

Министерство науки и высшего образования РФ
Федеральное государственное автономное образовательное учреждение
высшего образования
Национальный исследовательский университет ИТМО

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2
“АНАЛИЗ ДАННЫХ. ПОСТРОЕНИЕ ИНФОЛОГИЧЕСКОЙ МОДЕЛИ
ДАННЫХ БД”
по дисциплине “Проектирование и реализация баз данных”

Обучающийся: Дидык Глеб Олегович

Факультет: Прикладная информатика

Группа: К3239

Направление подготовки: 09.03.03 Прикладная информатика

Образовательная программа: Мобильные и сетевые технологии

Преподаватель: Говорова Марина Михайловна

Санкт-Петербург

2025/2026

Цель работы: овладеть практическими навыками проведения анализа данных системы и построения инфологической модели данных БД методом «сущность-связь».

Практическое задание:

1. Проанализировать предметную область согласно варианту задания.
2. Выполнить инфологическое моделирование базы данных по заданной предметной области с использованием метода ER-диаграмм («сущность-связь») в комбинированной нотации Питера Чена - Кириллова (задание 1.1 варианта).
3. Реализовать разработанную ИЛМ в нотации IDEF1X.

Индивидуальное задание: Вариант 6. БД "Поссажир"

Описание предметной области: Информационная система служит для продажи железнодорожных билетов. Билеты могут продаваться на текущие сутки или предварительно (не более чем за 45 суток). Цена билета при предварительной продаже снижается на 5%. Билет может быть приобретен в кассе или онлайн. Если билет приобретен в кассе, необходимо знать, в какой. Для каждой кассы известны номер и адрес. Кассы могут располагаться в различных населенных пунктах.

Поезда курсируют по расписанию, но могут назначаться дополнительные поезда на заданный период или определенные даты.

По всем промежуточным остановкам на маршруте известны название, тип населенного пункта, время прибытия, отправления, время стоянки. На каждый рейс по расписанию формируется состав из вагонов определенного типа (общий, плацкарт, купе, СВ, люкс). Все вагоны в составе на рейс имеют номера.

Необходимо учитывать, что местом посадки и высадки пассажира могут быть промежуточные пункты по маршруту.

БД должна содержать следующий минимальный набор сведений: Номер поезда. Название поезда. Тип поезда. Пункт назначения. Пункт назначения для проданного билета. Номер вагона. Тип вагона. Количество мест в вагоне. Цена билета. Дата отправления. Дата прибытия. Дата прибытия для пункта назначения проданного билета. Время отправления. Номер вагона в поезде. Номер билета. Место. Тип места. Фамилия пассажира. Имя пассажира.

Отчество пассажира. Паспортные данные. Дополните состав атрибутов на основе анализа предметной области.

Задание 1.1 (ЛР 2 БД). Выполните инфологическое моделирование базы данных системы. (Ограничения задать самостоятельно.)

Задание 1.2. Создайте логическую модель БД, используя ИЛМ (задание 1.1). Используйте необходимые средства поддержки целостности данных в СУБД.

Выполнение:

1. Название создаваемой БД: “Спортивный клуб”
2. Состав реквизитов сущностей:
 - Билет (ID, ID_Вагон, ID_Рейс, ID_Касса, ID_Станция, Начальная станция, Конечная станция, Цена, Дата и время продажи, Номер места)
 - Касса(ID, Номер, Адрес)
 - Пассажир(ID, ID_Паспорт, Отчество, Имя, Фамилия)
 - Паспорт (Серия, Номер, Дата выдачи)
 - Вагон (ID, Количество мест, Номер, Тип)
 - Поезд_Вагон (ID_Вагон, ID_Поезд)
 - Поезд (ID_Категории, Номер, Тип)
 - Рейс (ID, ID_Маршрут, ID_Поезд, Номер, Тип, Дата и время прибытия на конечную станцию, Дата и время отправления от начальной станции)
 - Маршрут (ID, Номер, Начальная станция, Конечная станция)
 - Маршрут_Станция(ID_Маршрут, ID_Станция, Порядок, Стоянка, Расстояние)
 - Станция (ID, Название, Населенный пункт)

```

    erDiagram
        ticket_office ||--o{ ticket : "has"
        ticket ||--o{ passenger : "has"
        ticket ||--o{ trip : "has"
        passenger ||--o{ passport : "has"
        trip ||--o{ train : "has"
        trip ||--o{ route : "has"
        train ||--o{ train_carriage : "has"
        train_carriage ||--o{ carriage : "has"
        route ||--o{ route_station : "has"
        route_station ||--o{ station : "has"

        ticket_office {
            string id_SERIAL PK
            string address VARCHAR(255)
            string number VARCHAR(20)
        }

        ticket {
            string id_SERIAL PK
            int id_station FK
            int id_ticket_office FK
            int id_passenger FK
            int id_trip FK
            int id_carriage FK
            int departure_station
            int arrival_station
            decimal ticket_price DECIMAL(10,2)
            datetime sale_datetime
            int seat_number
        }

        passenger {
            string id_SERIAL PK
            int id_passport FK
            string last_name VARCHAR(50)
            string first_name VARCHAR(50)
            string middle_name VARCHAR(50)
        }

        passport {
            string id_SERIAL PK
            string series VARCHAR(10)
            string number VARCHAR(20)
            date issue_date
        }

        trip {
            string id_SERIAL PK
            int id_route FK
            int id_train FK
            datetime departure_datetime
            datetime arrival_datetime
        }

        train {
            string id_SERIAL PK
            string number VARCHAR(10)
            string type VARCHAR(20)
        }

        train_carriage {
            int id_train FK
            int id_carriage FK
        }

        carriage {
            string id_SERIAL PK
            string number VARCHAR(10)
            string type VARCHAR(20)
            int seats_count
        }

        route {
            string id_SERIAL PK
            string number VARCHAR(20)
            int start_station
            int end_station
        }

        route_station {
            int id_route FK
            int id_station FK
            int station_order
            int stop_duration
            int distance_from_start
        }

        station {
            string id_SERIAL PK
            string name VARCHAR(100)
            string city VARCHAR(100)
        }
  
```

Наименование атрибута	Тип	Первичный ключ		Внешний ключ	Обязательность	Ограничения целостности
		Собственный атрибут	Внешний ключ			
ticket						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
id_station	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности station
<u>id_passenger</u>	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности passenger
<u>id_trip</u>	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности trip
<u>id_carriage</u>	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности carriage
departure station	INTEGER				+	NOT NULL
arrival station	INTEGER				+	NOT NULL
ticket_price	DECIMAL(10,2)				+	NOT NULL

sale_datetime	DATETIME				+	NOT NULL
seat_number	INTEGER				+	NOT NULL
ticket_Office						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
address	DATE				+	NOT NULL
number	TIME				+	NOT NULL
passport						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
series	VARCHAR(10)				+	NOT NULL
number	VARCHAR(20)				+	NOT NULL
IssueDate	DATE				+	NOT NULL
passenger						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
id_passport				+	+	NOT NULL, UNIQUE, Значение соответствует первичному ключу сущности passport
last_name	VARCHAR(50)				+	NOT NULL
first_name	VARCHAR(50)				+	NOT NULL
middle_name	VARCHAR(50)				-	NULL
trip						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения

<u>id_route</u>	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности route
<u>Id_train</u>	INTEGER			+	+	NOT NULL, Значение соответствует первичному ключу сущности train
departure_datetime	DATETIME				+	NOT NULL
arrival_datetime	DATETIME				+	NOT NULL
train						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
number	VARCHAR(10)				+	NOT NULL
type	VARCHAR(20)				+	NOT NULL
Train Carriage						
<u>id_train</u>	INTEGER		+	+	+	NOT NULL, Значение соответствует первичному ключу сущности train
<u>id_carriage</u>	INTEGER		+	+	+	NOT NULL, Значение соответствует первичному ключу сущности carriage

carriage						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
number	VARCHAR(10)				+	NOT NULL
type	VARCHAR(20)				+	NOT NULL
seats count	INTEGER				+	NOT NULL
route						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения
number	VARCHAR(20)				+	NOT NULL
start_station	INTEGER				+	NOT NULL
end_station	INTEGER				+	NOT NULL
route_station						
<u>id_route</u>	INTEGER		+	+	+	NOT NULL, Значение соответствует первичному ключу сущности <u>route</u>
<u>id_station</u>	INTEGER		+	+	+	NOT NULL, Значение соответствует первичному ключу сущности <u>station</u>
station_order	INTEGER				+	NOT NULL
stop_duration	INTEGER					
distance_from_start	INTEGER					
station						
id	INTEGER	+			+	Уникален, необходимо обеспечить автоматическую генерацию значения

name	VARCHAR(10 0)				+	NOT NULL
city	VARCHAR(10 0)				+	NOT NULL

Вывод:

В ходе выполнения заданий я овладел практическими навыками проведения анализа данных системы и построения инфологической модели данных БД методом «сущность-связь», изучены основные нотации, используемы при создании инфологических моделей баз данных, для оформления которых использовал BD.drawio