Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский экономический университет им. Г.В. Плеханова»

Московский приборостроительный техникум

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

Общепрофессиональная дисциплина: ОП 08 Основы проектирования баз данных

МПТ.09.02.07-П.08.ОПБД.П50-5-20.01.22

Отчёт к практической работе № 8

«Реализация хранимых процедур»

Тема: Разработка базы данных на примере предметной области «Организация наземных

грузоперевозок». Реализация подсистем: учёт автомобильного парка, организация доставки грузов.

|  |  |
| --- | --- |
| Проверил: | Выполнил: |
| Щаников И.М.\_\_\_\_\_\_\_\_\_\_ | Студент(ка) группы |
| (\_\_\_\_\_\_\_\_)\_\_\_\_\_\_\_\_\_\_\_\_\_ | П50-5-20 |
| «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022г. | \_\_\_\_\_\_\_\_\_\_\_\_Атаниязов А.Ч. |
|  | «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2022г. |

2022

СОДЕРЖАНИЕ

[1.](#_30j0zll) ЦЕЛИ ПРАКТИЧЕСКОЙ РАБОТЫ 3

[2.](#_1fob9te) ЭТАПЫ ВЫПОЛНЕНИЯ 4

[3.](#_3znysh7) ВЫВОД 5

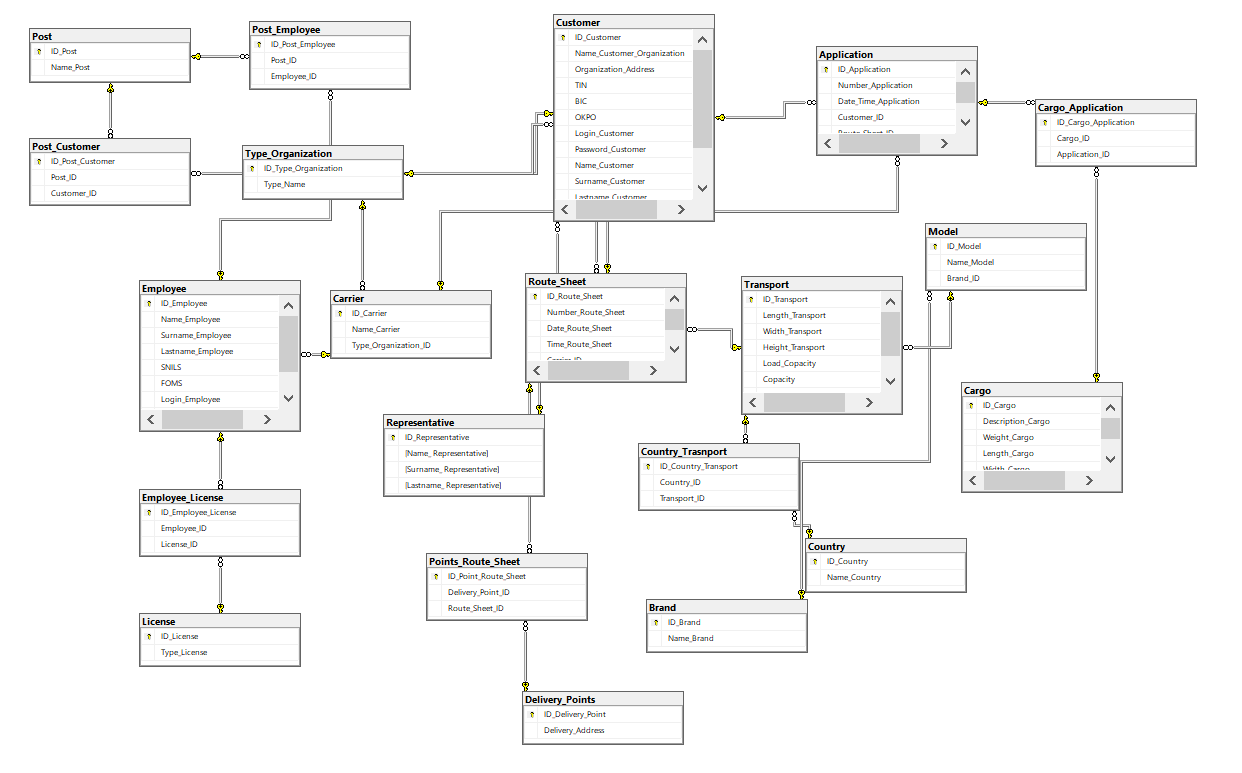
[4.](#_2et92p0) СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ 6

1. ЦЕЛИ ПРАКТИЧЕСКОЙ РАБОТЫ

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

2.ЭТАПЫ ВЫПОЛНЕНИЯ

1. Иллюстрация диаграммы 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].[License]  (  [ID\_License] [int] not null identity(1,1),  [Type\_License] [varchar] (10) not null  constraint [PK\_License] primary key clustered  ([ID\_License] ASC) on [PRIMARY],  constraint [UQ\_Type\_License] unique ([Type\_License])  )  go | create or alter procedure [dbo].[License\_Insert]  @Type\_License [varchar] (10)  as  declare @exsist [int] = (select count(\*) from  [dbo].[License]  where [Type\_License] = @Type\_License)  if (@exsist > 0)  print('Данные о правах уже существует в таблице')  else  insert into [dbo].[License] ([Type\_License])  values (@Type\_License)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[License\_Update]  @ID\_License [int], @Type\_License [varchar] (10)  as  declare @exsist [int] = (select count(\*) from  [dbo].[License]  where [Type\_License] = @Type\_License)  if (@exsist > 0)  print('Данный о правах уже есть в таблице')  else  update [dbo].[License] set  [Type\_License] = @Type\_License  where  [ID\_License] = @ID\_License  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[License\_Delete]  @ID\_License [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[License]  where [ID\_License] = @ID\_License)  if (@any\_child\_record > 0)  print('Права не могут быть удалены так как в таблице  "Права сотрудника" есть связанные данные!')  else  delete from [dbo].[License]  where  [ID\_License] = @ID\_License  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Type\_Organization]  (  [ID\_Type\_Organization] [int] not null identity(1,1),  [Type\_Name] [varchar] (max) not null  constraint [PK\_Type\_Organization] primary key clustered  ([ID\_Type\_Organization] ASC) on [PRIMARY]  )  go | create or alter procedure [dbo].[Type\_Organization\_Insert]  @Type\_Name [varchar] (max)  as  begin try  insert into [Type\_Organization] ([Type\_Name])  values (@Type\_Name)  end try  begin catch  print('Ошибка при вводе типа организации')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Type\_Organization\_Update]  @ID\_Type\_Organization [int], @Type\_Name [varchar] (max)  as  begin try  update [dbo].[Type\_Organization] set  [Type\_Name] = @Type\_Name  where  [ID\_Type\_Organization] = @ID\_Type\_Organization  end try  begin catch  print ('Ошибка при вводе данных в таблицу тип организации!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Type\_Organization\_Delete]  @ID\_Type\_Organization [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Type\_Organization]  where [ID\_Type\_Organization] = @ID\_Type\_Organization)  if (@any\_child\_record > 0)  print('Тип организации не может быть удалены так как в таблице  "Заказчика", "Перевозчика" есть связанные данные!')  else  delete from [dbo].[Type\_Organization]  where  [ID\_Type\_Organization] = @ID\_Type\_Organization  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Brand]  (  [ID\_Brand] [int] not null identity(1,1),  [Name\_Brand] [varchar] (50) not null  constraint [PK\_Brand] primary key clustered  ([ID\_Brand] ASC) on [PRIMARY],  constraint [UQ\_Name\_Brand] unique ([Name\_Brand])  )  go | create or alter procedure [dbo].[Brand\_Insert]  @Name\_Brand [varchar] (50)  as  declare @exsist [int] = (select count(\*) from  [dbo].[Brand]  where [Name\_Brand] = @Name\_Brand)  if (@exsist > 0)  print('Данные о марке уже существует в таблице')  else  insert into [dbo].[Brand] ([Name\_Brand])  values (@Name\_Brand)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Brand\_Update]  @ID\_Brand [int], @Name\_Brand [varchar] (50)  as  declare @exsist [int] = (select count(\*) from  [dbo].[Brand]  where [Name\_Brand] = @Name\_Brand)  if (@exsist > 0)  print('Данный о правах уже есть в таблице')  else  update [dbo].[Brand] set  [Name\_Brand] = @Name\_Brand  where  [ID\_Brand] = @ID\_Brand  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Brand\_Delete]  @ID\_Brand [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Brand]  where [ID\_Brand] = @ID\_Brand)  if (@any\_child\_record > 0)  print('Бренд не может быть удалены так как в таблице  "Модель" есть связанные данные!')  else  delete from [dbo].[Brand]  where  [ID\_Brand] = @ID\_Brand  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Delivery\_Points]  (  [ID\_Delivery\_Point] [int] not null identity(1,1),  [Delivery\_Address] [varchar] (max) not null  constraint [PK\_Delivery\_Point] primary key clustered  ([ID\_Delivery\_Point] ASC) on [PRIMARY]  )  go | create or alter procedure [dbo].[Delivery\_Points\_Insert]  @Delivery\_Address [varchar] (max)  as  begin try  insert into [Delivery\_Points] ([Delivery\_Address])  values (@Delivery\_Address)  end try  begin catch  print('Ошибка при вводе точек доставки')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Delivery\_Points\_Update]  @ID\_Delivery\_Points [int], @Delivery\_Address [varchar] (max)  as  begin try  update [dbo].[Delivery\_Points] set  [Delivery\_Address] = @Delivery\_Address  where  [ID\_Delivery\_Point] = @ID\_Delivery\_Points  end try  begin catch  print ('Ошибка при вводе данных в таблицу точки доставки!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Delivery\_Points\_Delete]  @ID\_Delivery\_Points [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Delivery\_Points]  where [ID\_Delivery\_Point] = @ID\_Delivery\_Points)  if (@any\_child\_record > 0)  print('Точки доставки не может быть удалены так как в таблице  "Точки маршрутного листа" есть связанные данные!')  else  delete from [dbo].[Delivery\_Points]  where  [ID\_Delivery\_Point] = @ID\_Delivery\_Points  go  insert into [dbo].[Delivery\_Points] ([Delivery\_Address])  values ('Ул. Вернадского, д. 3, стр. 4'), ('Ул. Витебская, д. 8, к. 5'), ('Ул. Вернадского, д. 6, к. 3'), ('Ул. Арбат, д. 8, к. 3'), ('Ул. Арбат, д. 1, стр. 3')  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  declare @exsist [int] = (select count(\*) from  [dbo].[Post]  where [Name\_Post] = @Name\_Post)  if (@exsist > 0)  print('Данные о должностях уже существует в таблице')  else  insert into [dbo].[Post] ([Name\_Post])  values (@Name\_Post)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Post\_Update]  @ID\_Post [int], @Name\_Post [varchar] (10)  as  declare @exsist [int] = (select count(\*) from  [dbo].[Post]  where [Name\_Post] = @Name\_Post)  if (@exsist > 0)  print('Данный о должностях уже есть в таблице')  else  update [dbo].[Post] set  [Name\_Post] = @Name\_Post  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 [ID\_Post] = @ID\_Post)  if (@any\_child\_record > 0)  print('Права не могут быть удалены так как в таблице  "Права сотрудника" есть связанные данные!')  else  delete from [dbo].[Post]  where  [ID\_Post] = @ID\_Post  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  declare @exsist [int] = (select count(\*) from [dbo].[Country] where [Name\_Country] = @Name\_Country)  if (@exsist > 0)  begin  print('Данные о стране уже существует в таблице')  end  else  begin  insert into [dbo].[Country] ([Name\_Country])  values (@Name\_Country)  end  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Country\_Update]  @ID\_Country [int], @Name\_Country [varchar] (50)  as  declare @exsist [int] = (select count(\*) from [dbo].[Country] where [Name\_Country] = @Name\_Country)  if (@exsist > 0)  begin  print('Данный о стране уже есть в таблице')  end  else  begin  update [dbo].[Country] set  [Name\_Country] = @Name\_Country  where  [ID\_Country] = @ID\_Country  end  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].[Country]  where [ID\_Country] = @ID\_Country)  if (@any\_child\_record > 0)  begin  print('Страна не могут быть удалены так как в таблице "Страна транспорта" есть связанные данные!')  end  else  begin  delete from [dbo].[Country] where [ID\_Country] = @ID\_Country  end  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Cargo]  (  [ID\_Cargo] [int] not null identity(1,1),  [Description\_Cargo] [varchar] (max) not null,  [Weight\_Cargo] [int] not null,  [Length\_Cargo] [varchar] (8) not null,  [Width\_Cargo] [varchar] (8) not null,  [Height\_Cargo] [varchar] (8) not null  constraint [PK\_Cargo] primary key clustered  ([ID\_Cargo] ASC) on [PRIMARY],  constraint [CH\_Weight\_Cargo] check ([Weight\_Cargo] like ('[0-9][0-9][0-9]')  or [Weight\_Cargo] like ('[0-9][0-9][0-9][0-9]') or [Weight\_Cargo] like ('[0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_Length\_Cargo] check ([Length\_Cargo] like ('[0-9][0-9][0-9]')  or [Length\_Cargo] like ('[0-9][0-9][0-9][0-9]') or [Length\_Cargo] like ('[0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_Width\_Cargo] check ([Width\_Cargo] like ('[0-9][0-9][0-9]')  or [Width\_Cargo] like ('[0-9][0-9][0-9][0-9]') or [Width\_Cargo] like ('[0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_Height\_Cargo] check ([Height\_Cargo] like ('[0-9][0-9][0-9]')  or [Height\_Cargo] like ('[0-9][0-9][0-9][0-9]') or [Height\_Cargo] like ('[0-9][0-9][0-9][0-9][0-9]'))  )  go | create or alter procedure [dbo].[Cargo\_Insert]  @Description\_Cargo [varchar] (max), @Weight\_Cargo [varchar] (8), @Length\_Cargo [varchar] (8), @Width\_Cargo [varchar] (8), @Height\_Cargo [varchar] (8)  as  begin try  insert into [Cargo] ([Description\_Cargo], [Weight\_Cargo], [Length\_Cargo], [Width\_Cargo], [Height\_Cargo])  values (@Description\_Cargo, @Weight\_Cargo, @Length\_Cargo, @Width\_Cargo, @Height\_Cargo)  end try  begin catch  print('Ошибка при вводе данных в таблицу груз')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Cargo\_Update]  @ID\_Cargo [int], @Description\_Cargo [varchar] (max), @Weight\_Cargo [varchar] (8), @Length\_Cargo [varchar] (8), @Width\_Cargo [varchar] (8), @Height\_Cargo [varchar] (8)  as  begin try  update [dbo].[Cargo] set  [Description\_Cargo] = @Description\_Cargo,  [Weight\_Cargo] = @Weight\_Cargo,  [Length\_Cargo] = @Length\_Cargo,  [Width\_Cargo] = @Width\_Cargo,  [Height\_Cargo] = @Height\_Cargo  where  [ID\_Cargo] = @ID\_Cargo  end try  begin catch  print ('Ошибка при вводе данных в таблицу Груз!')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Cargo\_Delete]  @ID\_Cargo [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Cargo]  where [ID\_Cargo] = @ID\_Cargo)  if (@any\_child\_record > 0)  begin  print('Груз не может быть удален так как в таблице  "Заявка Груза" есть связанные данные!')  end  else  begin  delete from [dbo].[Cargo]  where  [ID\_Cargo] = @ID\_Cargo  end  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Model]  (  [ID\_Model] [int] not null identity(1,1),  [Name\_Model] [varchar] (50) not null,  [Brand\_ID] [int] not null  constraint [PK\_Model] primary key clustered  ([ID\_Model] ASC) on [PRIMARY],  constraint [UQ\_Name\_Model] unique ([Name\_Model]),  constraint [FK\_Brand\_Model] foreign key ([Brand\_ID])  references [dbo].[Brand] ([ID\_Brand])  )  go | create or alter procedure [dbo].[Model\_Insert]  @Name\_Model [varchar] (50), @Brand\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Model]  where [Name\_Model] = @Name\_Model and  [Brand\_ID] = @Brand\_ID)  if (@exsist > 0)  print('Данные о модели уже существует в таблице')  else  insert into [dbo].[Model] ([Name\_Model], [Brand\_ID])  values (@Name\_Model, @Brand\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Model\_Update]  @ID\_Model [int], @Name\_Model [varchar] (50), @Brand\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Model]  where [Brand\_ID] = @Brand\_ID and  [Name\_Model] = @Name\_Model)  if (@exsist > 0)  print('Данный о модели уже есть в таблице')  else  update [dbo].[Model] set  [Name\_Model] = @Name\_Model,  [Brand\_ID] = @Brand\_ID  where  [ID\_Model] = @ID\_Model  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Model\_Delete]  @ID\_Model [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Model]  where [ID\_Model] = @ID\_Model)  if (@any\_child\_record > 0)  print('Модели не могут быть удалены так как в таблице  "Транспорт" есть связанные данные!')  else  delete from [dbo].[Model]  where  [ID\_Model] = @ID\_Model  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Transport]  (  [ID\_Transport] [int] not null identity(1,1),  [Length\_Transport] [varchar] (8) not null,  [Width\_Transport] [varchar] (8) not null,  [Height\_Transport] [varchar] (8) not null,  [Load\_Copacity] [varchar] (8) not null,  [Copacity] [varchar] (9) not null,  [Number\_Transport] [varchar] (10) not null,  [Model\_ID] [int] not null  constraint [PK\_Transport] primary key clustered  ([ID\_Transport] ASC) on [PRIMARY],  constraint [CH\_Length\_Transport] check ([Length\_Transport] like ('[0-9][0-9][0-9][0-9]')),  constraint [CH\_Width\_Transport] check ([Width\_Transport] like ('[0-9][0-9][0-9][0-9]')),  constraint [CH\_Height\_Transport] check ([Height\_Transport] like ('[0-9][0-9][0-9][0-9]')),  constraint [CH\_Load\_Copacity] check ([Load\_Copacity] like ('[0-9][0-9][0-9][0-9]')  or [Load\_Copacity] like ('[0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_Copacity] check ([Copacity] like ('[0-9][0-9]')),  constraint [CH\_Number\_Transport] check ([Number\_Transport] like ('[А-Я][0-9][0-9][0-9][А-Я][А-Я]')),  constraint [UQ\_Number\_Transport] unique ([Number\_Transport]),  constraint [FK\_Model\_Transport] foreign key ([Model\_ID])  references [dbo].[Model] ([ID\_Model])  )  go | create or alter procedure [dbo].[Transport\_Insert]  @Length\_Transport [varchar] (8), @Width\_Transport [varchar] (8), @Height\_Transport [varchar] (8), @Load\_Copacity [varchar] (8), @Copacity [varchar] (9), @Number\_Transport [varchar] (10), @Model\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Transport]  where [Length\_Transport] = @Length\_Transport and  [Width\_Transport] = @Width\_Transport and  [Height\_Transport] = @Height\_Transport and  [Load\_Copacity] = @Load\_Copacity and  [Copacity] = @Copacity and  [Number\_Transport] = @Number\_Transport and  [Model\_ID] = @Model\_ID)  if (@exsist > 0)  print('Данные о транспорте уже существует в таблице')  else  insert into [dbo].[Transport] ([Length\_Transport], [Width\_Transport], [Height\_Transport], [Load\_Copacity], [Copacity], [Number\_Transport], [Model\_ID])  values (@Length\_Transport, @Width\_Transport, @Height\_Transport, @Load\_Copacity, @Copacity, @Number\_Transport, @Model\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Transport\_Update]  @ID\_Transport [int], @Length\_Transport [varchar] (8), @Width\_Transport [varchar] (8), @Height\_Transport [varchar] (8), @Load\_Copacity [varchar] (8), @Copacity [varchar] (9), @Number\_Transport [varchar] (10), @Model\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Transport]  where [Model\_ID] = @Model\_ID and  [Length\_Transport] = @Length\_Transport and  [Width\_Transport] = @Width\_Transport and  [Height\_Transport] = @Height\_Transport and  [Load\_Copacity] = @Load\_Copacity and  [Copacity] = @Copacity and  [Number\_Transport] = @Number\_Transport)  if (@exsist > 0)  print('Данный о транспорте уже есть в таблице')  else  update [dbo].[Transport] set  [Length\_Transport] = @Length\_Transport,  [Width\_Transport] = @Width\_Transport,  [Height\_Transport] = @Height\_Transport,  [Load\_Copacity] = @Load\_Copacity,  [Copacity] = @Copacity,  [Number\_Transport] = @Number\_Transport,  [Model\_ID] = @Model\_ID  where  [ID\_Transport] = @ID\_Transport  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Transport\_Delete]  @ID\_Transport [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Transport]  where [ID\_Transport] = @ID\_Transport)  if (@any\_child\_record > 0)  print('Транспорт не могут быть удалены так как в таблице  "Страна Транспорта", "Маршрутный лист" есть связанные данные!')  else  delete from [dbo].[Transport]  where  [ID\_Transport] = @ID\_Transport  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Customer]  (  [ID\_Customer] [int] not null identity(1,1),  [Name\_Customer\_Organization] [varchar] (50) not null,  [Organization\_Address] [varchar] (max) not null,  [TIN] [varchar] (12) not null,  [BIC] [varchar] (9) not null,  [OKPO] [varchar] (8) not null,  [Login\_Customer] [varchar] (50) not null,  [Password\_Customer] [varchar] (32) not null,  [Name\_Customer] [varchar] (32) not null,  [Surname\_Customer] [varchar] (32) not null,  [Lastname\_Customer] [varchar] (32) null default('-'),  [Type\_Organization\_ID] [int] not null  constraint [PK\_Customer] primary key clustered  ([ID\_Customer] ASC) on [PRIMARY],  constraint [CH\_TIN] check  ([TIN] 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 [UQ\_TIN] unique ([TIN]),  constraint [CH\_BIC] check ([BIC] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')),  constraint [CH\_OKPO] check ([OKPO] like ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]')),  constraint [UQ\_OKPO] unique ([OKPO]),  constraint [CH\_Login\_Customer] check  (len([Login\_Customer]) >=4),  constraint [UQ\_Login\_Customer] unique ([Login\_Customer]),  constraint [CH\_Password\_Customer] check  (len([Password\_Customer]) >=4),  constraint [FK\_Type\_Organization\_Customer] foreign key ([Type\_Organization\_ID])  references [dbo].[Type\_Organization] ([ID\_Type\_Organization])  )  go | create or alter procedure [dbo].[Customer\_Insert]  @Name\_Customer\_Organization [varchar] (50), @Organization\_Address [varchar] (max), @TIN [varchar] (12), @BIC [varchar] (9) ,@OKPO [varchar] (8),  @Login\_Customer [varchar] (50), @Password\_Customer [varchar] (32), @Name\_Customer [varchar] (32), @Surname\_Customer [varchar] (32),  @Lastname\_Customer [varchar] (32), @Type\_Organization\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Customer]  where [Name\_Customer\_Organization] = @Name\_Customer\_Organization and  [Organization\_Address] = @Organization\_Address and  [TIN] = @TIN and  [BIC] = @BIC and  [OKPO] = @OKPO and  [Login\_Customer] = @Login\_Customer and  [Password\_Customer] = @Password\_Customer and  [Name\_Customer] = @Name\_Customer and  [Surname\_Customer] = @Surname\_Customer and  [Lastname\_Customer] = @Lastname\_Customer and  [Type\_Organization\_ID] = @Type\_Organization\_ID)  if (@exsist > 0)  print('Данные о Заказчике уже существует в таблице')  else  insert into [dbo].[Customer] ([Name\_Customer\_Organization], [Organization\_Address], [TIN], [BIC], [OKPO], [Login\_Customer], [Password\_Customer], [Name\_Customer], [Surname\_Customer], [Lastname\_Customer], [Type\_Organization\_ID])  values (@Name\_Customer\_Organization, @Organization\_Address, @TIN, @BIC, @OKPO, @Login\_Customer, @Password\_Customer, @Name\_Customer, @Surname\_Customer, @Lastname\_Customer, @Type\_Organization\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Customer\_Update]  @ID\_Customer [int], @Name\_Customer\_Organization [varchar] (50), @Organization\_Address [varchar] (max), @TIN [varchar] (12), @BIC [varchar] (9) ,@OKPO [varchar] (8),  @Login\_Customer [varchar] (50), @Password\_Customer [varchar] (32), @Name\_Customer [varchar] (32), @Surname\_Customer [varchar] (32),  @Lastname\_Customer [varchar] (32), @Type\_Organization\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Customer]  where [Name\_Customer\_Organization] = @Name\_Customer\_Organization and  [Organization\_Address] = @Organization\_Address and  [TIN] = @TIN and  [BIC] = @BIC and  [OKPO] = @OKPO and  [Login\_Customer] = @Login\_Customer and  [Password\_Customer] = @Password\_Customer and  [Name\_Customer] = @Name\_Customer and  [Surname\_Customer] = @Surname\_Customer and  [Lastname\_Customer] = @Lastname\_Customer and  [Type\_Organization\_ID] = @Type\_Organization\_ID)  if (@exsist > 0)  print('Данный о Заказчике уже есть в таблице')  else  update [dbo].[Customer] set  [Name\_Customer\_Organization] = @Name\_Customer\_Organization,  [Organization\_Address] = @Organization\_Address,  [TIN] = @TIN,  [BIC] = @BIC,  [OKPO] = @OKPO,  [Login\_Customer] = @Login\_Customer,  [Password\_Customer] = @Password\_Customer,  [Name\_Customer] = @Name\_Customer,  [Surname\_Customer] = @Surname\_Customer,  [Lastname\_Customer] = @Lastname\_Customer,  [Type\_Organization\_ID] = @Type\_Organization\_ID  where  [ID\_Customer] = @ID\_Customer  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Customer\_Delete]  @ID\_Customer [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Customer]  where [ID\_Customer] = @ID\_Customer)  if (@any\_child\_record > 0)  print('Заказчик не может быть удалены так как в таблице  "Заявка", "Должность Заказчика" есть связанные данные!')  else  delete from [dbo].[Customer]  where  [ID\_Customer] = @ID\_Customer  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Carrier]  (  [ID\_Carrier] [int] not null identity(1,1),  [Name\_Carrier] [varchar] (50) not null,  [Type\_Organization\_ID] [int] not null  constraint [PK\_Carrier] primary key clustered  ([ID\_Carrier] ASC) on [PRIMARY],  constraint [UQ\_Name\_Carrier] unique ([Name\_Carrier]),  constraint [FK\_Type\_Organization\_Carrier] foreign key ([Type\_Organization\_ID])  references [dbo].[Type\_Organization] ([ID\_Type\_Organization])  )  go | create or alter procedure [dbo].[Carrier\_Insert]  @Name\_Carrier [varchar] (50), @Type\_Organization\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Carrier]  where [Name\_Carrier] = @Name\_Carrier and  [Type\_Organization\_ID] = @Type\_Organization\_ID)  if (@exsist > 0)  print('Данные о Перевозчике уже существует в таблице')  else  insert into [dbo].[Carrier] ([Name\_Carrier], [Type\_Organization\_ID])  values (@Name\_Carrier, @Type\_Organization\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Carrier\_Update]  @ID\_Carrier [int], @Name\_Carrier [varchar] (50), @Type\_Organization\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Carrier]  where [Name\_Carrier] = @Name\_Carrier and  [Type\_Organization\_ID] = @Type\_Organization\_ID)  if (@exsist > 0)  print('Данный о Перевозчике уже есть в таблице')  else  update [dbo].[Carrier] set  [Name\_Carrier] = @Name\_Carrier,  [Type\_Organization\_ID] = @Type\_Organization\_ID  where  [ID\_Carrier] = @ID\_Carrier  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Carrier\_Delete]  @ID\_Carrier [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Carrier]  where [ID\_Carrier] = @ID\_Carrier)  if (@any\_child\_record > 0)  print('Перевозчки не может быть удален так как в таблице  "Сотрудника", "Маршрутный лист" есть связанные данные!')  else  delete from [dbo].[Carrier]  where  [ID\_Carrier] = @ID\_Carrier  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Employee]  (  [ID\_Employee] [int] not null identity(1,1),  [Name\_Employee] [varchar] (32) not null,  [Surname\_Employee] [varchar] (32) not null,  [Lastname\_Employee] [varchar] (32) null default('-'),  [SNILS] [varchar] (14) not null,  [FOMS] [varchar] (16) not null,  [Login\_Employee] [varchar] (50) not null,  [Password\_Employee] [varchar] (32) not null,  [Carrier\_ID] [int] not null  constraint [PK\_Employee] primary key clustered  ([ID\_Employee] ASC) on [PRIMARY],  constraint [UQ\_SNILS] unique ([SNILS]),  constraint [CH\_SNILS] check  ([SNILS] like ('[0-9][0-9][0-9]-[0-9][0-9][0-9]-[0-9][0-9][0-9] [0-9][0-9]')),  constraint [UQ\_FOMS] unique ([FOMS]),  constraint [CH\_FOMS] check  ([FOMS] 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 [UQ\_Login\_Employee] unique ([Login\_Employee]),  constraint [CH\_Login\_Employee] check (len([Login\_Employee])>=8),  constraint [CH\_Password\_Employee] check  (len([Password\_Employee]) >=4),  constraint [FK\_Carrier\_Employee] foreign key ([Carrier\_ID])  references [dbo].[Carrier] ([ID\_Carrier])  )  go | create or alter procedure [dbo].[Employee\_Insert]  @Name\_Employee [varchar] (32), @Surname\_Employee [varchar] (32), @Lastname\_Employee [varchar] (32), @SNILS [varchar] (14), @FOMS [varchar] (16),  @Login\_Employee [varchar] (50), @Password\_Employee [varchar] (32), @Carrier\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Employee]  where [Name\_Employee] = @Name\_Employee and  [Surname\_Employee] = @Surname\_Employee and  [Lastname\_Employee] = @Lastname\_Employee and  [SNILS] = @SNILS and  [FOMS] = @FOMS and  [Login\_Employee] = @Login\_Employee and  [Password\_Employee] = @Password\_Employee and  [Carrier\_ID] = @Carrier\_ID)  if (@exsist > 0)  print('Данные о Сотруднике уже существует в таблице')  else  insert into [dbo].[Employee] ([Name\_Employee], [Surname\_Employee], [Lastname\_Employee], [SNILS], [FOMS], [Login\_Employee], [Password\_Employee], [Carrier\_ID])  values (@Name\_Employee, @Surname\_Employee, @Lastname\_Employee, @SNILS, @FOMS, @Login\_Employee, @Password\_Employee, @Carrier\_ID)  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Employee\_Update]  @ID\_Employee [int], @Name\_Employee [varchar] (32), @Surname\_Employee [varchar] (32), @Lastname\_Employee [varchar] (32), @SNILS [varchar] (14),  @FOMS [varchar] (16), @Login\_Employee [varchar] (50), @Password\_Employee [varchar] (32), @Carrier\_ID [int]  as  declare @exsist [int] = (select count(\*) from  [dbo].[Employee]  where [Name\_Employee] = @Name\_Employee and  [Surname\_Employee] = @Surname\_Employee and  [Lastname\_Employee] = @Lastname\_Employee and  [SNILS] = @SNILS and  [FOMS] = @FOMS and  [Login\_Employee] = @Login\_Employee and  [Password\_Employee] = @Password\_Employee and  [Carrier\_ID] = @Carrier\_ID)  if (@exsist > 0)  print('Данный о Сотруднике уже есть в таблице')  else  update [dbo].[Employee] set  [Name\_Employee] = @Name\_Employee,  [Surname\_Employee] = @Surname\_Employee,  [Lastname\_Employee] = @Lastname\_Employee,  [SNILS] = @SNILS,  [FOMS] = @FOMS,  [Login\_Employee] = @Login\_Employee,  [Password\_Employee] = @Password\_Employee,  [Carrier\_ID] = @Carrier\_ID  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].[Employee]  where [ID\_Employee] = @ID\_Employee)  if (@any\_child\_record > 0)  print('Сотрудники не могут быть удалены так как в таблице  "Права сотрудника", "Должность Сотрудника" есть связанные данные!')  else  delete from [dbo].[Employee]  where  [ID\_Employee] = @ID\_Employee  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Route\_Sheet]  (  [ID\_Route\_Sheet] [int] not null identity(1,1),  [Number\_Route\_Sheet] [varchar] (max) not null,  [Date\_Route\_Sheet] [date] not null default(getdate()),  [Time\_Route\_Sheet] [time] not null default(getdate()),  [Carrier\_ID] [int] not null,  [Transport\_ID] [int] not null  constraint [PK\_Route\_Sheet] primary key clustered  ([ID\_Route\_Sheet] ASC) on [PRIMARY],  constraint [CH\_Number\_Route\_Sheet] check  ([Number\_Route\_Sheet] like ('[0-9][0-9][0-9][0-9]-[0-9][0-9]')),  constraint [FK\_Carrier\_Route\_Sheet] foreign key ([Carrier\_ID])  references [dbo].[Carrier] ([ID\_Carrier]),  constraint [FK\_Transport\_Route\_Sheet] foreign key ([Transport\_ID])  references [dbo].[Transport] ([ID\_Transport]),  )  go | create or alter procedure [dbo].[Route\_Sheet\_Insert]  @Number\_Route\_Sheet [varchar] (max), @Date\_Route\_Sheet [date], @Time\_Route\_Sheet [time], @Carrier\_ID [int], @Transport\_ID [int]  as  begin try  insert into [Route\_Sheet] ([Number\_Route\_Sheet], [Date\_Route\_Sheet], [Time\_Route\_Sheet], [Carrier\_ID], [Transport\_ID])  values (@Number\_Route\_Sheet, @Date\_Route\_Sheet, @Time\_Route\_Sheet, @Carrier\_ID, @Transport\_ID)  end try  begin catch  print('Ошибка при вводе данных в маршрутный лист')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Route\_Sheet\_Update]  @ID\_Route\_Sheet [int], @Number\_Route\_Sheet [varchar] (max), @Date\_Route\_Sheet [date], @Time\_Route\_Sheet [time], @Carrier\_ID [int], @Transport\_ID [int]  as  begin try  update [dbo].[Route\_Sheet] set  [Number\_Route\_Sheet] = @Number\_Route\_Sheet,  [Date\_Route\_Sheet] = @Date\_Route\_Sheet,  [Time\_Route\_Sheet] = @Time\_Route\_Sheet,  [Carrier\_ID] = @Carrier\_ID,  [Transport\_ID] = @Transport\_ID  where  [ID\_Route\_Sheet] = @ID\_Route\_Sheet  end try  begin catch  print ('Ошибка при вводе данных в таблицу маршрутный лист')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Route\_Sheet\_Delete]  @ID\_Route\_Sheet [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Route\_Sheet]  where [ID\_Route\_Sheet] = @ID\_Route\_Sheet)  if (@any\_child\_record > 0)  print('Маршрутный лист не может быть удален так как в таблице  "Заявка", "Точки маршрутного листа" есть связанные данные!')  else  delete from [dbo].[Route\_Sheet]  where  [ID\_Route\_Sheet] = @ID\_Route\_Sheet  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Application]  (  [ID\_Application] [int] not null identity(1,1),  [Number\_Application] [varchar] (max) not null,  [Date\_Application] [date] not null default(getdate()),  [Time\_Application] [time] not null default(getdate()),  [Status\_Application] [varchar] (15) not null,  [Customer\_ID] [int] not null,  [Route\_Sheet\_ID] [int] not null  constraint [PK\_Application] primary key clustered  ([ID\_Application] ASC) on [PRIMARY],  constraint [CH\_Status\_Application] check  ([Status\_Application] in ('Обработан', 'Обрабатывается', 'Отменен')),  constraint [FK\_Customer\_Application] foreign key ([Customer\_ID])  references [dbo].[Customer] ([ID\_Customer]),  constraint [FK\_Route\_Sheet\_Application] foreign key ([Route\_Sheet\_ID])  references [dbo].[Route\_Sheet] ([ID\_Route\_Sheet])  )  go | create or alter procedure [dbo].[Application\_Insert]  @Number\_Application [varchar] (max), @Date\_Application [date], @Time\_Application [time], @Status\_Application [varchar] (15),  @Customer\_ID [int], @Route\_Sheet\_ID [int]  as  begin try  insert into [Application] ([Number\_Application], [Date\_Application], [Time\_Application], [Status\_Application], [Customer\_ID], [Route\_Sheet\_ID])  values (@Number\_Application, @Date\_Application, @Time\_Application, @Status\_Application, @Customer\_ID, @Route\_Sheet\_ID)  end try  begin catch  print('Ошибка при вводе данных в заявку')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Application\_Update]  @ID\_Application [int], @Number\_Application [varchar] (max), @Date\_Application [date], @Time\_Application [time], @Status\_Application [varchar] (15),  @Customer\_ID [int], @Route\_Sheet\_ID [int]  as  begin try  update [dbo].[Application] set  [Number\_Application] = @Number\_Application,  [Date\_Application] = @Date\_Application,  [Time\_Application] = @Time\_Application,  [Status\_Application] = @Status\_Application,  [Customer\_ID] = @Customer\_ID,  [Route\_Sheet\_ID] = @Route\_Sheet\_ID  where  [ID\_Application] = @ID\_Application  end try  begin catch  print ('Ошибка при вводе данных в таблицу заявка')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Application\_Delete]  @ID\_Application [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Application]  where [ID\_Application] = @ID\_Application)  if (@any\_child\_record > 0)  print('Таблица Заявка может быть удалена так как у неё нет связей')  else  delete from [dbo].[Application]  where  [ID\_Application] = @ID\_Application  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Points\_Route\_Sheet]  (  [ID\_Point\_Route\_Sheet] [int] not null identity(1,1),  [Delivery\_Point\_ID] [int] not null,  [Route\_Sheet\_ID] [int] not null  constraint [PK\_Point\_Route\_Sheet] primary key clustered  ([ID\_Point\_Route\_Sheet] ASC) on [PRIMARY],  constraint [FK\_Delivery\_Point] foreign key ([Delivery\_Point\_ID])  references [dbo].[Delivery\_Points] ([ID\_Delivery\_Point]),  constraint [FK\_Route\_Sheet] foreign key ([Route\_Sheet\_ID])  references [dbo].[Route\_Sheet] ([ID\_Route\_Sheet]),  )  go | create or alter procedure [dbo].[Points\_Route\_Sheet\_Insert]  @Delivery\_Point\_ID [int], @Route\_Sheet\_ID [int]  as  begin try  insert into [Points\_Route\_Sheet] ([Delivery\_Point\_ID], [Route\_Sheet\_ID])  values (@Delivery\_Point\_ID, @Route\_Sheet\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу точки маршрутного листа')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Points\_Route\_Sheet\_Update]  @ID\_Points\_Route\_Sheet [int], @Delivery\_Point\_ID [int], @Route\_Sheet\_ID [int]  as  begin try  update [dbo].[Points\_Route\_Sheet] set  [Delivery\_Point\_ID] = @Delivery\_Point\_ID,  [Route\_Sheet\_ID] = @Route\_Sheet\_ID  where  [ID\_Point\_Route\_Sheet] = @ID\_Points\_Route\_Sheet  end try  begin catch  print ('Ошибка при вводе данных в таблицу точки маршрутного листа')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Points\_Route\_Sheet\_Delete]  @ID\_Points\_Route\_Sheet [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Points\_Route\_Sheet]  where [ID\_Point\_Route\_Sheet] = @ID\_Points\_Route\_Sheet)  if (@any\_child\_record > 0)  print('Таблица точки маршрутного листа может быть удалена так как у неё нет связей')  else  delete from [dbo].[Points\_Route\_Sheet]  where  [ID\_Point\_Route\_Sheet] = @ID\_Points\_Route\_Sheet  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Post\_Customer]  (  [ID\_Post\_Customer] [int] not null identity(1,1),  [Post\_ID] [int] not null,  [Customer\_ID] [int] not null  constraint [PK\_Post\_Customer] primary key clustered  ([ID\_Post\_Customer] ASC) on [PRIMARY],  constraint [FK\_Post] foreign key ([Post\_ID])  references [dbo].[Post] ([ID\_Post]),  constraint [FK\_Customer] foreign key ([Customer\_ID])  references [dbo].[Customer] ([ID\_Customer]),  )  go | create or alter procedure [dbo].[Post\_Customer\_Insert]  @Post\_ID [int], @Customer\_ID [int]  as  begin try  insert into [Post\_Customer] ([Post\_ID], [Customer\_ID])  values (@Post\_ID, @Customer\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу должность заказчика')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Post\_Customer\_Update]  @ID\_Post\_Customer [int], @Post\_ID [int], @Customer\_ID [int]  as  begin try  update [dbo].[Post\_Customer] set  [Post\_ID] = @Post\_ID,  [Customer\_ID] = @Customer\_ID  where  [ID\_Post\_Customer] = @ID\_Post\_Customer  end try  begin catch  print ('Ошибка при вводе данных в таблицу должность заказчика')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Post\_Customer\_Delete]  @ID\_Post\_Customer [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Post\_Customer]  where [ID\_Post\_Customer] = @ID\_Post\_Customer)  if (@any\_child\_record > 0)  print('Таблица должность заказчика может быть удалена так как у неё нет связей')  else  delete from [dbo].[Post\_Customer]  where  [ID\_Post\_Customer] = @ID\_Post\_Customer  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Country\_Trasnport]  (  [ID\_Country\_Transport] [int] not null identity(1,1),  [Country\_ID] [int] not null,  [Transport\_ID] [int] not null  constraint [PK\_Country\_Transport] primary key clustered  ([ID\_Country\_Transport] ASC) on [PRIMARY],  constraint [FK\_Country] foreign key ([Country\_ID])  references [dbo].[Country] ([ID\_Country]),  constraint [FK\_Transport] foreign key ([Transport\_ID])  references [dbo].[Transport] ([ID\_Transport]),  )  go | create or alter procedure [dbo].[Country\_Trasnport\_Insert]  @Country\_ID [int], @Transport\_ID [int]  as  begin try  insert into [Country\_Trasnport] ([Country\_ID], [Transport\_ID])  values (@Country\_ID, @Transport\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу страна транспорта')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Country\_Trasnport\_Update]  @ID\_Country\_Transport [int], @Country\_ID [int], @Transport\_ID [int]  as  begin try  update [dbo].[Country\_Trasnport] set  [Country\_ID] = @Country\_ID,  [Transport\_ID] = @Transport\_ID  where  [ID\_Country\_Transport] = @ID\_Country\_Transport  end try  begin catch  print ('Ошибка при вводе данных в таблицу страна транспорта')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Country\_Trasnport\_Delete]  @ID\_Country\_Transport [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Country\_Trasnport]  where [ID\_Country\_Transport] = @ID\_Country\_Transport)  if (@any\_child\_record > 0)  print('Таблица страна транспорта может быть удалена так как у неё нет связей')  else  delete from [dbo].[Country\_Trasnport]  where  [ID\_Country\_Transport] = @ID\_Country\_Transport  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Post\_Employee]  (  [ID\_Post\_Employee] [int] not null identity(1,1),  [Post\_ID] [int] not null,  [Employee\_ID] [int] not null  constraint [PK\_Post\_Employee] primary key clustered  ([ID\_Post\_Employee] ASC) on [PRIMARY],  constraint [FK\_Post\_Employee] foreign key ([Post\_ID])  references [dbo].[Post] ([ID\_Post]),  constraint [FK\_Employee] foreign key ([Employee\_ID])  references [dbo].[Employee] ([ID\_Employee]),  )  go | create or alter procedure [dbo].[Post\_Employee\_Insert]  @Post\_ID [int], @Employee\_ID [int]  as  begin try  insert into [Post\_Employee] ([Post\_ID], [Employee\_ID])  values (@Post\_ID, @Employee\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу должность сотрудника')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Post\_Employee\_Update]  @ID\_Post\_Employee [int], @Post\_ID [int], @Employee\_ID [int]  as  begin try  update [dbo].[Post\_Employee] set  [Post\_ID] = @Post\_ID,  [Employee\_ID] = @Employee\_ID  where  [ID\_Post\_Employee] = @ID\_Post\_Employee  end try  begin catch  print ('Ошибка при вводе данных в таблицу должность сотрудника')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Post\_Employee\_Delete]  @ID\_Post\_Employee [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Post\_Employee]  where [ID\_Post\_Employee] = @ID\_Post\_Employee)  if (@any\_child\_record > 0)  print('Таблица должность сотрудника может быть удалена так как у неё нет связей')  else  delete from [dbo].[Post\_Employee]  where  [ID\_Post\_Employee] = @ID\_Post\_Employee  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Cargo\_Application]  (  [ID\_Cargo\_Application] [int] not null identity(1,1),  [Cargo\_ID] [int] not null,  [Application\_ID] [int] null  constraint [PK\_Cargo\_Application] primary key clustered  ([ID\_Cargo\_Application] ASC) on [PRIMARY],  constraint [FK\_Cargo] foreign key ([Cargo\_ID])  references [dbo].[Cargo] ([ID\_Cargo]),  constraint [FK\_Application] foreign key ([Application\_ID])  references [dbo].[Application] ([ID\_Application])  )  go | create or alter procedure [dbo].[Cargo\_Application\_Insert]  @Cargo\_ID [int], @Application\_ID [int]  as  begin try  insert into [Cargo\_Application] ([Cargo\_ID], [Application\_ID])  values (@Cargo\_ID, @Application\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу заявка груза')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Cargo\_Application\_Update]  @ID\_Cargo\_Application [int], @Cargo\_ID [int], @Application\_ID [int]  as  begin try  update [dbo].[Cargo\_Application] set  [Cargo\_ID] = @Cargo\_ID,  [Application\_ID] = @Application\_ID  where  [ID\_Cargo\_Application] = @ID\_Cargo\_Application  end try  begin catch  print ('Ошибка при вводе данных в таблицу заявка груза')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Cargo\_Application\_Delete]  @ID\_Cargo\_Application [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Cargo\_Application]  where [ID\_Cargo\_Application] = @ID\_Cargo\_Application)  if (@any\_child\_record > 0)  print('Таблица заявка груза может быть удалена так как у неё нет связей')  else  delete from [dbo].[Cargo\_Application]  where  [ID\_Cargo\_Application] = @ID\_Cargo\_Application  go |
| Скрипт таблицы | Хранимая процедура на добавление данных |
| create table [dbo].[Employee\_License]  (  [ID\_Employee\_License] [int] not null identity(1,1),  [Employee\_ID] [int] not null,  [License\_ID] [int] null  constraint [PK\_Employee\_License] primary key clustered  ([ID\_Employee\_License] ASC) on [PRIMARY],  constraint [FK\_Employee\_License] foreign key ([Employee\_ID])  references [dbo].[Employee] ([ID\_Employee]),  constraint [FK\_License] foreign key ([License\_ID])  references [dbo].[License] ([ID\_License])  )  go | create or alter procedure [dbo].[Employee\_License\_Insert]  @Employee\_ID [int], @License\_ID [int]  as  begin try  insert into [Employee\_License] ([Employee\_ID], [License\_ID])  values (@Employee\_ID, @License\_ID)  end try  begin catch  print('Ошибка при вводе данных в таблицу права сотрудника')  end catch  go |
| Хранимая процедура на изменение данных |
| create or alter procedure [dbo].[Employee\_License\_Update]  @ID\_Employee\_License [int], @Employee\_ID [int], @License\_ID [int]  as  begin try  update [dbo].[Employee\_License] set  [Employee\_ID] = @Employee\_ID ,  [License\_ID] = @License\_ID  where  [ID\_Employee\_License] = @ID\_Employee\_License  end try  begin catch  print ('Ошибка при вводе данных в таблицу права сотрудника')  end catch  go |
| Хранимая процедура на удаление данных |
| create or alter procedure [dbo].[Employee\_License\_Delete]  @ID\_Employee\_License [int]  as  declare @any\_child\_record [int] = (select count(\*) from [dbo].[Employee\_License]  where [ID\_Employee\_License] = @ID\_Employee\_License)  if (@any\_child\_record > 0)  print('Таблица права сотрудника может быть удалена так как у неё нет связей')  else  delete from [dbo].[Employee\_License]  where  [ID\_Employee\_License] = @ID\_Employee\_License  go |

1. Иллюстрация обозревателя объекта

