

Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики

Лабораторная работа № 2

**«Анализ данных. Построение инфологической
модели данных БД»**

Выполнил: Хайрнасов Андрей Константинович

Группа: К3241

Преподаватель: Говорова Марина Михайловна

Санкт – Петербург

2021

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

Оборудование: компьютерный класс, мультимедийный проектор.

Программное обеспечение: CA ERwin Data Modeler (или аналог), Draw.io, ZOOM.

Практическое задание:

Проанализировать предметную область согласно варианту задания.

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).

Реализовать разработанную ИЛМ в нотации IDEF1X.

Ход работы:

1. Название БД

Вариант 7. «Курсы»

Описание предметной области: Подразделение занимается организацией внебюджетного образования. Имеется несколько типов краткосрочных курсов, предназначенных для определенных специальностей, связанных с программным обеспечением ИТ. Каждый тип курсов имеет определенную длительность и свой перечень изучаемых дисциплин. На каждую программу может быть набрано несколько групп обучающихся. По каждой дисциплине могут проводиться лекционные и лабораторные занятия. Подразделение обеспечивает следующие ресурсы: учебные классы, лекционные аудитории и преподавателей. Необходимо составить расписание занятий.

БД должна содержать следующий минимальный набор сведений: Фамилия слушателя. Имя слушателя. Паспортные данные. Контакты. Код программы.

Программа. Тип программы. Объем часов. Номер группы. максимальное количество человек в группе (для набора). Дата начала обучения. Дата окончания обучения. Название дисциплины. Количество часов. Дата занятий. Номер пары. Номер аудитории. Тип аудитории. Адрес площадки. Вид занятий (лекционные, практические или лабораторные). Фамилия преподавателя. Имя и отчество преподавателя. Должность преподавателя. Дисциплины, которые может вести преподаватель.

2. Состав реквизитов сущностей:

- a) **Программа** (Идентификатор программы, название, дата начала дата конца, идентификатор типа программы)
- b) **Тип программы**(Идентификатор типа программы, название, квалификация, тип документа)
- c) **Дисциплины** (Идентификатор дисциплины, название дисциплины, лекционные часы, лабораторные часы, практические часы)
- d) **Группа** (Идентификатор группы, номер группы, вместимость, идентификатор программы)
- e) **Слушатель** (Номер паспорта, контакты, имя слушателя, фамилия слушателя)
- f) **Обучение** (Номер паспорта, идентификатор группы, статус, номер документа об обучении)
- g) **Занятие** (Идентификатор аудитории, идентификатор дисциплины, идентификатор группы, табельный номер, дата занятия, статус занятия, номер пары, тип занятия)
- h) **Аудитория** (Идентификатор аудитории, тип аудитории, адрес, номер аудитории)
- i) **Преподаватель** (Табельный номер, фамилия, имя, отчество, идентификатор должности)
- j) **Должность** (Идентификатор должности, название)

3. Схема инфологической модели данных БД в нотации Питера Чена (рис. 1):

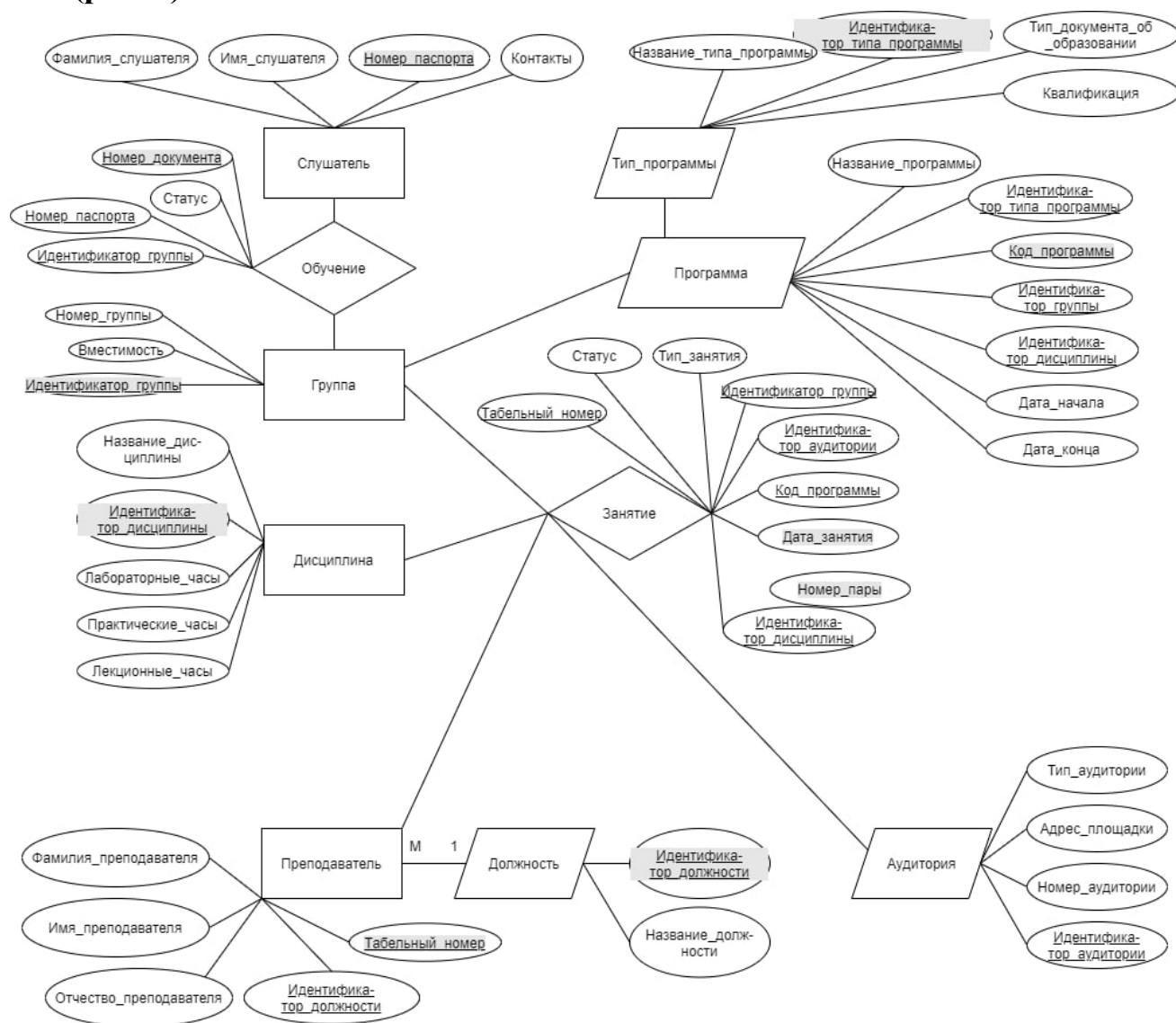


Рисунок 1

4. Схема инфологической модели данных БД в нотации IDEF1X (рис.2):

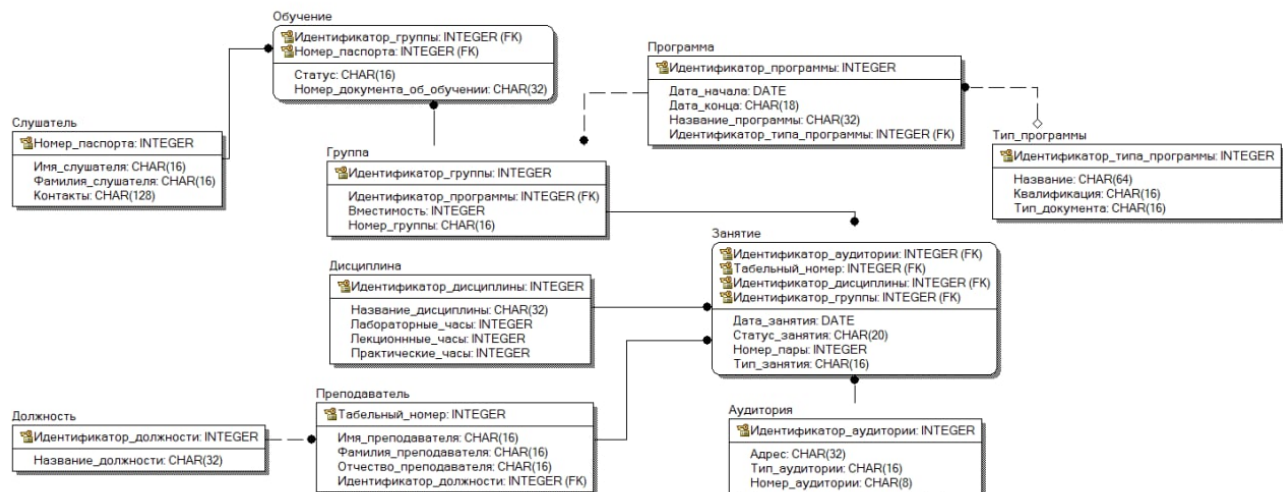


Рисунок 2

5. Описание атрибутов сущностей и ограничений на данные:

Наименование атрибута	Тип	Первичный ключ		Внешний ключ	Обязательность	Ограничения целостности
		Собственный атрибут	Внешний ключ			
Сущность “Программа”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Имя	CHAR(32)				+	
Идентификатор Типа программы	INTEGER			+	+	Внешний ключ на Тип

						программы
Идентификатор Группы	INTEGER			+	+	Внешний ключ на Группу
Дата начала	DATE				+	
Дата конца	DATE				+	
Сущность “Занятие”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Идентификатор Дисциплины	INTEGER			+	+	Внешний ключ на Дисциплину
Идентификатор Аудитории	INTEGER			+	+	Внешний ключ на Аудиторию
Дата занятия	DATE				+	
Номер пары	INTEGER				+	
Тип занятия	CHAR(16)				+	Выбор типа из Лекция, Практика
Статус	CHAR(20)				+	
Табельный номер	INTEGER				+	Внешний ключ на Преподавателя
Сущность “Аудитория”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Адрес	CHAR(32)				+	Адрес здания
Номер аудитории	CHAR(8)				+	
Тип аудитории	CHAR(16)				+	
Сущность “Тип программы”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Название	CHAR(64)				+	

Тип документа об образовании	CHAR(16)				+	
Квалификация	CHAR(16)				+	
Сущность “Слушатель”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Имя	CHAR(16)				+	
Фамилия	CHAR(16)				+	
Номер паспорта	INTEGER				+	
Контакты	CHAR(128)				+	
Сущность “Группа”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Номер группы	CHAR(16)				+	
Вместимость	INTEGER				+	
Сущность “Обучение”						
Идентификатор Группы	INTEGER		+	+	+	Unique
Идентификатор Слушателя	INTEGER		+	+	+	
Статус	CHAR(16)					
Номер документа	CHAR(32)				+	
Сущность “Дисциплина”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Название	CHAR(32)				+	
Лабораторные часы	INTEGER				+	
Практические часы	INTEGER				+	
Лекционные часы	INTEGER				+	
Сущность “Должность”						
Идентификатор	INTEGER	+			+	Unique, Autoincrement
Название	CHAR(32)				+	

Сущность “Преподаватель”						
Табельный номер	INTEGER	+			+	Unique, Autoincrement
Имя	CHAR(16)				+	
Фамилия	CHAR(16)				+	
Отчество	CHAR(16)				+	
Идентификатор Должности	INTEGER			+	+	Внешний ключ на Должность

Вывод:

В ходе выполнения лабораторной работы было составлено представление об учете выполнения проектов, построена инфологическая модель базы данных в комбинированной нотации Питера Чена – Кириллова, отражающая эту работу (в упрощенном виде), а также сделана реализация данной модели в нотации IDEF1X. Нотации Питера Чена – Кириллова излишне перегружена и при условии, что у нас имеется около 20 таблиц по 20 атрибутов в каждой, диаграмма станет сложной для прочтения, а при увеличении объемов информации станет вообще непонятной. IDEF1X более простой для чтения и среды для разработчиков (прим, DataGrip) позволяют легко построить подобную нотацию по существующей базе данных.