])  values (@Goods\_Type\_Name)  return 1  end try  begin catch  print('Не удалось вставить тип товара')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Goods\_Type\_update]* @ID\_Goods\_Type int, @Goods\_Type\_Name varchar(30) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Goods\_Type]  where [Goods\_Type\_Name] = @Goods\_Type\_Name)  if (@exist\_record > 0)  print('Такое название типа товара уже есть в таблице!')  else  update [dbo].[Goods\_Type] set  [Goods\_Type\_Name] = @Goods\_Type\_Name  where [ID\_Goods\_Type] = @ID\_Goods\_Type go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Goods\_Type\_delete]* @ID\_Goods\_Type int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Goods] where Goods\_Type\_ID = @ID\_Goods\_Type)  if (@any\_child\_record > 0)  print('Типт товара не может быть удален, т.к. используеся в таблице "Товар"')  else  delete from [dbo].[Goods\_Type] where ID\_Goods\_Type = @ID\_Goods\_Type go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Outlets\_Name] (  [ID\_Name] [int] not null identity(1,1),  [Outlet\_Name] [varchar] (50) not null  constraint [PK\_Outlets\_Name] primary key clustered  ([ID\_Name] ASC) on [PRIMARY],  constraint [UQ\_Outlet\_Name] unique ([Outlet\_Name]) ) go | create or alter procedure [dbo].*[Outlets\_Name\_insert]* @Outlet\_Name varchar(50) as  begin try  insert into [dbo].[Outlets\_Name] ([Outlet\_Name])  values (@Outlet\_Name)  return 1  end try  begin catch  print('Не удалось вставить имя торговой точки')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Outlets\_Name\_update]* @ID\_Name int, @Outlet\_Name varchar(50) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Outlets\_Name]  where [Outlet\_Name] = @Outlet\_Name)  if (@exist\_record > 0)  print('Такое название торговой точки уже есть в таблице!')  else  update [dbo].[Outlets\_Name] set  [Outlet\_Name] = @Outlet\_Name  where [ID\_Name] = @ID\_Name go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Outlets\_Name\_delete]* @ID\_Name int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Outlet] where Name\_ID = @ID\_Name)  if (@any\_child\_record > 0)  print('Название торговых точек не может быть удалено, т.к. используеся в таблице "Торговая точка"')  else  delete from [dbo].[Outlets\_Name] where ID\_Name = @ID\_Name go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Outlet] (  [ID\_Outlet] [int] not null identity(1,1),  [Outlet\_Adress] [varchar] (MAX) not null,  [Name\_ID] [int] not null  constraint [PK\_Outlet] primary key clustered  ([ID\_Outlet] ASC) on [PRIMARY],  constraint [FK\_Name\_Outlet] foreign key ([Name\_ID])  references [dbo].[Outlets\_Name] ([ID\_Name]) ) go | create or alter procedure [dbo].*[Outlet\_insert]* @Outlet\_Adress varchar(max), @Name\_ID int as  begin try  insert into [dbo].[Outlet] ([Outlet\_Adress],[Name\_ID])  values (@Outlet\_Adress, @Name\_ID)  return 1  end try  begin catch  print('Не удалось вставить торговую точку')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Outlet\_update]* @ID\_Outlet int, @Outlet\_Adress varchar(max), @Name\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Outlet]  where [Outlet\_Adress] = @Outlet\_Adress)  if (@exist\_record > 0)  print('Торговая точка с таким адресом уже есть в таблице!')  else  update [dbo].[Outlet] set  [Outlet\_Adress] = @Outlet\_Adress,  [Name\_ID] = @Name\_ID  where [ID\_Outlet] = @ID\_Outlet go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Outlet\_delete]* @ID\_Outlet int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Employee] where Outlet\_ID = @ID\_Outlet)  if (@any\_child\_record > 0)  print('Торговая точка не может быть удалена, т.к. используеся в таблице "Сотрудник"')  else  delete from [dbo].[Outlet] where ID\_Outlet = @ID\_Outlet go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Post] (  [ID\_Post] [int] not null identity(1,1),  [Post\_Name] [varchar] (50) not null,  [Superior\_Post\_ID] [int] not null,  [Post\_Salary] [int] not null  constraint [PK\_Post] primary key clustered  ([ID\_Post] ASC) on [PRIMARY],  constraint [UQ\_Post\_Name] unique ([Post\_Name]),  constraint [CH\_Post\_Salary] check ([Post\_Salary] > 0),  constraint [FK\_Superior\_Post] foreign key ([Superior\_Post\_ID])  references [dbo].[Post] ([ID\_Post]) ) go | create or alter procedure [dbo].*[Post\_insert]* @Post\_Name varchar(50), @Post\_Salary int, @Superior\_Post\_ID int as  begin try  set @Superior\_Post\_ID = (select *max*(ID\_Post) from Post) + 1  insert into [dbo].[Post] ([Post\_Name],[Post\_Salary],[Superior\_Post\_ID])  values (@Post\_Name, @Post\_Salary, @Superior\_Post\_ID)  return 1  end try  begin catch  print('Не удалось вставить должность')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Post\_update]* @ID\_Post int, @Post\_Name varchar(50), @Post\_Salary int, @Superior\_Post\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Post]  where [Post\_Name] = @Post\_Name)  if (@exist\_record > 0)  print('Такое название должности уже есть в таблице!')  else  set @Superior\_Post\_ID = (select ID\_Post from Post where ID\_Post = @ID\_Post)  update [dbo].[Post] set  [Post\_Name] = @Post\_Name,  [Post\_Salary] = @Post\_Salary,  [Superior\_Post\_ID] = @Superior\_Post\_ID  where [ID\_Post] = @ID\_Post 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].[Post] where Superior\_Post\_ID = @ID\_Post) + (select *count*(\*)  from [dbo].[Employee] where Post\_ID = @ID\_Post)  if (@any\_child\_record > 0)  print('Должность не может быть удалена, т.к. используеся в таблице "Сотрудник" или является превосходящей')  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),  [Emp\_First\_Name] [varchar] (30) not null,  [Emp\_Second\_Name] [varchar] (30) not null,  [Emp\_Middle\_Name] [varchar] (30) null default ('-'),  [Post\_ID] [int] not null,  [Outlet\_ID] [int] not null,  [Emp\_Login] [varchar] (32) not null,  [Emp\_Password] [varchar] (32) not null  constraint [PK\_Employee] primary key clustered  ([ID\_Employee] ASC) on [PRIMARY],  constraint [UQ\_Emp\_Login] unique ([Emp\_Login]),  constraint [CH\_Emp\_Login] check (*len*([Emp\_Login])>=8),  constraint [CH\_Emp\_Password\_Upper] check ([Emp\_Password] like ('%[A-Z]%')),  constraint [CH\_Emp\_Password\_Lower] check ([Emp\_Password] like ('%[a-z]%')),  constraint [CH\_Emp\_Password\_Symbols] check ([Emp\_Password] like ('%[!@#$%^&\*()]%')),  constraint [FK\_Post\_Emp] foreign key ([Post\_ID])  references [dbo].[Post] ([ID\_Post]),  constraint [FK\_Outlet\_Emp] foreign key ([Outlet\_ID])  references [dbo].[Outlet] ([ID\_Outlet]) ) go | create or alter procedure [dbo].*[Employee\_insert]* @Emp\_First\_Name varchar(30), @Emp\_Second\_Name varchar(30), @Emp\_Middle\_Name varchar(30), @Post\_ID int, @Outlet\_ID int, @Emp\_Login varchar(32), @Emp\_Password varchar(32) as  begin try  insert into [dbo].[Employee] ([Emp\_First\_Name],[Emp\_Second\_Name],  [Emp\_Middle\_Name],[Post\_ID],[Outlet\_ID],[Emp\_Login],[Emp\_Password])  values (@Emp\_First\_Name, @Emp\_Second\_Name, @Emp\_Middle\_Name,  @Post\_ID, @Outlet\_ID, @Emp\_Login, @Emp\_Password)  return 1  end try  begin catch  print('Не удалось вставить сотрудника')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Employee\_update]* @ID\_Employee int, @Emp\_First\_Name varchar(30), @Emp\_Second\_Name varchar(30), @Emp\_Middle\_Name varchar(30), @Post\_ID int, @Outlet\_ID int, @Emp\_Login varchar(32), @Emp\_Password varchar(32) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Employee]  where [Emp\_Login] = @Emp\_Login)  if (@exist\_record > 0)  print('Сортудник с таким логином уже есть в таблице!')  else  update [dbo].[Employee] set  [Emp\_First\_Name] = @Emp\_First\_Name,  [Emp\_Second\_Name] = @Emp\_Second\_Name,  [Emp\_Middle\_Name] = @Emp\_Middle\_Name,  [Post\_ID] = @Post\_ID,  [Outlet\_ID] = @Outlet\_ID,  [Emp\_Login] = @Emp\_Login,  [Emp\_Password] = @Emp\_Password  where [ID\_Employee] = @ID\_Employee 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].[Requests\_For\_Supply] where Employee\_ID = @ID\_Employee) + (select *count*(\*)  from [dbo].[Receipt] where Employee\_ID = @ID\_Employee)  if (@any\_child\_record > 0)  print('Сотрудник не может быть удален, т.к. используеся в таблице "Чек"')  else  delete from [dbo].[Employee] where ID\_Employee = @ID\_Employee go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Passport] (  [ID\_Passport] [int] not null identity(1,1),  [Buyer\_ID] [int] not null,  [Buyer\_First\_Name] [varchar] (30) not null,  [Buyer\_Second\_Name] [varchar] (30) not null,  [Buyer\_Middle\_Name] [varchar] (30) null default ('-'),  [Passport\_Series] [varchar] (5) not null,  [Passport\_Code] [varchar] (6) not null,  [Buyer\_Gender] [varchar] (3) not null,  [Buyer\_Birthday] [date] not null,  [Buyer\_Birthplace] [varchar] (30) not null,  [Passport\_Issued\_By] [varchar] (100) not null  constraint [PK\_Passport] primary key clustered  ([ID\_Passport] ASC) on [PRIMARY],  constraint [FK\_Passport\_Buyer] foreign key ([Buyer\_ID])  references [dbo].[Buyer] ([ID\_Buyer]),  constraint [CH\_Passport\_Series] check ([Passport\_Series] like ('[0-9][0-9] [0-9][0-9]')),  constraint [UQ\_Passport\_Code] unique ([Passport\_Code]),  constraint [CH\_Passport\_Code] check ([Passport\_Code] like '[0-9][0-9][0-9][0-9][0-9][0-9]'),  constraint [CH\_Buyer\_Gender] check ([Buyer\_Gender] in ('МУЖ','ЖЕН')),  constraint [CH\_Buyer\_Birthday] check (*DATEDIFF*(YEAR, [Buyer\_Birthday], *getdate*()) >= 14) ) go | create or alter procedure [dbo].*[Passport\_insert]* @Buyer\_ID int, @Buyer\_First\_Name varchar(30), @Buyer\_Second\_Name varchar(30), @Buyer\_Middle\_Name varchar(30), @Passport\_Series varchar(5), @Passport\_Code varchar(6), @Buyer\_Gender varchar(3), @Buyer\_Birthday date, @Buyer\_Birthplace varchar(30), @Passport\_Issued\_By varchar(100) as  begin try  insert into [dbo].[Passport] ([Buyer\_ID],[Buyer\_First\_Name],  [Buyer\_Second\_Name],[Buyer\_Middle\_Name],[Passport\_Series],  [Passport\_Code],[Buyer\_Gender],[Buyer\_Birthday],  [Buyer\_Birthplace],[Passport\_Issued\_By])  values (@Buyer\_ID, @Buyer\_First\_Name, @Buyer\_Second\_Name,  @Buyer\_Middle\_Name, @Passport\_Series, @Passport\_Code, @Buyer\_Gender,  @Buyer\_Birthday,@Buyer\_Birthplace,@Passport\_Issued\_By)  return 1  end try  begin catch  print('Не удалось вставить паспорт покупателя')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Passport\_update]* @ID\_Passport int, @Buyer\_ID int, @Buyer\_First\_Name varchar(30), @Buyer\_Second\_Name varchar(30), @Buyer\_Middle\_Name varchar(30), @Passport\_Series varchar(5), @Passport\_Code varchar(6), @Buyer\_Gender varchar(3), @Buyer\_Birthday date, @Buyer\_Birthplace varchar(30), @Passport\_Issued\_By varchar(100) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Passport]  where [Passport\_Series] = @Passport\_Series and [Passport\_Code] = @Passport\_Code)  if (@exist\_record > 0)  print('Паспорт с таким номером уже есть в таблице!')  else  update [dbo].[Passport] set  [Buyer\_ID] = @Buyer\_ID,  [Buyer\_First\_Name] = @Buyer\_First\_Name,  [Buyer\_Second\_Name] = @Buyer\_Second\_Name,  [Buyer\_Middle\_Name] = @Buyer\_Middle\_Name,  [Passport\_Series] = @Passport\_Series,  [Passport\_Code] = @Passport\_Code,  [Buyer\_Gender] = @Buyer\_Gender,  [Buyer\_Birthday] = @Buyer\_Birthday,  [Buyer\_Birthplace] = @Buyer\_Birthplace,  [Passport\_Issued\_By] = @Passport\_Issued\_By  where [ID\_Passport] = @ID\_Passport go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Passport\_delete]* @ID\_Passport int as  begin try  delete from [dbo].[Passport] where ID\_Passport = @ID\_Passport  end try  begin catch  print('Не получилось удалить паспорт')  end catch go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Payment\_Type] (  [ID\_Payment\_Type] [int] not null identity(1,1),  [Payment\_Type\_Name] [varchar] (30) not null  constraint [PK\_Payment\_Type] primary key clustered  ([ID\_Payment\_Type] ASC) on [PRIMARY],  constraint [UQ\_Payment\_Type] unique ([Payment\_Type\_Name]) ) go | create or alter procedure [dbo].*[Payment\_Type\_insert]* @Payment\_Type\_Name varchar(30) as  begin try  insert into [dbo].[Payment\_Type] ([Payment\_Type\_Name])  values (@Payment\_Type\_Name)  return 1  end try  begin catch  print('Не удалось вставить тип оплаты')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Payment\_Type\_update]* @ID\_Payment\_Type int, @Payment\_Type\_Name varchar(30) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Payment\_Type]  where [Payment\_Type\_Name] = @Payment\_Type\_Name)  if (@exist\_record > 0)  print('Такое название типа оплаты уже есть в таблице!')  else  update [dbo].[Payment\_Type] set  [Payment\_Type\_Name] = @Payment\_Type\_Name  where [ID\_Payment\_Type] = @ID\_Payment\_Type 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].[Order] where Payment\_Type\_ID = @ID\_Payment\_Type)  if (@any\_child\_record > 0)  print('Ти оплаты не может быть удален, т.к. используеся в таблице "Заказ"')  else  delete from [dbo].[Payment\_Type] where ID\_Payment\_Type = @ID\_Payment\_Type go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Order] (  [ID\_Order] [int] not null identity(1,1),  [Buyer\_Card\_ID] [int] not null,  [Order\_Date] [date] null default *getdate*(),  [Order\_Time] [time] null default *getdate*(),  [Payment\_Type\_ID] [int] not null  constraint [PK\_Order] primary key clustered  ([ID\_Order] ASC) on [PRIMARY],  constraint [FK\_Card\_Order] foreign key ([Buyer\_Card\_ID])  references [dbo].[Buyer\_Card] ([ID\_Buyer\_Card]),  constraint [CH\_Order\_Date] check ([Order\_Date] = *cast*(*getdate*() as date)),  constraint [CH\_Order\_Time] check ([Order\_Time] between *DATEADD*(MINUTE, -1, *cast*(*getdate*() as time)) and *DATEADD*(MINUTE, 1, *cast*(*getdate*() as time))),  constraint [FK\_Payment\_Type\_Order] foreign key ([Payment\_Type\_ID])  references [dbo].[Payment\_Type] ([ID\_Payment\_Type]) ) go | create or alter procedure [dbo].*[Order\_insert]* @Buyer\_Card\_ID int, @Payment\_Type\_ID int as  begin try  insert into [dbo].[Order] ([Buyer\_Card\_ID],[Payment\_Type\_ID])  values (@Buyer\_Card\_ID,@Payment\_Type\_ID)  return 1  end try  begin catch  print('Не удалось вставить заказ')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Order\_update]* @ID\_Order int, @Buyer\_Card\_ID int, @Payment\_Type\_ID int as  update [dbo].[Order] set  [Buyer\_Card\_ID] = @Buyer\_Card\_ID,  [Payment\_Type\_ID] = @Payment\_Type\_ID  where [ID\_Order] = @ID\_Order go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Order\_delete]* @ID\_Order int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Receipt] where Order\_ID = @ID\_Order) + (select *count*(\*)  from [dbo].[Order\_Parts] where Order\_ID = @ID\_Order)  if (@any\_child\_record > 0)  print('Заказ не может быть удален, т.к. используеся в таблице "Чек" или "Часть заказа"')  else  delete from [dbo].[Order] where ID\_Order = @ID\_Order go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Receipt] (  [ID\_Receipt] [int] not null identity(1,1),  [Order\_ID] [int] not null,  [Receipt\_Sum] [int] not null,  [Receipt\_Total] [int] not null,  [Employee\_ID] [int] not null,  [Closure\_Date] [date] null default *getdate*(),  [Closure\_Time] [time] null default *getdate*()  constraint [PK\_Receipt] primary key clustered  ([ID\_Receipt] ASC) on [PRIMARY],  constraint [FK\_Order\_Receipt] foreign key ([Order\_ID])  references [dbo].[Order] ([ID\_Order]),  constraint [CH\_Receipt\_Sum] check ([Receipt\_Sum] > 0),  constraint [CH\_Receipt\_Total] check ([Receipt\_Total] >= 0),  constraint [FK\_Employee\_Receipt] foreign key ([Employee\_ID])  references [dbo].[Employee] ([ID\_Employee]),  constraint [CH\_Closure\_Date] check ([Closure\_Date] = *cast*(*getdate*() as date)),  constraint [CH\_Closure\_Time] check ([Closure\_Time] between *cast*(*getdate*() as time) and *DATEADD*(MINUTE, 1, *cast*(*getdate*() as time))) ) go | create or alter procedure [dbo].*[Receipt\_insert]* @Order\_ID int, @Employee\_ID int as  begin try  declare @Receipt\_Sum [money] = (select *sum*(G.Goods\_Price \* OP.Order\_Goods\_Amount)  from Goods G  join [dbo].[Order\_Parts] OP on OP.Order\_ID = @Order\_ID  where G.ID\_Goods = OP.Goods\_ID)  declare @Receipt\_Total [money] = @Receipt\_Sum \* 1.2  insert into [dbo].[Receipt] ([Order\_ID],[Receipt\_Sum],[Receipt\_Total],[Employee\_ID])  values (@Order\_ID,@Receipt\_Sum,@Receipt\_Total,@Employee\_ID)  return 1  end try  begin catch  print('Не удалось вставить чек')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Receipt\_update]* @ID\_Receipt int, @Order\_ID int, @Receipt\_Sum int, @Receipt\_Total int, @Employee\_ID int as  update [dbo].[Receipt] set  [Order\_ID] = @Order\_ID,  [Receipt\_Sum] = @Receipt\_Sum,  [Receipt\_Total] = @Receipt\_Total,  [Employee\_ID] = @Employee\_ID  where [ID\_Receipt] = @ID\_Receipt go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Receipt\_delete]* @ID\_Receipt int as  begin try  delete from [dbo].[Receipt] where ID\_Receipt = @ID\_Receipt  end try  begin catch  print('Не удалось удалить чек')  end catch go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Requests\_For\_Supply] (  [ID\_Request] [int] not null identity(1,1),  [Request\_Status] [varchar] (10) not null,  [Request\_Date] [date] null default *getdate*(),  [Request\_Time] [time] null default *getdate*(),  [Contract\_ID] [int] not null,  [Employee\_ID] [int] not null  constraint [PK\_Requests\_For\_Supply] primary key clustered  ([ID\_Request] ASC) on [PRIMARY],  constraint [CH\_Request\_Status] check ([Request\_Status] in ('ОТПРАВЛЕНО','ПОЛУЧЕНО','НАРУШЕНО')),  constraint [CH\_Request\_Date] check ([Request\_Date] = *cast*(*getdate*() as date)),  constraint [CH\_Request\_Time] check ([Request\_Time] between *cast*(*getdate*() as time) and *DATEADD*(MINUTE, 1, *cast*(*getdate*() as time))),  constraint [FK\_Contract\_Request] foreign key ([Contract\_ID])  references [dbo].[Contract] ([ID\_Contract]),  constraint [FK\_Employee\_Request] foreign key ([Employee\_ID])  references [dbo].[Employee] ([ID\_Employee]) ) go | create or alter procedure [dbo].*[Requests\_For\_Supply\_insert]* @Request\_Status varchar(10), @Contract\_ID int, @Employee\_ID int as  begin try  insert into [dbo].[Requests\_For\_Supply] ([Request\_Status],[Contract\_ID],[Employee\_ID])  values (@Request\_Status,@Contract\_ID,@Employee\_ID)  return 1  end try  begin catch  print('Не удалось вставить запрос на поставку')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Requests\_For\_Supply\_update]* @ID\_Request int, @Request\_Status varchar(10), @Contract\_ID int, @Employee\_ID int as  update [dbo].[Requests\_For\_Supply] set  [Request\_Status] = @Request\_Status,  [Contract\_ID] = @Contract\_ID,  [Employee\_ID] = @Employee\_ID  where [ID\_Request] = @ID\_Request go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Requests\_For\_Supply\_delete]* @ID\_Request int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Requests\_Parts] where Request\_ID = @ID\_Request)  if (@any\_child\_record > 0)  print('Заявка на поставку не может быть удалена, т.к. используеся в таблице "Часть заявки на поставку"')  else  delete from [dbo].[Requests\_For\_Supply] where ID\_Request = @ID\_Request go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Goods] (  [ID\_Goods] [int] not null identity(1,1),  [Goods\_Name] [varchar] (100) not null,  [Goods\_Article] [varchar] (100) not null,  [Goods\_ID] [int] not null,  [Manufacturer\_ID] [int] not null,  [Goods\_Price] [decimal] (10,2) not null,  [Photo\_Tag] [varchar] (64) not null,  [Goods\_Type\_ID] [int] not null  constraint [PK\_Goods] primary key clustered  ([ID\_Goods] ASC) on [PRIMARY],  constraint [UQ\_Goods\_Article] unique ([Goods\_Article]),  constraint [FK\_Goods\_Goods] foreign key ([Goods\_ID])  references [dbo].[Goods] ([ID\_Goods]),  constraint [FK\_Manufacturer\_Goods] foreign key ([Manufacturer\_ID])  references [dbo].[Manufacturer] ([ID\_Manufacturer]),  constraint [CH\_Goods\_Price] check ([Goods\_Price] > 0),  constraint [UQ\_Photo\_Tag] unique ([Photo\_Tag]),  constraint [FK\_Type\_Goods] foreign key ([Goods\_Type\_ID])  references [dbo].[Goods\_Type] ([ID\_Goods\_Type]) ) go | create or alter procedure [dbo].*[Goods\_insert]* @Goods\_Name varchar(100), @Goods\_Article varchar(100), @Goods\_ID int, @Manufacturer\_ID int, @Goods\_Price decimal(10,2), @Photo\_Tag varchar(64), @Goods\_Type\_ID int as  begin try  insert into [dbo].[Goods] ([Goods\_Name],[Goods\_Article],[Goods\_ID],[Manufacturer\_ID],[Goods\_Price],[Photo\_Tag],[Goods\_Type\_ID])  values (@Goods\_Name,@Goods\_Article,@Goods\_ID,@Manufacturer\_ID,@Goods\_Price,@Photo\_Tag,@Goods\_Type\_ID)  return 1  end try  begin catch  print('Не удалось вставить товар')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Goods\_update]* @ID\_Goods int, @Goods\_Name varchar(100), @Goods\_Article varchar(100), @Goods\_ID int = @ID\_Goods, @Manufacturer\_ID int, @Goods\_Price decimal(10,2), @Photo\_Tag varchar(64), @Goods\_Type\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Goods]  where [Goods\_Article] = @Goods\_Article)  if (@exist\_record > 0)  print('Товар с данным артикулом уже есть в таблице!')  else  update [dbo].[Goods] set  [Goods\_Name] = @Goods\_Name,  [Goods\_Article] = @Goods\_Article,  [Goods\_ID] = @Goods\_ID,  [Manufacturer\_ID] = @Manufacturer\_ID,  [Goods\_Price] = @Goods\_Price,  [Photo\_Tag] = @Photo\_Tag,  [Goods\_Type\_ID] = @Goods\_Type\_ID  where [ID\_Goods] = @ID\_Goods go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Goods\_delete]* @ID\_Goods int as  declare @any\_child\_record [int] = (select *count*(\*)  from [dbo].[Characteristics] where Goods\_ID = @ID\_Goods) + (select *count*(\*)  from [dbo].[Requests\_Parts] where Goods\_ID = @ID\_Goods) + (select *count*(\*)  from [dbo].[Order\_Parts] where Goods\_ID = @ID\_Goods)  if (@any\_child\_record > 0)  print('Товар не может быть удален, т.к. используеся в таблице "Характеристики", "Часть заявки на поставку" или "Часть заказа"')  else  delete from [dbo].[Goods] where ID\_Goods = @ID\_Goods go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Order\_Parts] (  [ID\_Order\_Parts] [int] not null identity(1,1),  [Goods\_ID] [int] not null,  [Order\_Goods\_Amount] [int] not null,  [Order\_ID] [int] not null  constraint [PK\_Order\_Parts] primary key clustered  ([ID\_Order\_Parts] ASC) on [PRIMARY],  constraint [FK\_Order\_Parts\_Goods] foreign key ([Goods\_ID])  references [dbo].[Goods] ([ID\_Goods]),  constraint [CH\_Order\_Goods\_Amount] check ([Order\_Goods\_Amount] > 0),  constraint [FK\_Order\_Parts\_Order] foreign key ([Order\_ID])  references [dbo].[Order] ([ID\_Order]) ) go | create or alter procedure [dbo].*[Order\_Parts\_insert]* @Goods\_ID int, @Order\_Goods\_Amount int, @Order\_ID int as  begin try  insert into [dbo].[Order\_Parts] ([Goods\_ID], [Order\_Goods\_Amount], [Order\_ID])  values (@Goods\_ID, @Order\_Goods\_Amount, @Order\_ID)  return 1  end try  begin catch  print('Не удалось вставить часть заказа')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Order\_Parts\_update]* @ID\_Order\_Parts int, @Goods\_ID int, @Order\_Goods\_Amount int, @Order\_ID int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Order\_Parts]  where [Goods\_ID] = @Goods\_ID and [Order\_ID] = @Order\_ID)  if (@exist\_record > 0)  print('Данный товар уже есть в заказе!')  else  update [dbo].[Order\_Parts] set  [Goods\_ID] = @Goods\_ID,  [Order\_Goods\_Amount] = @Order\_Goods\_Amount,  [Order\_ID] = @Order\_ID  where [ID\_Order\_Parts] = @ID\_Order\_Parts go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Order\_Parts\_delete]* @ID\_Order\_Parts int as  begin try  delete from [dbo].[Order\_Parts] where ID\_Order\_Parts = @ID\_Order\_Parts  end try  begin catch  print('Не удалось удалить часть заказа')  end catch go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Requests\_Parts] (  [ID\_Requests\_Parts] [int] not null identity(1,1),  [Goods\_ID] [int] not null,  [Request\_ID] [int] not null,  [Requests\_Goods\_Amount] [int] not null  constraint [PK\_Requests\_Parts] primary key clustered  ([ID\_Requests\_Parts] ASC) on [PRIMARY],  constraint [FK\_Requests\_Parts\_Goods] foreign key ([Goods\_ID])  references [dbo].[Goods] ([ID\_Goods]),  constraint [FK\_Requests\_Parts\_Request] foreign key ([Request\_ID])  references [dbo].[Requests\_For\_Supply] ([ID\_Request]),  constraint [CH\_Requests\_Goods\_Amount] check ([Requests\_Goods\_Amount] > 0) ) go | create or alter procedure [dbo].*[Requests\_Parts\_insert]* @Goods\_ID int, @Request\_ID int, @Requests\_Goods\_Amount int as  begin try  insert into [dbo].[Requests\_Parts] ([Goods\_ID], [Request\_ID], [Requests\_Goods\_Amount])  values (@Goods\_ID, @Request\_ID, @Requests\_Goods\_Amount)  return 1  end try  begin catch  print('Не удалось вставить часть запроса')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Requests\_Parts\_update]* @ID\_Requests\_Parts int, @Goods\_ID int, @Request\_ID int, @Requests\_Goods\_Amount int as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Requests\_Parts]  where [Goods\_ID] = @Goods\_ID and [Request\_ID] = @Request\_ID)  if (@exist\_record > 0)  print('Данный товар уже есть в заявке!')  else  update [dbo].[Requests\_Parts] set  [Goods\_ID] = @Goods\_ID,  [Request\_ID] = @Request\_ID,  [Requests\_Goods\_Amount] = @Requests\_Goods\_Amount  where [ID\_Requests\_Parts] = @ID\_Requests\_Parts go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Requests\_Parts\_delete]* @ID\_Requests\_Parts int as  begin try  delete from [dbo].[Requests\_Parts] where ID\_Requests\_Parts = @ID\_Requests\_Parts  end try  begin catch  print('Не удалось удалить часть заявки на поставку')  end catch go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Characteristics] (  [ID\_Characteristics] [int] not null identity(1,1),  [Goods\_ID] [int] not null,  [Characteristic\_Name\_ID] [int] not null,  [Characteristics\_Value] [varchar] (50) not null  constraint [PK\_Characteristics] primary key clustered  ([ID\_Characteristics] ASC) on [PRIMARY],  constraint [FK\_Characteristics\_Goods] foreign key ([Goods\_ID])  references [dbo].[Goods] ([ID\_Goods]),  constraint [FK\_Characteristics\_Characteristic\_Name] foreign key ([Characteristic\_Name\_ID])  references [dbo].[Characteristics\_Name] ([ID\_Characteristic\_Name]) ) go | create or alter procedure [dbo].*[Characteristics\_insert]* @Goods\_ID int, @Characteristic\_Name\_ID int, @Characteristics\_Value varchar(50) as  begin try  insert into [dbo].[Characteristics] ([Goods\_ID], [Characteristic\_Name\_ID], [Characteristics\_Value])  values (@Goods\_ID, @Characteristic\_Name\_ID, @Characteristics\_Value)  return 1  end try  begin catch  print('Не удалось вставить характеристику')  end catch go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].*[Characteristics\_update]* @ID\_Characteristics int, @Goods\_ID int, @Characteristic\_Name\_ID int, @Characteristics\_Value varchar(50) as  declare @exist\_record [int] = (select *count*(\*) from [dbo].[Characteristics]  where [Goods\_ID] = @Goods\_ID and [Characteristic\_Name\_ID] = @Characteristic\_Name\_ID)  if (@exist\_record > 0)  print('У товара уже есть данная характеристика!')  else  update [dbo].[Characteristics] set  [Goods\_ID] = @Goods\_ID,  [Characteristic\_Name\_ID] = @Characteristic\_Name\_ID,  [Characteristics\_Value] = @Characteristics\_Value  where [ID\_Characteristics] = @ID\_Characteristics go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].*[Characteristics\_delete]* @ID\_Characteristics int as  begin try  delete from [dbo].[Characteristics] where ID\_Characteristics = @ID\_Characteristics  end try  begin catch  print('Не удалось удалить характеристику')  end catch go |