

**Федеральное государственное образовательное бюджетное  
учреждение высшего образования  
«Финансовый университет при Правительстве Российской Федерации»  
(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЁТ**

По учебной практике

Специальность 09.02.07 «Информационные системы и программирование»  
(код) (наименование)

Профессиональный модуль ПМ.01 Разработка модулей программного  
обеспечения для компьютерных систем  
(код) (наименование)

Междисциплинарный курс МДК.01.04 Системное программирование  
(код) (наименование)

Выполнил:

Студент 4 курса 4ИСИП-321 учебной группы  
(номер) (номер)

  
(подпись)

Д.О. Слуданов  
(инициалы, фамилия)

Проверил:

Руководитель практики от Колледжа информатики и  
программирования

Преподаватель 1КК, к.т.н., доцент Е.Л. Альшакова  
(квалификационная категория или звание, должность) (инициалы, фамилия)

### Перечень работ, выполненных в ходе учебной практики

№ п/п	Виды работ	Оценка
1	Проектирование базы данных и разработка объектов базы данных	Отлично
2	Разработка приложения с использованием WPF	Отлично

## **Содержание**

<b>Учебная практика 2 .....</b>	<b>5</b>
<b>Проектирование базы данных и разработка объектов базы данных.....</b>	<b>5</b>
<b>Учебная практика 2 .....</b>	<b>18</b>
<b>Разработка приложения с использованием WPF .....</b>	<b>18</b>

Федеральное государственное образовательное бюджетное  
учреждение высшего образования  
**«Финансовый университет при Правительстве Российской Федерации»**  
**(Финансовый университет)**  
Колледж информатики и программирования

## **ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1**

**Тема: Проектирование базы данных и разработка объектов базы данных**

Студент: Слуданов Д. О.

Группа: 4ИСИП-321

Преподаватель: Альшакова Е.Л.

Дата: 13.11.2024

## Учебная практика 2

### Проектирование базы данных и разработка объектов базы данных

**Цель работы:** разработать базу данных в соответствии с заданием, наполнить ее сущностями и атрибутами, реализовать ER-диаграмму.

Проанализируем предметную область. Компания занимается производством и реализует свою продукцию через партнеров, которые доставляют продукцию компании до конечных потребителей. Для эффективного взаимодействия с партнерами и контроля их работы требуется система, позволяющая обрабатывать всю информацию в цифровом формате.

Исходя из предметной области реализуем базу данных. На рисунке 1 представлена ER-диаграмма базы данных.

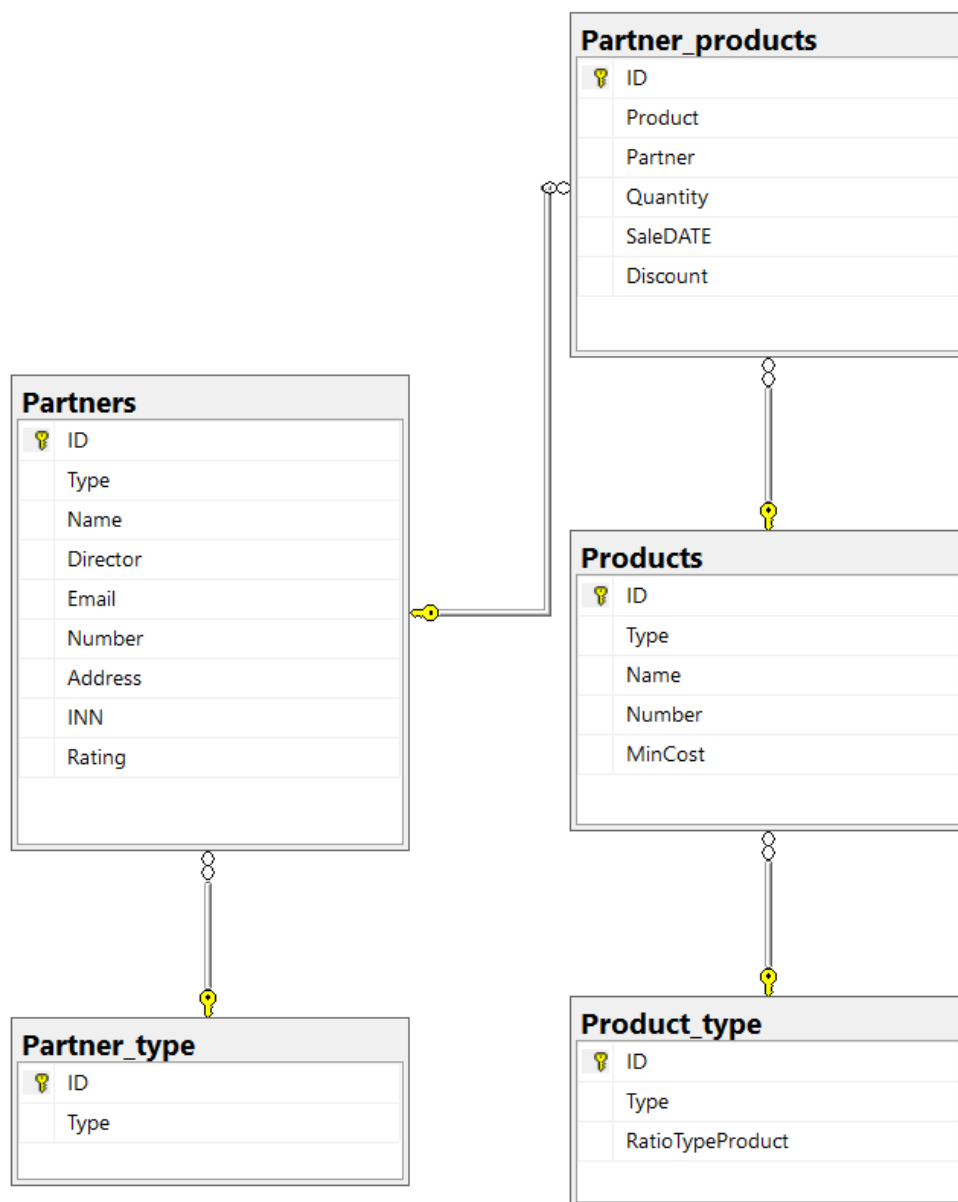


Рисунок 1 – ER-диаграмма базы данных

После того, как мы спроектировали базу данных, нам требуется заполнить ее начальными данными, чтобы в ходе разработки WPF приложения, мы могли понимать, как мы взаимодействуем с данными.

На рисунках 2-6 представлены данные для каждой сущности.

	ID	Product	Partner	Quantity	SaleDATE	Discount
1	1	1	1	15500	2023-03-23	NULL
2	2	3	1	12350	2023-12-18	NULL
3	3	4	1	37400	2024-06-07	NULL
4	4	2	2	35000	2022-12-02	NULL
5	5	5	2	1250	2023-05-17	NULL
6	6	3	2	1000	2024-06-07	NULL
7	7	1	2	7550	2024-07-01	NULL
8	8	1	3	7250	2023-01-22	NULL
9	9	2	3	2500	2024-07-05	NULL
10	10	4	4	59050	2023-03-20	NULL
11	11	3	4	37200	2024-03-12	NULL
12	12	5	4	4500	2024-05-14	NULL
13	13	3	11	50000	2023-09-19	NULL
14	14	4	11	670000	2023-11-10	NULL
15	15	1	11	35000	2024-04-15	NULL
16	16	2	11	25000	2024-06-12	NULL
17	17	1	12	50000	2024-12-17	NULL
18	18	2	13	100000	2024-12-17	NULL
19	19	2	14	50000	2024-12-17	NULL
20	20	3	15	200000	2024-12-17	NULL

Рисунок 2 – Таблица «Partner\_products»

	ID	Type
1	1	ЗАО
2	2	ООО
3	3	ПАО
4	4	ОАО

Рисунок 3 – Таблица «Partner\_type»

Results Messages									
	ID	Type	Name	Director	Email	Number	Address	INN	Rating
1	1	1	База Строитель	Иванова Александра Ивановна	aleksandraivanova@ml.ru	493 123 45 6	652050, Кемеровская область, город Юрга, ул. Лесн...	2222455179	7
2	2	2	Паркет 29	Петров Василий Петрович	vppetrov@vl.ru	987 123 56 7	164500, Архангельская область, город Северодвинс...	3333888520	7
3	3	3	Стройсервис	Соловьев Андрей Николаевич	ansolovev@st.ru	812 223 32 0	188910, Ленинградская область, город Приморск, у...	4440391035	7
4	4	4	Ремонт и отделка	Воробьева Екатерина Валерьевна	ekaterina.vorobeveva@ml.ru	+79299172902	143960, Московская область, город Реутов, ул. Сво...	1111520857	8
5	11	1	МонтажПро	Степанов Степан Сергеевич	stepanov@stepan.ru	+79128883333	309500, Белгородская область, город Старый Оскол...	5552431140	10
6	12	1	ООО "БанкПроБанк"	Петров Петр Петрович	bankprobank@gmail.com	+79999999999	ул. Центральная, 32, Москва	3612783123	10
7	13	2	МонтажПоу	Богданов Богдан Богданович	bogdanovich@mail.ru	+79666666666	ул. Центральная, 33, Москва	3462718934	10
8	14	2	вылоаовьрд	Тестов Тест Тестович	testovich@mail.ru	+78754221212	какая-то улица	3362178128	10
9	15	1	тестович	Тестов Тест Тестович	testovich@yandex.ru	+79299172902	ул. Маркситская	3162783267	10

Рисунок 4 – Таблица «Partners»

Results Messages			
	ID	Type	RatioTypeProduct
1	1	Ламинат	2,35
2	2	Массивная доска	5,15
3	3	Паркетная доска	4,34
4	4	Пробковое покрытие	1,5

Рисунок 5 – Таблица «Product\_type»

Results Messages					
	ID	Type	Name	Number	MinCost
1	1	3	Паркетная доска Ясень темный однополосная 14 мм	8758385	4456.90
2	2	3	Инженерная доска Дуб Французская елка однополосн...	8858958	7330.99
3	3	1	Ламинат Дуб дымчато-белый 33 класс 12 мм	7750282	1799.33
4	4	1	Ламинат Дуб серый 32 класс 8 мм с фаской	7028748	3890.41
5	5	4	Пробковое напольное клеевое покрытие 32 класс 4 мм	5012543	5450.59

Рисунок 6 – Таблица «Products»

Также, был создан скрипт базы данных, который представлен ниже в листинге 1.

Листинг 1 – Скрипт базы данных

```
USE [MasterFloor]
GO
/***** Object: Table [dbo].[Material_type]      Script Date: 19.12.24 20:01:31
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Material_type] (
```



```

        [ID] [int] IDENTITY(1,1) NOT NULL,
        [Type] [nvarchar](50) NOT NULL,
        [ScrapRate] [float] NULL,
PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Partner_products]      Script Date: 19.12.24
20:01:31 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partner_products](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Product] [int] NOT NULL,
    [Partner] [int] NOT NULL,
    [Quantity] [int] NULL,
    [SaleDATE] [date] NULL,
    [Discount] [decimal](3, 2) NULL,
PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Partner_type]      Script Date: 19.12.24 20:01:31
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partner_type](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Type] [nvarchar](50) NOT NULL,
PRIMARY KEY CLUSTERED
(
    [ID] ASC

```

```

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Partners]      Script Date: 19.12.24 20:01:31
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partners](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Type] [int] NOT NULL,
    [Name] [nvarchar](150) NOT NULL,
    [Director] [nvarchar](120) NOT NULL,
    [Email] [nvarchar](80) NULL,
    [Number] [char](12) NULL,
    [Address] [nvarchar](150) NOT NULL,
    [INN] [char](10) NULL,
    [Rating] [int] NULL,
    CONSTRAINT [PK__Partners__3214EC27BA9CF023] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Product_type]  Script Date: 19.12.24 20:01:31
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Product_type](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Type] [nvarchar](50) NOT NULL,
    [RatioTypeProduct] [float] NULL,
    PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]

```

```

) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Products]      Script Date: 19.12.24 20:01:31
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Products](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Type] [int] NULL,
    [Name] [nvarchar](150) NOT NULL,
    [Number] [char](10) NOT NULL,
    [MinCost] [decimal](10, 2) NULL,
PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
SET IDENTITY_INSERT [dbo].[Material_type] ON
GO
INSERT [dbo].[Material_type] ([ID], [Type], [ScrapRate]) VALUES (1, N'Тип
материала 1', 0.001)
GO
INSERT [dbo].[Material_type] ([ID], [Type], [ScrapRate]) VALUES (2, N'Тип
материала 2', 0.0095)
GO
INSERT [dbo].[Material_type] ([ID], [Type], [ScrapRate]) VALUES (3, N'Тип
материала 3', 0.0028)
GO
INSERT [dbo].[Material_type] ([ID], [Type], [ScrapRate]) VALUES (4, N'Тип
материала 4', 0.0055)
GO
INSERT [dbo].[Material_type] ([ID], [Type], [ScrapRate]) VALUES (5, N'Тип
материала 5', 0.0034)
GO
SET IDENTITY_INSERT [dbo].[Material_type] OFF
GO
SET IDENTITY_INSERT [dbo].[Partner_products] ON
GO

```

```
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (1, 1, 1, 15500, CAST(N'2023-03-23' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (2, 3, 1, 12350, CAST(N'2023-12-18' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (3, 4, 1, 37400, CAST(N'2024-06-07' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (4, 2, 2, 35000, CAST(N'2022-12-02' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (5, 5, 2, 1250, CAST(N'2023-05-17' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (6, 3, 2, 1000, CAST(N'2024-06-07' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (7, 1, 2, 7550, CAST(N'2024-07-01' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (8, 1, 3, 7250, CAST(N'2023-01-22' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (9, 2, 3, 2500, CAST(N'2024-07-05' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (10, 4, 4, 59050, CAST(N'2023-03-20' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (11, 3, 4, 37200, CAST(N'2024-03-12' AS Date),
NULL)
GO
```

```

INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (12, 5, 4, 4500, CAST(N'2024-05-14' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (13, 3, 11, 50000, CAST(N'2023-09-19' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (14, 4, 11, 670000, CAST(N'2023-11-10' AS
Date), NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (15, 1, 11, 35000, CAST(N'2024-04-15' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (16, 2, 11, 25000, CAST(N'2024-06-12' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (17, 1, 12, 50000, CAST(N'2024-12-17' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (18, 2, 13, 100000, CAST(N'2024-12-17' AS
Date), NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (19, 2, 14, 50000, CAST(N'2024-12-17' AS Date),
NULL)
GO
INSERT [dbo].[Partner_products] ([ID], [Product], [Partner], [Quantity],
[SaleDATE], [Discount]) VALUES (20, 3, 15, 200000, CAST(N'2024-12-17' AS
Date), NULL)
GO
SET IDENTITY_INSERT [dbo].[Partner_products] OFF
GO
SET IDENTITY_INSERT [dbo].[Partner_type] ON
GO
INSERT [dbo].[Partner_type] ([ID], [Type]) VALUES (1, N'3AO')
GO
INSERT [dbo].[Partner_type] ([ID], [Type]) VALUES (2, N'OOO')
GO
INSERT [dbo].[Partner_type] ([ID], [Type]) VALUES (3, N'ΠAO')

```

```

GO
INSERT [dbo].[Partner_type] ([ID], [Type]) VALUES (4, N'ОАО')
GO
SET IDENTITY_INSERT [dbo].[Partner_type] OFF
GO
SET IDENTITY_INSERT [dbo].[Partners] ON
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (1, 1, N'База Строитель', N'Иванова
Александра Ивановна', N'aleksandraivanova@ml.ru', N'493 123 45 6', N'652050,
Кемеровская область, город Юрга, ул. Лесная, 15', N'2222455179', 7)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (2, 2, N'Паркет 29', N'Петров Василий
Петрович', N'vprpetrov@vl.ru', N'987 123 56 7', N'164500, Архангельская
область, город Северодвинск, ул. Строителей, 18', N'3333888520', 7)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (3, 3, N'Стройсервис', N'Соловьев Андрей
Николаевич', N'ansolovev@st.ru', N'812 223 32 0', N'188910, Ленинградская
область, город Приморск, ул. Парковая, 21', N'4440391035', 7)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (4, 4, N'Ремонт и отделка', N'Воробьева
Екатерина Валерьевна', N'ekaterina.vorobeva@ml.ru', N'+79299172902', N'143960,
Московская область, город Реутов, ул. Свободы, 51', N'1111520857', 8)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (11, 1, N'МонтажПро', N'Степанов Степан
Сергеевич', N'stepanov@stepan.ru', N'+79128883333', N'309500, Белгородская
область, город Старый Оскол, ул. Рабочая, 122', N'5552431140', 10)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (12, 1, N'ООО "БанкПроБанк"', N'Петров Петр
Петрович', N'bankprobank@gmail.com', N'+79999999999 ', N'ул. Центральная, 32,
Москва', N'3612783123', 10)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (13, 2, N'МонтажЛоу', N'Богданов Богдан
Богданович', N'bogdanovich@mai.ru', N'+79666666666', N'ул. Центральная, 33,
Москва', N'3462718934', 10)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (14, 2, N'ВЫЛОАОВЫРД', N'Тестов Тест

```

```

Тестович', N'testovich@mail.ru', N'+78754221212', N'какая-то улица',
N'3362178128', 10)
GO
INSERT [dbo].[Partners] ([ID], [Type], [Name], [Director], [Email], [Number],
[Address], [INN], [Rating]) VALUES (15, 1, N'тестович', N'Тестов Тест
Тестович', N'testovich@yandex.ru', N'+79299172902', N'ул, Маркситская',
N'3162783267', 10)
GO
SET IDENTITY_INSERT [dbo].[Partners] OFF
GO
SET IDENTITY_INSERT [dbo].[Product_type] ON
GO
INSERT [dbo].[Product_type] ([ID], [Type], [RatioTypeProduct]) VALUES (1,
N'Ламинат', 2.35)
GO
INSERT [dbo].[Product_type] ([ID], [Type], [RatioTypeProduct]) VALUES (2,
N'Массивная доска', 5.15)
GO
INSERT [dbo].[Product_type] ([ID], [Type], [RatioTypeProduct]) VALUES (3,
N'Паркетная доска', 4.34)
GO
INSERT [dbo].[Product_type] ([ID], [Type], [RatioTypeProduct]) VALUES (4,
N'Пробковое покрытие', 1.5)
GO
SET IDENTITY_INSERT [dbo].[Product_type] OFF
GO
SET IDENTITY_INSERT [dbo].[Products] ON
GO
INSERT [dbo].[Products] ([ID], [Type], [Name], [Number], [MinCost]) VALUES (1,
3, N'Паркетная доска Ясень темный однополосная 14 мм', N'8758385 ',
CAST(4456.90 AS Decimal(10, 2)))
GO
INSERT [dbo].[Products] ([ID], [Type], [Name], [Number], [MinCost]) VALUES (2,
3, N'Инженерная доска Дуб Французская елка однополосная 12 мм', N'8858958 ',
CAST(7330.99 AS Decimal(10, 2)))
GO
INSERT [dbo].[Products] ([ID], [Type], [Name], [Number], [MinCost]) VALUES (3,
1, N'Ламинат Дуб дымчато-белый 33 класс 12 мм', N'7750282 ', CAST(1799.33 AS
Decimal(10, 2)))
GO
INSERT [dbo].[Products] ([ID], [Type], [Name], [Number], [MinCost]) VALUES (4,
1, N'Ламинат Дуб серый 32 класс 8 мм с фаской', N'7028748 ', CAST(3890.41 AS
Decimal(10, 2)))
GO

```

```
INSERT [dbo].[Products] ([ID], [Type], [Name], [Number], [MinCost]) VALUES (5,
4, N'Пробковое напольное клеевое покрытие 32 класс 4 мм', N'5012543 ',
CAST(5450.59 AS Decimal(10, 2)))
GO
SET IDENTITY_INSERT [dbo].[Products] OFF
GO
ALTER TABLE [dbo].[Partner_products] WITH CHECK ADD CONSTRAINT
[FK__Partner_p__Partn__440B1D61] FOREIGN KEY([Partner])
REFERENCES [dbo].[Partners] ([ID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[Partner_products] CHECK CONSTRAINT
[FK__Partner_p__Partn__440B1D61]
GO
ALTER TABLE [dbo].[Partner_products] WITH CHECK ADD FOREIGN KEY([Product])
REFERENCES [dbo].[Products] ([ID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[Partners] WITH CHECK ADD CONSTRAINT
[FK__Partners__Type__403A8C7D] FOREIGN KEY([Type])
REFERENCES [dbo].[Partner_type] ([ID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[Partners] CHECK CONSTRAINT [FK__Partners__Type__403A8C7D]
GO
ALTER TABLE [dbo].[Products] WITH CHECK ADD FOREIGN KEY([Type])
REFERENCES [dbo].[Product_type] ([ID])
ON DELETE CASCADE
GO
```



Федеральное государственное образовательное бюджетное  
учреждение высшего образования  
**«Финансовый университет при Правительстве Российской Федерации»**  
**(Финансовый университет)**  
Колледж информатики и программирования

## **ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2**

**Тема: Разработка кода программного продукта**

Студент: Слуданов Д. О.

Группа: 4ИСИП-321

Преподаватель: Альшакова Е.Л.

Дата: 16.12.2024

## Учебная практика 2

### Разработка приложения с использованием WPF

**Цель работы:** разработать WPF-приложение в соответствии с заданием, наполнить ее сущностями и атрибутами, написать листинг программы.

Было реализовано WPF-приложение под требования заказчика. При запуске данного приложения, у нас открывается окно, где выводиться список партнеров, краткая информация о партнере, а также скидка для партнера.

Помимо списка, были добавлены кнопки. Кнопка «Посмотреть продажи» действует тогда, когда пользователь выбрал партнера. После нажатия, производится переход на страницу, где пользователь может посмотреть продажи партнера. Кнопка «Добавить партнера» перенаправляет пользователя на страницу, где он вносит данные о партнере. К примеру, тип организации, название организации и другие данные. На рисунке 7 представлен пример стартовой страницы.

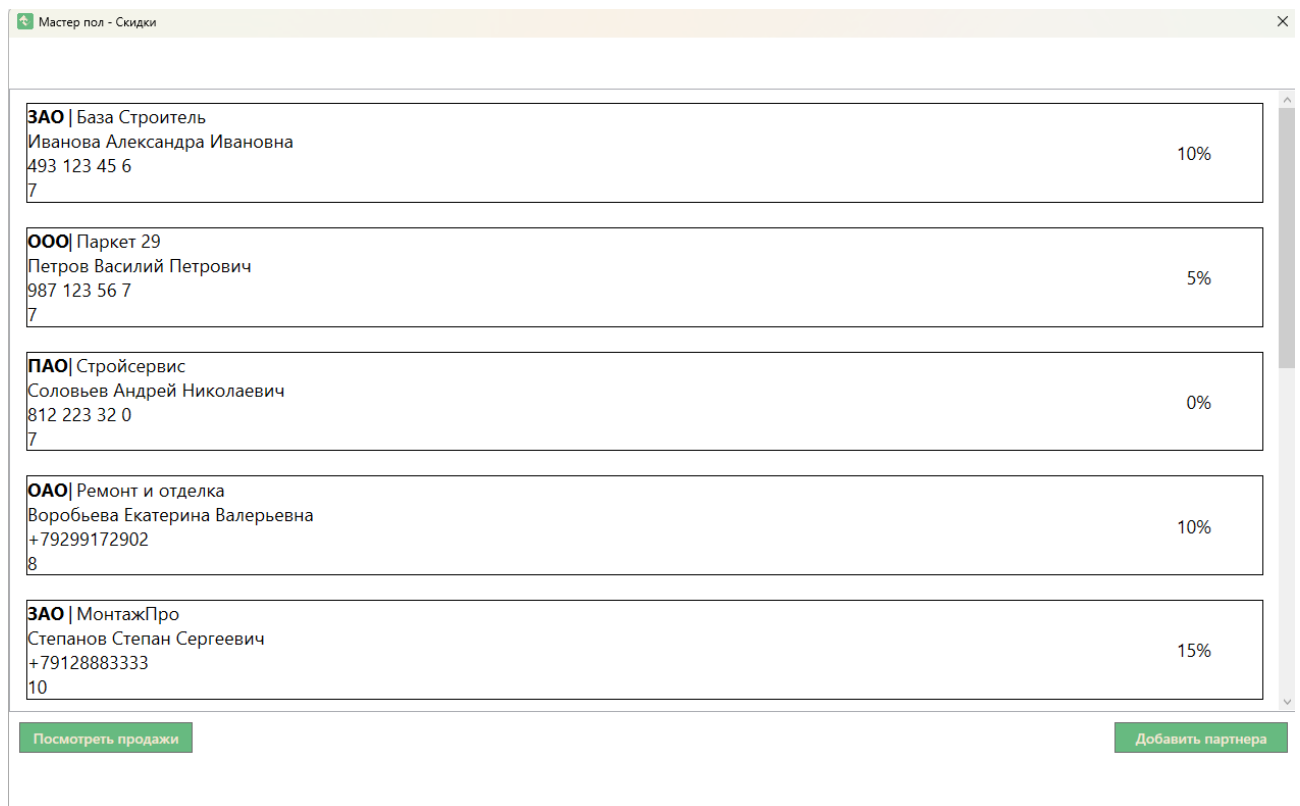


Рисунок 7 – Стартовая страница

Теперь посмотрим, что закулисами данной страницы. Ниже представлен хaml код, который отвечает за отображение элементов на стартовой странице, он



```

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Grid.Column="0"
                Text="{Binding Partner.Partner_type.Type}"
FontSize="18" FontWeight="Bold"
                Margin="0,0,133,0" />

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="180"
                Grid.Column="0" HorizontalAlignment="Left"
FontSize="18"
                Text="{Binding Partner.Name}"
Margin="48,0,0,0" />

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="10"
                Grid.Column="0" HorizontalAlignment="Left"
FontSize="18"
                Text="|" Margin="40,0,0,0"/>

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="370"
                Grid.Row="2" HorizontalAlignment="Left"
FontSize="18"
                Text="{Binding Partner.Director}" />

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="200"
                Grid.Row="3" HorizontalAlignment="Left"
FontSize="18"
                Text="{Binding Partner.Number}" />

        <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="200"
                Grid.Row="4" HorizontalAlignment="Left"
FontSize="18"
                Text="{Binding Partner.Rating}" />

        <TextBlock VerticalAlignment="Center"
TextAlignment="Right" Width="200"
                Grid.Row="2" Grid.Column="4"
HorizontalAlignment="Right" FontSize="18"
                Text="{Binding Discount,
StringFormat={} {0}%" Margin="0,0,50,0"
                Grid.RowSpan="2" />
    </Grid>

```

```

        </Border>
    </DataTemplate>
</ListView.ItemTemplate>
</ListView>

<Button x:Name="BtnAdd" Content="Добавить партнера" Grid.Row="2"
HorizontalAlignment="Right" VerticalAlignment="Center" Margin="0 0 10 0"
Click="BtnAdd_Click" Style="{StaticResource Button}"/>

<Button x:Name="BtnShowSales" Content="Посмотреть продажи"
Grid.Row="2" HorizontalAlignment="Left" VerticalAlignment="Center" Margin="10
0 0 0" Click="BtnShowSales_Click" Style="{StaticResource Button}"/>

</Grid>
</Page>

```

### Листинг 3 – Код логики работы стартовой страницы

```

using System.Data;
using System.Linq;
using System.Windows;
using System.Windows.Controls;

namespace MasterFloorAPP.Pages
{
    public partial class Discounts : Page
    {
        private readonly Entities db; // Контекст базы данных

        public Discounts()
        {
            InitializeComponent();
            db = new Entities(); // Инициализация контекста базы данных
            LoadDiscounts(); // Загружаем данные о скидках при инициализации
страницы
        }

        // Метод для загрузки скидок по партнерам
        private void LoadDiscounts()
        {
            // Группируем данные по партнерам и считаем общее количество
товаров по каждому партнеру
            var partnerSales = db.Partner_products
                .GroupBy(p => p.Partner) // Группируем по партнеру
                .Select(g => new
            {
                Partner = g.Key, // Идентификатор партнера

```

```

        TotalQuantity = g.Sum(p => p.Quantity ?? 0) // Суммируем
количество товаров для каждого партнера (с проверкой на null)
    })
    .ToList(); // Преобразуем результат в список

    // Для каждого партнера рассчитываем скидку на основе общего
количества товаров
    var partnerDiscounts = partnerSales.Select(ps => new
    {
        Partner = db.Partners.FirstOrDefault(p => p.ID == ps.Partner),
// Находим партнера по его ID
        TotalQuantity = ps.TotalQuantity, // Общее количество товаров
        Discount = CalculateDiscount(ps.TotalQuantity) // Рассчитываем
скидку на основе количества товаров
    }).ToList(); // Преобразуем результат в список

    // Устанавливаем источник данных для списка, который отображает
скидки
    ListUser.ItemsSource = partnerDiscounts;
}

// Метод для расчета скидки в зависимости от общего количества товаров
private decimal CalculateDiscount(int totalQuantity)
{
    // Логика определения скидки в зависимости от количества
    if (totalQuantity <= 10000)
        return 0; // Если общее количество товаров менее или равно
10000, скидка 0%
    else if (totalQuantity <= 50000)
        return 5; // Если общее количество товаров от 10001 до 50000,
скидка 5%
    else if (totalQuantity <= 300000)
        return 10; // Если общее количество товаров от 50001 до
300000, скидка 10%
    else
        return 15; // Если общее количество товаров больше 300000,
скидка 15%
}

private void BtnAdd_Click(object sender,
System.Windows.RoutedEventArgs e)
{
    NavigationService?.Navigate(new AddPartner(null));
}

```

```

        private void BtnShowSales_Click(object sender,
System.Windows.RoutedEventArgs e)
        {
            if (ListUser.SelectedItem != null) // Проверяем, что элемент
выбран
            {
                // Получаем выбранного партнера
                var selectedPartner = (dynamic)ListUser.SelectedItem;
                int partnerId = selectedPartner.Partner.ID;

                // Переход на страницу Sales с передачей ID партнёра
                NavigationService?.Navigate(new Sales(partnerId));
            }
            else
            {
                // Выводим сообщение, если элемент не выбран
                MessageBox.Show("Пожалуйста, выберите партнёра из списка.",
"Предупреждение", MessageBoxButton.OK, MessageBoxImage.Warning);
            }
        }

        private void ListUser_SelectionChanged(object sender,
SelectionChangedEventArgs e)
        {
            if (ListUser.SelectedItem != null)
            {
                var selectedPartner = (dynamic)ListUser.SelectedItem;
                var partner = selectedPartner.Partner;

                // Переход на страницу AddPartner с передачей выбранного
партнера
                NavigationService?.Navigate(new AddPartner(partner));
            }
        }
    }
}

```

Важно отметить, что мы в коде логики работы обращаемся к базе данных. База данных подключается через специальный фреймворк, он носит название «EntityFramework». После подключения, мы в модели экземпляром должны прописать следующий код, который представлен в листинге 4. А также представлен рисунок 8, где отображен результат подключения бд.

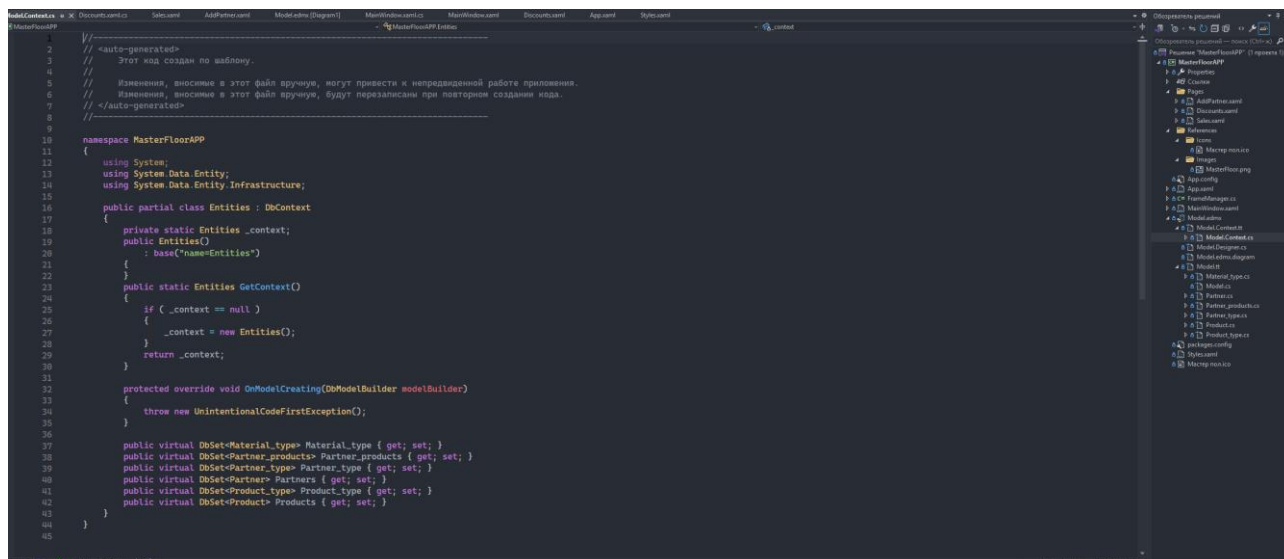


Рисунок 8 – Результат подключения базы данных

Листинг 4 – Код для создания экземпляра БД.

```
//-----
--
// <auto-generated>
//     Этот код создан по шаблону.
//
//     Изменения, вносимые в этот файл вручную, могут привести к непредвиденной
//     работе приложения.
//     Изменения, вносимые в этот файл вручную, будут перезаписаны при повторном
//     создании кода.
// </auto-generated>
//-----
--

namespace MasterFloorAPP
{
    using System;
    using System.Data.Entity;
    using System.Data.Entity.Infrastructure;

    public partial class Entities : DbContext
    {
        private static Entities _context;
        public Entities()
            : base("name=Entities")
        {
        }

        public static Entities GetContext()
        {
        }
    }
}
```



```

        if ( _context == null )
        {
            _context = new Entities();
        }
        return _context;
    }

    protected override void OnModelCreating(DbModelBuilder modelBuilder)
    {
        throw new UnintentionalCodeFirstException();
    }

    public virtual DbSet<Material_type> Material_type { get; set; }
    public virtual DbSet<Partner_products> Partner_products { get; set; }
    public virtual DbSet<Partner_type> Partner_type { get; set; }
    public virtual DbSet<Partner> Partners { get; set; }
    public virtual DbSet<Product_type> Product_type { get; set; }
    public virtual DbSet<Product> Products { get; set; }
}
}

```

Теперь перейдем к странице добавление партнера. На рисунке 9 представлена страница. В листинге 5 xaml код страницы, а в листинге 6 представлен код логики работы страницы.

Мастер пол - Добавление партнера

Назад

Тип организации	Адрес
<input type="text"/>	<input type="text"/>
Название организации	ИНН
<input type="text"/>	<input type="text"/>
Директор	Рейтинг
<input type="text"/>	<input type="text"/>
Почта организации	Тип продукции
<input type="text"/>	<input type="text"/>
Номер телефона	Продукция
<input type="text"/>	<input type="text"/>
	Количество проданного товара
	<input type="text"/>

Сохранить    Назад

Рисунок 9 – Страница добавления партнера

## Листинг 5 – код xaml страницы с клиентами

```
<Page x:Class="MasterFloorAPP.Pages.AddPartner"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:local="clr-namespace:MasterFloorAPP.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="Добавление партнера">

    <Grid Background="White">
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="*/>
            <RowDefinition Height="Auto"/>
        </Grid.RowDefinitions>

        <!-- Основное содержимое в центральной части -->
        <Grid Grid.Row="1" HorizontalAlignment="Center"
              VerticalAlignment="Center" Width="600">
            <Grid.ColumnDefinitions>
                <ColumnDefinition Width="1*" />
                <ColumnDefinition Width="1*" />
            </Grid.ColumnDefinitions>

            <!-- Первая колонка -->
            <StackPanel Grid.Column="0" Margin="10">
                <TextBlock Text="Тип организации" FontSize="16" Margin="0 0 0
10"/>
                <ComboBox x:Name="Type" Height="30" FontSize="16"
VerticalContentAlignment="Center" Margin="0 0 0 10"></ComboBox>

                <TextBlock Text="Название организации" FontSize="16" Margin="0
0 0 10"/>
                <TextBox x:Name="NameOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

                <TextBlock Text="Директор" FontSize="16" Margin="0 0 0 10"/>
                <TextBox x:Name="DirectorOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

                <TextBlock Text="Почта организации" FontSize="16" Margin="0 0
0 10"/>
            </StackPanel>
        </Grid>
    </Grid>
</Page>
```

```

        <TextBox x:Name="EmailOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

        <TextBlock Text="Номер телефона" FontSize="16" Margin="0 0 0
10"/>

        <TextBox x:Name="NumberOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>
    </StackPanel>

    <!-- Вторая колонка -->
    <StackPanel Grid.Column="1" Margin="10">
        <TextBlock Text="Адрес" FontSize="16" Margin="0 0 0 10"/>
        <TextBox x:Name="AddressOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

        <TextBlock Text="ИНН" FontSize="16" Margin="0 0 0 10"/>
        <TextBox x:Name="INNOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10" MaxLength="10"/>

        <TextBlock Text="Рейтинг" FontSize="16" Margin="0 0 0 10"/>
        <TextBox x:Name="RatingOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

        <TextBlock Text="Тип продукции" FontSize="16" Margin="0 0 0
10"/>

        <ComboBox x:Name="TovariTypes" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

        <TextBlock Text="Продукция" FontSize="16" Margin="0 0 0 10"/>
        <ComboBox x:Name="TovariOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>

        <TextBlock Text="Количество проданного товара" FontSize="16"
Margin="0 0 0 10"/>
        <TextBox x:Name="QuantityOrg" Height="30"
VerticalContentAlignment="Center" Margin="0 0 0 10"/>
    </StackPanel>
</Grid>

    <!-- Нижние кнопки -->
    <StackPanel Orientation="Horizontal" HorizontalAlignment="Center"
Grid.Row="2" Margin="0, 30, 0 0">

```

```

        <Button x:Name="BtnSave" Content="Сохранить" Width="100"
Height="30" FontSize="16" Margin="0 0 10 10" Click="BtnSave_Click"
ToolTip="Нажав на кнопку, вы обновите / добавите партнера."
Style="{StaticResource Button}"/>

        <Button x:Name="BtnBack" Content="Назад" Width="100" Height="30"
FontSize="16" Margin="0 0 0 10" Click="BtnBack_Click" ToolTip="Нажав на кнопку
назад, вы вернетесь на предыдущую страницу." Style="{StaticResource Button}"/>
    </StackPanel>
</Grid>
</Page>

```

## Листинг 6 – код логики работы страницы

```

using System;
using System.IO;
using System.Linq;
using System.Text.RegularExpressions;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using System.Windows.Navigation;

namespace MasterFloorAPP.Pages
{
    public partial class AddPartner : Page
    {
        // Регулярные выражения для проверки форматов
        private static readonly Regex FIOregex = new Regex(@"^[А-ЯЁ][а-
яё]+$"); // Для ФИО директора
        private static readonly Regex EmailRegex = new
Regex(@"^[^@\s]+@[^@\s]+\.[^@\s]+$"); // Для электронной почты
        private static readonly Regex INNRegex = new Regex(@"^\d{10}$"); //
Для ИНН (10 цифр)
        private static readonly Regex QuantityRegex = new Regex(@"^\d+$");
        private static readonly Regex PhoneNumberRegex = new Regex(@"^\+?[1-
9]\d{9,14}$"); // Для номера телефона с международным кодом

        private readonly Entities db; // Контекст базы данных
        private readonly Partner currentPartner; // Текущий партнер, если
редактирование

        public AddPartner(Partner selectedPartner = null)
        {
            InitializeComponent();
            db = new Entities(); // Инициализация контекста базы данных

```

```

LoadTypes(); // Загрузка типов партнеров в комбобокс
LoadProducts();
LoadProductTypes();
currentPartner = selectedPartner; // Если передан партнер, значит
редактируем его

// Если текущий партнер не null, заполняем поля данными партнера
if (currentPartner != null)
{
    NameOrg.Text = currentPartner.Name;
    EmailOrg.Text = currentPartner.Email;
    NumberOrg.Text = currentPartner.Number;
    DirectorOrg.Text = currentPartner.Director;
    AddressOrg.Text = currentPartner.Address;
    INNOrg.Text = currentPartner.INN; // Здесь могут быть
проблемы с форматированием ИНН
    RatingOrg.Text = currentPartner.Rating.ToString();

    if (currentPartner.Type != null)
    {
        Type.SelectedValue = currentPartner.Type; // Выбираем тип
партнера
    }

    var partnerProduct = db.Partner_products
        .FirstOrDefault(pp => pp.Partner ==
currentPartner.ID);

    if (partnerProduct != null)
    {
        // Выбираем соответствующий товар и тип товара, если такой
есть
        ТовариOrg.SelectedValue = partnerProduct.Product;
        QuantityOrg.Text = partnerProduct.Quantity.ToString();

        // Найдём тип товара и установим его в комбобокс
        var product = db.Products.FirstOrDefault(p => p.ID ==
partnerProduct.Product);
        if (product != null)
        {
            var productType = db.Product_type.FirstOrDefault(pt =>
pt.ID == product.Type);
            if (productType != null)
            {

```

```

TovariTypes.SelectedValue = productType.ID; //
Устанавливаем тип товара
    }
    }
    }
}

// Метод для загрузки типов партнеров в комбобокс
private void LoadTypes()
{
    var types = db.Partner_type.ToList(); // Получаем все типы из базы
    Type.ItemsSource = types; // Привязываем типы к источнику данных
    Type.DisplayMemberPath = "Type"; // Отображаем название типа
    Type.SelectedValuePath = "ID"; // ID будет значением, которое
хранится в базе
}

private void LoadProducts()
{
    var products = db.Products.ToList();
    TovariOrg.ItemsSource = products;
    TovariOrg.DisplayMemberPath = "Name";
    TovariOrg.SelectedValuePath = "ID";
}

private void LoadProductTypes()
{
    var productTypes = db.Product_type.ToList();
    TovariTypes.ItemsSource = productTypes;
    TovariTypes.DisplayMemberPath = "Type";
    TovariTypes.SelectedValuePath = "ID";
}

private void BtnSave_Click(object sender, RoutedEventArgs e)
{
    string NameOrganization = NameOrg.Text;
    string DirectorOrganization = DirectorOrg.Text;
    string EmailOrganization = EmailOrg.Text;
    string NumberOrganization = NumberOrg.Text;
    string INNOrganization = INNOrg.Text;
    string RatingOrganization = RatingOrg.Text;
    string AddressOrganization = AddressOrg.Text;

    int? selectedType = (int?)Type.SelectedValue;

```

```
string Quantity = QuantityOrg.Text;
int? selectedProductTypeID = (int?)TovariTypes.SelectedValue;
int? selectedProductID = (int?)TovariOrg.SelectedValue;

try
{
    // Валидация данных
    if (!FIOValidation(DirectorOrganization))
    {
        MessageBox.Show(
            "ФИО директора должно содержать от 2 до 5 слов,
            начинаться с заглавной буквы и содержать только буквы кириллицы.",
            "Ошибка: Некорректное ФИО",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
        return;
    }

    if (!PhoneNumberRegex.IsMatch(NumberOrganization))
    {
        MessageBox.Show(
            "Номер телефона должен быть в международном формате,
            например: +1234567890.",
            "Ошибка: Некорректный номер телефона",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
        return;
    }

    if (!EmailRegex.IsMatch(EmailOrganization))
    {
        MessageBox.Show(
            "Электронная почта должна быть в формате
            example@domain.com.",
            "Ошибка: Некорректная электронная почта",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
        return;
    }

    if (!INNRegex.IsMatch(INNOrganization))
    {
        MessageBox.Show(
            "ИНН должен содержать ровно 10 цифр.",
```

```
        "Ошибка: Некорректный ИНН",
        MessageBoxButton.OK,
        MessageBoxImage.Warning);
    return;
}

    if (!int.TryParse(RatingOrganization, out int ratingValue) ||
ratingValue < 1 || ratingValue > 10)
    {
        MessageBox.Show(
            "Рейтинг должен быть числом от 1 до 10.",
            "Ошибка: Некорректный рейтинг",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
        return;
    }

    if (!QuantityRegex.IsMatch(Quantity))
    {
        MessageBox.Show(
            "Количество должно быть числовым значением.",
            "Ошибка: Некорректное количество",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
        return;
    }

    // Работа с партнером
    if (currentPartner != null) // Редактирование
    {
        if (MessageBox.Show(
            "Вы уверены, что хотите сохранить изменения?",
            "Подтверждение",
            MessageBoxButton.YesNo,
            MessageBoxImage.Question) != MessageBoxResult.Yes)
        {
            return;
        }

        // Обновление данных партнера
        currentPartner.Name = NameOrganization;
        currentPartner.Director = DirectorOrganization;
        currentPartner.Email = EmailOrganization;
        currentPartner.Number = NumberOrganization;
        currentPartner.INN = INNOrganization;
    }
}
```



```

        currentPartner.Address = AddressOrganization;
        currentPartner.Rating = ratingValue;
        currentPartner.Type = (int)selectedType;

        db.SaveChanges();

        int partnerId = currentPartner.ID;

        var partnerProduct = db.Partner_products
                                .FirstOrDefault(pp => pp.Partner ==
partnerId);

        if (partnerProduct != null)
        {
            partnerProduct.Product = selectedProductID ??
partnerProduct.Product;
            partnerProduct.Quantity = int.TryParse(Quantity, out
int quantityValue)
                ? quantityValue
                : partnerProduct.Quantity;
        }
        else if (selectedProductID.HasValue &&
int.TryParse(Quantity, out int quantityValue))
        {
            db.Partner_products.Add(new Partner_products
            {
                Partner = partnerId,
                Product = selectedProductID.Value,
                Quantity = quantityValue,
                SaleDATE = DateTime.Now
            });
        }

        db.SaveChanges();
        MessageBox.Show(
            "Данные партнера успешно обновлены.",
            "Успех",
            MessageBoxButton.OK,
            MessageBoxImage.Information);
    }
    else // Создание нового партнера
    {
        var newPartner = new Partner
        {
            Type = (int)selectedType,

```

```

        Name = NameOrganization,
        Director = DirectorOrganization,
        Email = EmailOrganization,
        Number = NumberOrganization,
        INN = INNOrganization,
        Address = AddressOrganization,
        Rating = ratingValue
    };

    db.Partners.Add(newPartner);
    db.SaveChanges();

    int partnerId = newPartner.ID;

    if (int.TryParse(Quantity, out int quantityValue) &&
selectedProductID.HasValue)
    {
        db.Partner_products.Add(new Partner_products
        {
            Partner = partnerId,
            Product = selectedProductID.Value,
            Quantity = quantityValue,
            SaleDATE = DateTime.Now
        });
        db.SaveChanges();

        MessageBox.Show(
            "Партнер и продукция успешно добавлены.",
            "Успех",
            MessageBoxButton.OK,
            MessageBoxImage.Information);
    }
    else
    {
        MessageBox.Show(
            "Некорректное количество продукции или продукт не
выбран.",
            "Предупреждение",
            MessageBoxButton.OK,
            MessageBoxImage.Warning);
    }
}

NavigationService?.Navigate(new Discounts());
ClearFields();

```

```

    }
    catch (Exception ex)
    {
        MessageBox.Show(
            $"Произошла ошибка: {ex.Message}\nПожалуйста, обратитесь к
администратору, если проблема повторяется.",
            "Критическая ошибка",
            MessageBoxButton.OK,
            MessageBoxImage.Error);
    }
}

private void BtnBack_Click(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack(); // Возвращаемся на предыдущую страницу
    ClearFields(); // Очищаем форму
}

// Валидация ФИО директора
private bool FIOValidation(string FIO)
{
    var splitFIO = FIO.Split(new[] { ' ' },
StringSplitOptions.RemoveEmptyEntries); // Разделяем ФИО по пробелам

    // Проверка, что ФИО состоит из 2-5 частей (имя, фамилия, и
возможно отчество)
    if (splitFIO.Length < 2 || splitFIO.Length > 5)
    {
        return false;
    }

    // Проверка каждой части ФИО на соответствие регулярному выражению
и длину
    foreach (var part in splitFIO)
    {
        if (string.IsNullOrWhiteSpace(part) || !FIOregex.IsMatch(part)
|| part.Length > 50)
        {
            return false;
        }
    }
    return true;
}

```

```

// Очистка всех полей формы
private void ClearFields()
{
    Type.SelectedItem = null;
    NameOrg.Text = string.Empty;
    DirectorOrg.Text = string.Empty;
    AddressOrg.Text = string.Empty;
    NumberOrg.Text = string.Empty;
    INNOrg.Text = string.Empty;
    RatingOrg.Text = string.Empty;
    EmailOrg.Text = string.Empty;
    ТовариTypes.SelectedItem = null;
    ТовариOrg.SelectedItem = null;
    QuantityOrg.Text = string.Empty;
}
}
}

```

Стоит отметить, что страница добавления партнера, работает также и на редактирование данных о партнере. Для того, чтобы редактировать данные о партнере, требуется два раза нажать на левую кнопку мыши в списке партнеров, после чего перенаправить на страницу добавления партнера, но уже с заполненными данными. Данный пример представлен на рисунке 10.

Тип организации	Адрес
<input type="text" value="ЗАО"/>	<input type="text" value="652050, Кемеровская область, город Юрга, ул. Лес"/>
Название организации	ИНН
<input type="text" value="База Строитель"/>	<input type="text" value="2222455179"/>
Директор	Рейтинг
<input type="text" value="Иванова Александра Ивановна"/>	<input type="text" value="7"/>
Почта организации	Тип продукции
<input type="text" value="aleksandraivanova@ml.ru"/>	<input type="text" value="Паркетная доска"/>
Номер телефона	Продукция
<input type="text" value="493 123 45 6"/>	<input type="text" value="Паркетная доска Ясень темный однополосная"/>
	Количество проданного товара
	<input type="text" value="15500"/>

Рисунок 10 – Редактирование партнера

Теперь перейдем на страницу продажи. На рисунке 11 представлен пример, как выглядит данная страница. В листинге 7 представлен код xaml для отображения элементов на странице, а в листине 8 представлен код логики работы данной страницы.

Мастер пол - Продажи

Назад

Продукция	Партнер	Количество	Дата продажи
Ламинат Дуб дымчато-белый 33 класс 12 мм	МонтажПро	50000	19.09.2023
Ламинат Дуб серый 32 класс 8 мм с фаской	МонтажПро	670000	10.11.2023
Паркетная доска Ясень темный однополосная 14 мм	МонтажПро	35000	15.04.2024
Инженерная доска Дуб Французская елка однополосная	МонтажПро	25000	12.06.2024

Рисунок 11 – Страница просмотра продаж по конкретному партнеру

Листинг 7 – xaml код страница для просмотра продаж

```
<Page x:Class="MasterFloorAPP.Pages.Sales"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:local="clr-namespace:MasterFloorAPP.Pages"
    mc:Ignorable="d"
    d:DesignHeight="450" d:DesignWidth="800"
    Title="Продажи">

    <Grid Background="White">
        <Grid.RowDefinitions>
            <RowDefinition Height="377*" />
            <RowDefinition Height="50*" />
        </Grid.RowDefinitions>
```

```

        <DataGrid Grid.Row="0" x:Name="DataGridSales"
AutoGenerateColumns="False" IsReadOnly="True">
            <DataGrid.Columns>
                <DataGridTextColumn Header="Продукция" Width="*"
Binding="{Binding Product1.Name}"/>
                <DataGridTextColumn Header="Партнер" Width="*"
Binding="{Binding Partner1.Name}"/>
                <DataGridTextColumn Header="Количество" Width="*"
Binding="{Binding Quantity}"/>
                <DataGridTextColumn Header="Дата продажи" Width="*"
Binding="{Binding SaleDATE, StringFormat='dd.MM.yyyy}'"/>
            </DataGrid.Columns>
        </DataGrid>

    </Grid>
</Page>

```

## Листинг 8 – Код логики работы страницы просмотра продаж

```

using System.Linq;
using System.Windows.Controls;

namespace MasterFloorAPP.Pages
{
    public partial class Sales : Page
    {
        private readonly Entities db;
        public Sales(int partnerID)
        {
            InitializeComponent();
            db = new Entities();

            var partnerSales = db.Partner_products.Where(p => p.Partner ==
partnerID).ToList();

            DataGridSales.ItemsSource = partnerSales;
        }
    }
}

```

Исходя из технического задания требуется прописать стили для приложения. Для этого был создан файл «Styles.xaml», который подключен в программное решение. В листинге 9 представлен код со стилем приложения.

#### Листинг 9 – Код, в котором прописаны стили для приложения

```
<ResourceDictionary
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">

    <Style x:Key="Button" TargetType="Button">
        <Setter Property="Width" Value="170"/>
        <Setter Property="Height" Value="30"/>
        <Setter Property="Background" Value="#67BA80"/>
        <Setter Property="Foreground" Value="#F4E8D3"/>
        <Setter Property="FontFamily" Value="Segoe UI"/>
        <Setter Property="FontWeight" Value="DemiBold"/>
        <Setter Property="FontSize" Value="14"/>
    </Style>

</ResourceDictionary>
```