**Лабораторная работа № 3**

**Логическая модель данных. Правила целостности.**

**Задание**

1.На основе концептуальной модели предметной области построить логическую модель данных. Использовать реляционную модель. Модель реализовать в среде MS SQL Server. Построить схему (диаграмму) базы данных.

2.Спроектировать и реализовать необходимые правила (ограничения) целостности.

3.В базу необходимо ввести некоторое количество тестовых данных. В каждой таблице должно быть не менее 7 записей. В ассоциациях, не менее 20 записей.

4.В пояснительную записку надо включить:

* логическую модель в виде схемы (диаграммы) базы данных;
* логическую модель в виде текста с подробным описанием правил (ограничений) целостности;
* анализ модели на соответствие принципам нормализации.

**Учебный пример разработки базы данных**

**Построение логической модели данных**

Логическая модель строится на основе концептуальной модели.

Из всех возможных видов логических моделей используем реляционную логическую модель. Это самый распространенный вид логических моделей.

Необходимо выполнить следующие шаги процедуры проектирования логической модели.

1. Представить каждую сущность как таблицу базы данных и специфицировать первичный ключ этой таблицы. При необходимости специфицировать внешний ключ и ограничения на внешний ключ этой таблицы.

2. Представить каждую ассоциацию как таблицу. Использовать в этой таблице внешние ключи для идентификации участников ассоциации и специфицировать ограничения, связанные с каждым из этих внешних ключей.

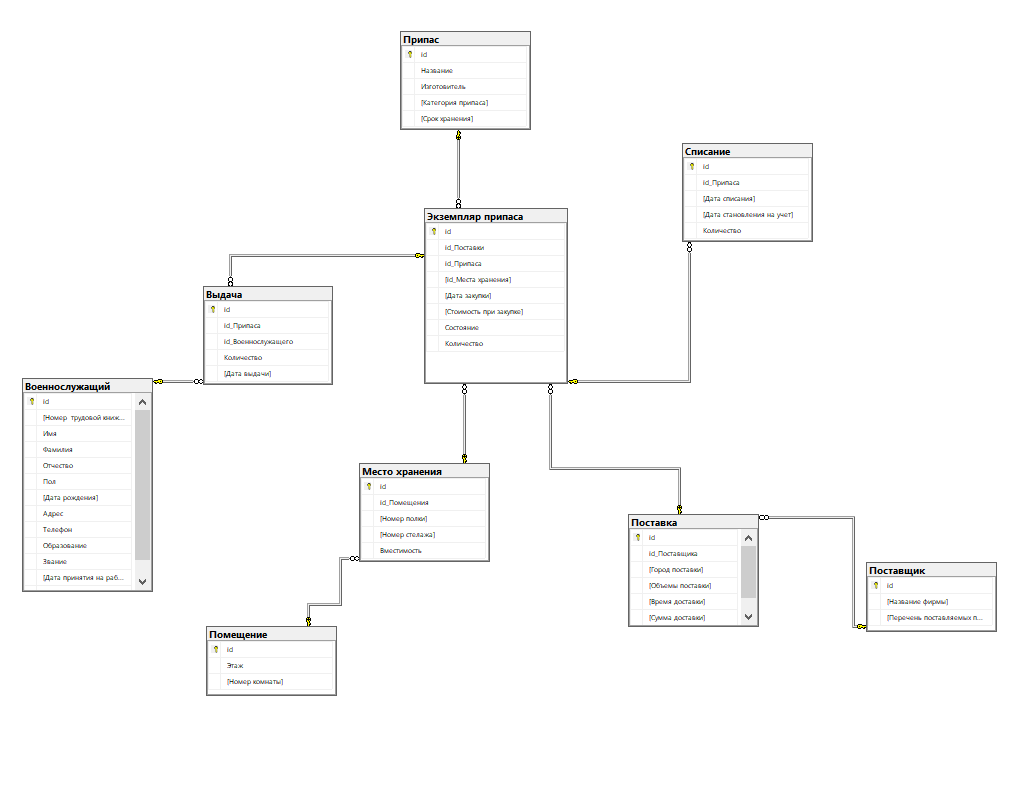
5. Представить каждый атрибут сущности как поле в таблице, представляющей сущность.

6. Для того чтобы исключить в проекте непреднамеренные нарушения каких-либо принципов нормализации, выполнить процедуру проверки отношений на соответствие 3-й нормальной форме (3НФ) или нормальной форме Бойса-Кодда (НФБК), описание процедуры см. в курсе лекций.

7. Если в процессе проверки было выявлено несоответствие и произведено разделение каких-либо таблиц, то следует модифицировать концептуальную модель базы данных и повторить перечисленные шаги.

8. Кроме уже введенных, ввести необходимые правила целостности, в частности правила доменной целостности. Дать краткое описание полученных таблиц и их полей.

Для нашей учебной базы данных «Военный склад» логическая модель в виде схемы (диаграммы) базы данных будет выглядеть так:



Логическая модель в виде текста с подробным описанием правил(ограничений) целостности описывается так:

CREATE TABLE [dbo].[Военнослужащий](

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

[Номер трудовой книжки] [varchar](50) NOT NULL,

[Имя] [varchar](50) NOT NULL,

[Фамилия] [varchar](50) NOT NULL,

[Отчество] [varchar](50) NULL,

[Пол] [varchar](10) NOT NULL,

[Дата рождения] [date] NOT NULL,

[Адрес] [varchar](100) NOT NULL,

[Телефон] [varchar](50) NOT NULL,

[Образование] [varchar](100) NOT NULL,

[Звание] [varchar](50) NOT NULL,

[Дата принятия на работу] [date] NOT NULL,

[Дата увольнения] [date] NULL,

CONSTRAINT [PK\_Военнослужащий] 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

ALTER TABLE [dbo].[Военнослужащий] ADD CONSTRAINT [DF\_Военнослужащий\_Дата увольнения] DEFAULT (NULL) FOR [Дата увольнения]

GO

ALTER TABLE [dbo].[Военнослужащий] WITH CHECK ADD CONSTRAINT [CK\_Военнослужащий] CHECK (([Пол]='муж' OR [Пол]='жен'))

GO

ALTER TABLE [dbo].[Военнослужащий] CHECK CONSTRAINT [CK\_Военнослужащий]

GO

CREATE TABLE [dbo].[Выдача](

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

[id\_Припаса] [int] NOT NULL,

[id\_Военнослужащего] [int] NOT NULL,

[Количество] [int] NOT NULL,

[Дата выдачи] [date] NOT NULL,

CONSTRAINT [PK\_Выдача] 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

ALTER TABLE [dbo].[Выдача] ADD CONSTRAINT [DF\_Выдача\_Количество] DEFAULT ((1)) FOR [Количество]

GO

ALTER TABLE [dbo].[Выдача] WITH CHECK ADD CONSTRAINT [FK\_Выдача\_Военнослужащий] FOREIGN KEY([id\_Военнослужащего])

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

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Выдача] CHECK CONSTRAINT [FK\_Выдача\_Военнослужащий]

GO

ALTER TABLE [dbo].[Выдача] WITH CHECK ADD CONSTRAINT [FK\_Выдача\_Экземпляр припаса] FOREIGN KEY([id\_Припаса])

REFERENCES [dbo].[Экземпляр припаса] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Выдача] CHECK CONSTRAINT [FK\_Выдача\_Экземпляр припаса]

GO

CREATE TABLE [dbo].[Место хранения](

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

[id\_Помещения] [int] NOT NULL,

[Номер полки] [int] NOT NULL,

[Номер стелажа] [int] NOT NULL,

[Вместимость] [int] NOT NULL,

CONSTRAINT [PK\_Место хранения] 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

ALTER TABLE [dbo].[Место хранения] ADD CONSTRAINT [DF\_Место хранения\_Вместимость] DEFAULT ((50)) FOR [Вместимость]

GO

ALTER TABLE [dbo].[Место хранения] WITH CHECK ADD CONSTRAINT [FK\_Место хранения\_Помещение] FOREIGN KEY([id\_Помещения])

REFERENCES [dbo].[Помещение] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Место хранения] CHECK CONSTRAINT [FK\_Место хранения\_Помещение]

GO

ALTER TABLE [dbo].[Место хранения] WITH CHECK ADD CONSTRAINT [CK\_Место хранения] CHECK (([Номер полки]>(0)))

GO

ALTER TABLE [dbo].[Место хранения] CHECK CONSTRAINT [CK\_Место хранения]

GO

ALTER TABLE [dbo].[Место хранения] WITH CHECK ADD CONSTRAINT [CK\_Место хранения\_1] CHECK (([Номер стелажа]>(0)))

GO

ALTER TABLE [dbo].[Место хранения] CHECK CONSTRAINT [CK\_Место хранения\_1]

GO

CREATE TABLE [dbo].[Помещение](

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

[Этаж] [int] NOT NULL,

[Номер комнаты] [int] NOT NULL,

CONSTRAINT [PK\_Помещение] 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

ALTER TABLE [dbo].[Помещение] WITH CHECK ADD CONSTRAINT [CK\_Помещение] CHECK (([Этаж]>(0)))

GO

ALTER TABLE [dbo].[Помещение] CHECK CONSTRAINT [CK\_Помещение]

GO

ALTER TABLE [dbo].[Помещение] WITH CHECK ADD CONSTRAINT [CK\_Помещение\_1] CHECK (([Номер комнаты]>(0)))

GO

ALTER TABLE [dbo].[Помещение] CHECK CONSTRAINT [CK\_Помещение\_1]

GO

CREATE TABLE [dbo].[Поставка](

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

[id\_Поставщика] [int] NOT NULL,

[Город поставки] [varchar](50) NOT NULL,

[Объемы поставки] [int] NOT NULL,

[Время доставки] [datetime] NOT NULL,

[Сумма доставки] [int] NOT NULL,

CONSTRAINT [PK\_Поставка] 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

ALTER TABLE [dbo].[Поставка] WITH CHECK ADD CONSTRAINT [FK\_Поставка\_Поставщик] FOREIGN KEY([id\_Поставщика])

REFERENCES [dbo].[Поставщик] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Поставка] CHECK CONSTRAINT [FK\_Поставка\_Поставщик]

GO

CREATE TABLE [dbo].[Поставщик](

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

[Название фирмы] [varchar](50) NOT NULL,

[Перечень поставляемых предметов] [varchar](1000) NOT NULL,

CONSTRAINT [PK\_Поставщик] 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

CREATE TABLE [dbo].[Припас](

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

[Название] [varchar](100) NOT NULL,

[Изготовитель] [varchar](100) NOT NULL,

[Категория припаса] [varchar](100) NOT NULL,

[Срок хранения] [date] NOT NULL,

CONSTRAINT [PK\_Припас] 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

CREATE TABLE [dbo].[Списание](

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

[id\_Припаса] [int] NOT NULL,

[Дата списания] [date] NOT NULL,

[Дата становления на учет] [date] NOT NULL,

[Количество] [int] NOT NULL,

CONSTRAINT [PK\_Списание] 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

ALTER TABLE [dbo].[Списание] ADD CONSTRAINT [DF\_Списание\_Дата списания] DEFAULT (NULL) FOR [Дата списания]

GO

ALTER TABLE [dbo].[Списание] ADD CONSTRAINT [DF\_Списание\_Количество] DEFAULT ((1)) FOR [Количество]

GO

ALTER TABLE [dbo].[Списание] WITH CHECK ADD CONSTRAINT [FK\_Списание\_Экземпляр припаса] FOREIGN KEY([id\_Припаса])

REFERENCES [dbo].[Экземпляр припаса] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Списание] CHECK CONSTRAINT [FK\_Списание\_Экземпляр припаса]

GO

ALTER TABLE [dbo].[Списание] WITH CHECK ADD CONSTRAINT [CK\_Списание] CHECK (([Количество]>(0)))

GO

ALTER TABLE [dbo].[Списание] CHECK CONSTRAINT [CK\_Списание]

GO

CREATE TABLE [dbo].[Экземпляр припаса](

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

[id\_Поставки] [int] NOT NULL,

[id\_Припаса] [int] NOT NULL,

[id\_Места хранения] [int] NOT NULL,

[Дата закупки] [date] NOT NULL,

[Стоимость при закупке] [float] NOT NULL,

[Состояние] [varchar](50) NOT NULL,

[Количество] [int] NOT NULL,

CONSTRAINT [PK\_Экземпляр Припаса] 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

ALTER TABLE [dbo].[Экземпляр припаса] ADD CONSTRAINT [DF\_Экземпляр припаса\_Стоимость при закупке] DEFAULT ((0)) FOR [Стоимость при закупке]

GO

ALTER TABLE [dbo].[Экземпляр припаса] ADD CONSTRAINT [DF\_Экземпляр припаса\_Состояние] DEFAULT ('хорошее') FOR [Состояние]

GO

ALTER TABLE [dbo].[Экземпляр припаса] ADD CONSTRAINT [DF\_Экземпляр припаса\_Количество] DEFAULT ((1)) FOR [Количество]

GO

ALTER TABLE [dbo].[Экземпляр припаса] WITH CHECK ADD CONSTRAINT [FK\_Экземпляр припаса\_Место хранения] FOREIGN KEY([id\_Места хранения])

REFERENCES [dbo].[Место хранения] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Экземпляр припаса] CHECK CONSTRAINT [FK\_Экземпляр припаса\_Место хранения]

GO

ALTER TABLE [dbo].[Экземпляр припаса] WITH CHECK ADD CONSTRAINT [FK\_Экземпляр припаса\_Поставка] FOREIGN KEY([id\_Поставки])

REFERENCES [dbo].[Поставка] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Экземпляр припаса] CHECK CONSTRAINT [FK\_Экземпляр припаса\_Поставка]

GO

ALTER TABLE [dbo].[Экземпляр припаса] WITH CHECK ADD CONSTRAINT [FK\_Экземпляр припаса\_Припас] FOREIGN KEY([id\_Припаса])

REFERENCES [dbo].[Припас] ([id])

ON UPDATE CASCADE

GO

ALTER TABLE [dbo].[Экземпляр припаса] CHECK CONSTRAINT [FK\_Экземпляр припаса\_Припас]

GO

ALTER TABLE [dbo].[Экземпляр припаса] WITH CHECK ADD CONSTRAINT [CK\_Экземпляр припаса] CHECK (([Стоимость при закупке]>(0)))

GO

ALTER TABLE [dbo].[Экземпляр припаса] CHECK CONSTRAINT [CK\_Экземпляр припаса]

GO