ПРИДНЕСТРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

им. Т.Г. Шевченко

Физико-математический факультет

Кафедра прикладной математики и информатики

|  |  |
| --- | --- |
|  | ПРОВЕРЕНО  групповой руководитель практики  преподаватель кафедры ПМиИ  Горб Евгений Александрович  «\_\_»\_\_\_\_\_\_\_\_ 20\_\_\_г. |

**О Т Ч Ё Т**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ**

СТУДЕНТКИ Липчанской Анны Владимировны

|  |  |
| --- | --- |
| Факультет: | физико-математический |
| Направление подготовки: | Прикладная математика и информатика |
| Профиль: | Базы данных |
| Тема: | Даталогическое проектирование. |
| Группа: | 303 |
| Номер лабораторной работы: | 4 |
| Номер варианта: | 1 |
|  |

Тирасполь, 2020

**Вариант 1. База данных «Платный прием в поликлинике»**

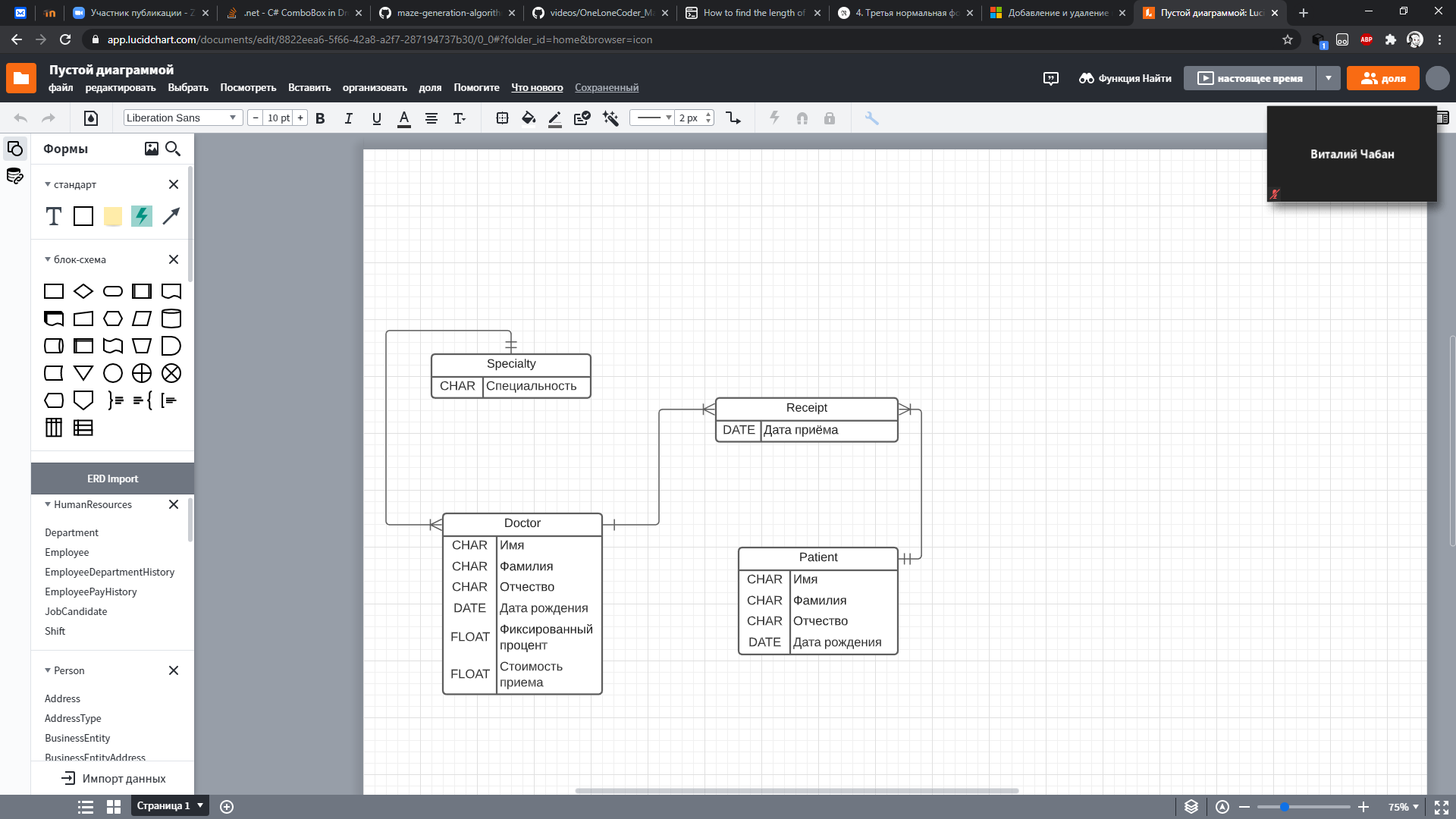
Платный прием пациентов проводится врачами разных специальностей (хирург, терапевт, кардиолог, офтальмолог и т.д.). При оформлении приема должна быть сформирована квитанция об оплате приема, в которой указывается информация о пациенте, о враче, который консультирует пациента, о стоимости приема, о дате приема.

Пациент оплачивает за прием некоторую сумму, которая устанавливается персонально для каждого врача. За каждый прием врачу отчисляется фиксированный процент от стоимости приема. Процент отчисления от стоимости приема на зарплату врача также устанавливается персонально для каждого врача.

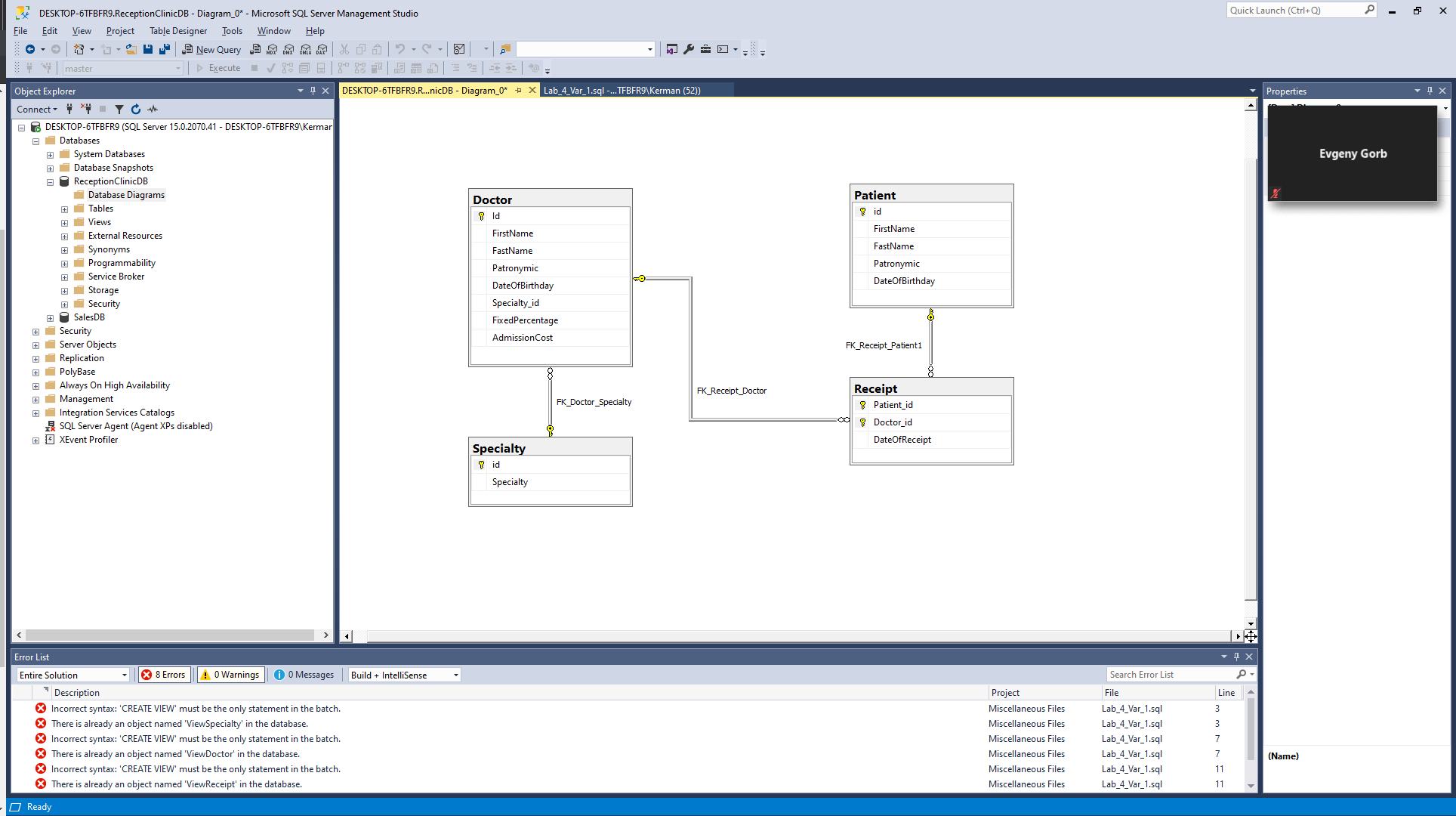
Размер начисляемой врачу заработной платы за каждый прием вычисляется по формуле:

Зарплата = Стоимость приема \* Процент отчисления на зарплату.

Из этой суммы вычитается подоходный налог, составляющий 13% от начисленной зарплаты.

Инфологической модели предметной области: 

Преобразуем инфологическую модель в даталогическую. В результате получим следующую даталогическую модель:



**SQL запросы для создания БД и таблиц вместе со всеми ограничениями и индексами**

USE [ReceptionClinicDB]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Specialty] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Specialty](

[id] [int] IDENTITY(1,1) NOT NULL,

[Specialty] [nvarchar](50) NOT NULL,

CONSTRAINT [PK\_Specialty] 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: View [dbo].[ViewSpecialty] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[ViewSpecialty]

AS SELECT id, Specialty

FROM [dbo].Specialty

GO

/\*\*\*\*\*\* Object: Table [dbo].[Doctor] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Doctor](

[Id] [int] IDENTITY(1,1) NOT NULL,

[FirstName] [nvarchar](50) NOT NULL,

[FastName] [nvarchar](50) NOT NULL,

[Patronymic] [nvarchar](50) NOT NULL,

[DateOfBirthday] [date] NOT NULL,

[Specialty\_id] [int] NOT NULL,

[FixedPercentage] [float] NOT NULL,

[AdmissionСost] [money] NOT NULL,

CONSTRAINT [PK\_Doctor] 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: View [dbo].[ViewDoctor] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[ViewDoctor]

AS SELECT id, FirstName, FastName, Patronymic, DateOfBirthday, AdmissionСost, FixedPercentage, Specialty\_id

FROM [dbo].Doctor

GO

/\*\*\*\*\*\* Object: Table [dbo].[Receipt] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Receipt](

[Patient\_id] [int] NOT NULL,

[Doctor\_id] [int] NOT NULL,

[DateOfReceipt] [date] NOT NULL,

CONSTRAINT [IX\_PatientDoctor] PRIMARY KEY CLUSTERED

(

[Patient\_id] ASC,

[Doctor\_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: View [dbo].[ViewReceipt] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[ViewReceipt]

AS SELECT Patient\_id, Doctor\_id, DateOfReceipt

FROM [dbo].Receipt

GO

/\*\*\*\*\*\* Object: Table [dbo].[Patient] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Patient](

[id] [int] IDENTITY(1,1) NOT NULL,

[FirstName] [varchar](50) NOT NULL,

[FastName] [varchar](50) NOT NULL,

[Patronymic] [nvarchar](50) NOT NULL,

[DateOfBirthday] [date] NOT NULL,

CONSTRAINT [PK\_Patient] 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: View [dbo].[ViewPatient] Script Date: 11/25/2020 12:13:22 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE VIEW [dbo].[ViewPatient]

AS SELECT id, FirstName, FastName, Patronymic, DateOfBirthday

FROM [dbo].Patient

GO

ALTER TABLE [dbo].[Doctor] WITH CHECK ADD CONSTRAINT [FK\_Doctor\_Specialty] FOREIGN KEY([Specialty\_id])

REFERENCES [dbo].[Specialty] ([id])

GO

ALTER TABLE [dbo].[Doctor] CHECK CONSTRAINT [FK\_Doctor\_Specialty]

GO

ALTER TABLE [dbo].[Receipt] WITH CHECK ADD CONSTRAINT [FK\_Receipt\_Doctor] FOREIGN KEY([Doctor\_id])

REFERENCES [dbo].[Doctor] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Receipt] CHECK CONSTRAINT [FK\_Receipt\_Doctor]

GO

ALTER TABLE [dbo].[Receipt] WITH CHECK ADD CONSTRAINT [FK\_Receipt\_Patient1] FOREIGN KEY([Patient\_id])

REFERENCES [dbo].[Patient] ([id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Receipt] CHECK CONSTRAINT [FK\_Receipt\_Patient1]

GO

ALTER TABLE [dbo].[Doctor] WITH CHECK ADD CONSTRAINT [CK\_Doctor\_DateOfBirthday] CHECK (([DateOfBirthday]<getdate() AND datediff(year,[DateOfBirthday],getdate())<=(100)))

GO

ALTER TABLE [dbo].[Doctor] CHECK CONSTRAINT [CK\_Doctor\_DateOfBirthday]

GO

ALTER TABLE [dbo].[Patient] WITH CHECK ADD CONSTRAINT [CK\_DateOfBirthday] CHECK (([DateOfBirthday]<getdate() AND datediff(year,[DateOfBirthday],getdate())<=(100)))

GO

ALTER TABLE [dbo].[Patient] CHECK CONSTRAINT [CK\_DateOfBirthday]

GO

**SQL запросы для создания представлений**

USE [ReceptionClinicDB]

CREATE VIEW [dbo].ViewSpecialty

AS SELECT id, Specialty

FROM [dbo].Specialty

CREATE VIEW [dbo].ViewDoctor

AS SELECT id, FirstName, FastName, Patronymic, DateOfBirthday, AdmissionСost, FixedPercentage, Specialty\_id

FROM [dbo].Doctor

CREATE VIEW [dbo].ViewReceipt

AS SELECT Patient\_id, Doctor\_id, DateOfReceipt

FROM [dbo].Receipt

CREATE VIEW [dbo].ViewPatient

AS SELECT id, FirstName, FastName, Patronymic, DateOfBirthday

FROM [dbo].Patient

«\_\_» \_\_\_\_\_\_\_\_ 20\_\_\_г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Липчанская А.В./