

Введение:

- 1) описать, зачем нужна БД;
- 2) указать цель вашей работы;
- 3) описать современные СУБД (несколько).

Глава 1. Проектирование базы данных

2.1 Анализ предметной области

Для проектирования базы данных нужно тщательно проанализировать предметную область, в которой будет работать информационная система, с целью правильного функционирования, так как она должна отвечать всем требованиям пользователя.

Нужно понимать, как именно работает заведующий и каковы его обязанности.

Объектом исследования является деятельность завуча Муниципального бюджетного образовательного учреждения «Средней общеобразовательной школы № 152» г. Красноярска.

Завуч – это заведующий учебной частью, заместитель директора школы (училища) по учебно-воспитательной работе.

Функции завуча:

- Завуч по учебно-воспитательной работе – составление расписания, методическая работа и внутришкольный контроль.
- Завуч по научно-методической работе – планирование и контроль научной работы учащихся.
- Завуч по административно-хозяйственной работе – контроль хозяйственной деятельности школы, материально-техническое обеспечение учебного процесса.

Обязанности завуча регулируются уставом школы, должностной инструкцией и законом РФ «Об образовании».

Завуч занимается методическим руководством педагогического коллектива. Принимает участие по подбору и расстановке кадров, также устанавливает объем их учебной нагрузки и составляет расписание. Занимается организацией по приему и выпуску учащихся.

В должностные обязанности завуча входит – руководство организацией учебного процесса, он должен обеспечивать полное и качественное выполнение учебных планов и программ, а также контроль над уровнем знаний учащихся.

Заведующий руководит работой преподавателей, и других педагогических работников, осуществляет учебный процесс, организует систему методической работы, направленной на совершенствование содержания, форм и методов обучения, организует работу по методическому обеспечению учебного процесса.

В результате анализа деятельности завуча были выявлены следующие сущности:

1. Ученики;
2. Класс;
3. Родители;
4. Учителя;
5. Категория;
6. Учебный план;
7. Предметы.

Среди всех сущностей главной будут являться ученики. Так как завуч руководит организацией учебного процесса, что не может происходить без наличия учеников в школе.

2.2 Этапы проектирования базы данных

2.2.1 Инфологическое проектирование

Итак, перейдем к инфологическому проектированию базы данных.

1. У родителей может быть несколько детей (учеников), а у одного ребенка может быть только один родитель (Рисунок 1).



Рисунок 1 – Инфологическая модель «Родители – Ученики»

2. Определенная группа учеников объединяет в себе один определенный класс, а один класс содержит в себе некоторое количество учеников (Рисунок 2).



Рисунок 2 – Инфологическая модель «Ученики – Класс»

3. Определенный класс включает в себя несколько учебных планов, а некоторое количество учебных планов могут быть в одном классе (Рисунок 3).



Рисунок 3 – Инфологическая модель «Класс – Учебный план»

4. Один предмет может быть включен в несколько учебных планов, а учебный план включает в себя только один предмет (Рисунок 4).

Учебный план ∞ _____ Содержит _____ 1 Предметы

Рисунок 4 – Инфологическая модель «Учебный план – Предметы»

5. У каждого учителя имеется несколько учебных планов. А учебный план включает в себя одного учителя (Рисунок 5).

Учебный план ∞ _____ Имеется _____ 1 Учителя

Рисунок 5 – Инфологическая модель «Учебный план – Учителя»

6. У каждого учителя может быть только одна определенная категория, но у категории может быть несколько учителей (Рисунок 6).

Учителя ∞ _____ Имеют _____ 1 Категория

Рисунок 6 – Инфологическая модель «Учителя – Категория»

2.2.2 Логическое проектирование

В сущности «Родители» содержатся следующие атрибуты:

- КодРодителя;

- ФИОмамы;
- ФИОпапы;
- Адрес;
- Телефон.

Следующие атрибуты содержатся в таблице «Ученики»:

- КодРодителя;
- ФИО;
- ДатаРождения;
- Класс.

«Класс» содержит такие атрибуты:

- Класс;
- КоличествоУчеников;
- КоличествоОтличников;
- КоличествоХорошистов;
- КоличествоНеАттестованных;
- Код_Предмета.

В сущность «Учебный план» включены атрибуты:

- Класс;
- КодПредмета;
- Код.

В сущности «Учителя» содержатся атрибуты:

- Код;
- ФИО;
- КлассноеРуководство;
- Стаж;
- КодКатегории.

«Предметы» содержат:

- КодПредмета;
- Предмет.

В базе сущности «Категория» содержатся:

- КодКатегории;
- Категория.

2.2.3 Нормализация базы данных

Проверим базу данных на нормализацию по первой форме.

1. Между таблицами «Учителя» и «Предметы» установлена связь «многие-ко-многим», что не соответствует правилам нормализации. И поэтому, для корректной работы системы вводится переходная таблица «Учебный план», которая примет следующий вид (Рисунок 7):



Рисунок 7 – Схема связи таблицы «Учителя» и «Предметы» при помощи переходной таблицы «Учебный план»

И таким образом, данная таблица будет содержать поля «Код» и «КодПредмета».

2. В каждой таблице все данные логически связаны между собой.
3. Наличие уникального кода у каждой записи отдельной таблицы.

Проверим базу данных на нормализацию по второй форме.

1. Созданы отдельные таблицы для наборов значений, относящихся к нескольким записям.
2. Все таблицы связаны с помощью первичного ключа.
3. И все таблицы находятся также и в первой нормальной форме.

Проверим БД на нормализацию по третьей форме.

1. Все поля зависят от первичного ключа.
2. Все таблицы соответствуют второй нормальной форме.

Таким образом, нормализация базы данных выполнена.

2.2.4 Физическое проектирование

Перейдем к физическому проектированию нашей базы данных.

«Родители» – описание структуры таблицы (таблица 1):

1. КодРодителя – тип данных – Числовой. Поле, в которое вводится код данных родителей. Поле является первичным ключом;
2. ФИОмамы – тип данных – Короткий текст. Поле, в которое вводится Фамилия, Имя и Отчество мамы;
3. ФИОпапы – тип данных – Короткий текст. Поле, в котором вводится Фамилия, Имя и Отчество папы;
4. Адрес – тип данных – Короткий текст. Поле, в котором вводится место жительства родителей;
5. Телефон – тип данных – Числовой. Поле, где вводится номер телефона родителей.

Таблица 1 – Структура таблицы «Родители»

Ключ	Имя поля	Тип данных	Размер	Ограничения
✓	КодРодителя	Числовой	Длинное целое	Primary key
	ФИОмамы	Короткий текст	250	
	ФИОпапы	Короткий текст	250	
	Адрес	Короткий текст	255	
	Телефон	Числовой	Длинное целое	

«Ученики» – описание структуры таблицы (таблица2):

1. КодРодителя – тип данных – Числовой. Поле, в которое вводится код родителей;

2. ФИО – тип данных Короткий текст. Поле, в котором вводится Фамилия, Имя и Отчество ученика;
3. ДатаРождения – тип данных – Дата и время. В этом поле вводится дата рождения ученика;
4. Класс – тип данных – Короткий текст. В данном поле записывается класс ученика.

Таблица 2 – Структура таблицы «Ученики»

Ключ	Имя поля	Тип данных	Размер	Ограничения
	КодРодителя	Числовой	Длинное целое	
	ФИО	Короткий текст	255	
	ДатаРождения	Дата и время	Краткий формат даты	
	Класс	Короткий текст	255	

«Класс» – описание структуры таблицы (таблица 3):

1. Класс – тип данных – Короткий текст. В данной поле вводится класс. Поле является ключевым;
2. КоличествоУчеников – тип данных – Числовой. В данное поле вводится количество учеников класса;
3. КоличествоОтличников – тип данных – Числовой. В данное поле вводится количество отличников класса;
4. КоличествоХорошистов – тип данных – Числовой. В этом поле вводится количество хорошистов;
5. КоличествоНеАттест – тип данных – Числовой. В данном поле вводится количество неаттестованных учеников;

6. Код_Предмета – тип данных – Числовой. В данном поле вводится код предмета.

Таблица 3 – Структура таблицы «Класс»

Ключ	Имя поля	Тип данных	Размер	Ограничения
✓	Класс	Короткий текст	255	Primary key
	КоличествоУчеников	Числовой	Длинное целое	
	КоличествоОтличников	Числовой	Длинное целое	
	КоличествоХорошистов	Числовой	Длинное целое	
	КоличествоНеАттест	Числовой	Длинное целое	
	Код_Предмета	Числовой	Длинное целое	

«Учебный план» – описание структуры таблицы (таблица 4):

1. Класс – тип данных – Короткий текст. В этом поле вводится класс;
2. КодПредмета – тип данных – Числовой. В данном поле вводится код предмета;
3. Код – тип данных – Числовой. Поле, в которое вводится код.

Таблица 4 – Структура таблицы «Учебный план»

Ключ	Имя поля	Тип данных	Размер	Ограничения
------	----------	------------	--------	-------------

	Класс	Короткий текст	255	
	КодПредмета	Числовой	Длинное целое	
	Код	Числовой	Длинное целое	

«Учителя» – описание структуры таблицы (таблица 5):

1. Код – тип данных – Числовой. Поле, где вводится код. Данное поле является ключевым;
2. ФИО – тип данных – Короткий текст. Поле, в которое вводится Фамилия, Имя и Отчество учителя;
3. КлассноеРуководство – тип данных – Короткий текст. Поле, где вводится классное руководство;
4. Стаж – тип данных – Числовой. Поле, в которое вводится стаж учителя;
5. КодКатегории – тип данных – Числовой. Поле, где вводится код категории.

Таблица 5 – Структура таблицы «Учителя»

Ключ	Имя поля	Тип данных	Размер	Ограничения
✓	Код	Числовой	Длинное целое	Primary key
	ФИО	Короткий текст	150	
	КлассноеРуководство	Короткий текст	100	
	Стаж	Числовой	Длинное	

			целое	
	КодКатегории	Числовой	Длинное целое	

«Категория» – описание структуры таблицы (таблица 6):

1. КодКатегории – тип данных – Числовой. В данном поле записывается код категории. Поле является ключевым;
2. Категория – тип данных – Короткий текст. Поле, куда записывается категория учителя.

Таблица 6 – Структура таблицы «Категория»

Ключ	Имя поля	Тип данных	Размер	Ограничения
✓	КодКатегории	Числовой	Длинное целое	Primary key
	Категория	Короткий текст	100	

«Предметы» – описание структуры таблицы (таблица 7):

1. КодПредмета – тип данных – Числовой. В данном поле указывается код предмета. Оно является ключевым;
2. Предметы – тип данных – Короткий текст. В данном поле записывается предмет.

Таблица 7 – Структура таблицы «Предметы»

Ключ	Имя поля	Тип данных	Размер	Ограничения
✓	КодПредмета	Числовой	Длинное целое	Primary key

	Предмет	Короткий текст	100	
--	---------	-------------------	-----	--

Данный этап был завершающим этапом проектирования базы данных. Поэтому сейчас переходим к разработке клиентской части информационной системы.

3.2 Создание базы данных в Microsoft Office Access

Для создания новой базы данных нужно:

- Загрузить Access, в появившемся окне из всех предложенных вариантов выбираем «Пустая база данных рабочего стола» (Рисунок 9);
- В появившемся окне задаем имя нашей базы данных и указываем место расположения БД. Далее щелкаем кнопку «Создать» (Рисунок 10).

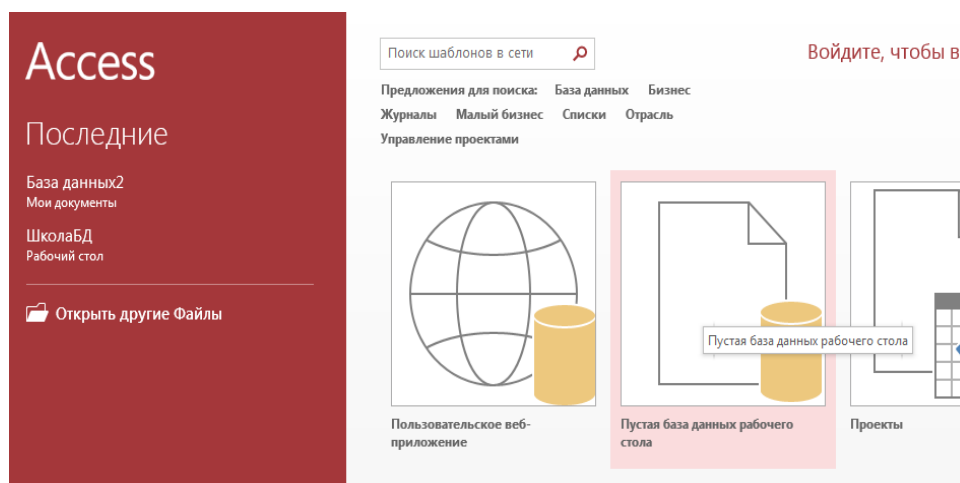


Рисунок 9 – Создание базы данных

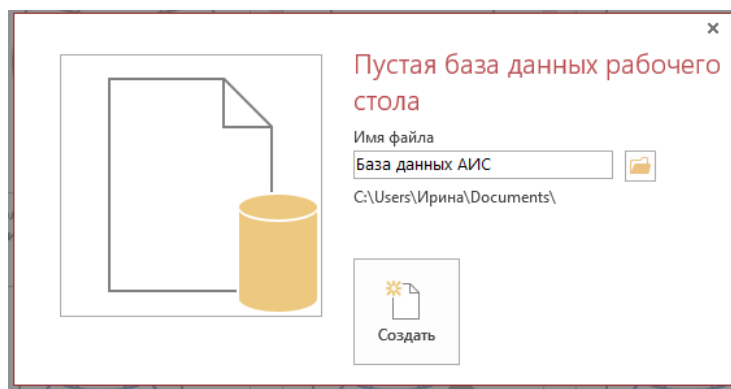


Рисунок 10 – Сохранение БД

Далее все продолжается с создания таблиц, в которых и будет храниться информация об изучаемой области или предмете. База данных обычно включает несколько взаимосвязанных таблиц.

Создать таблицу можно в разных режимах: режиме таблицы, конструктора, мастера таблиц, импорта таблиц и связи с таблицами.

Рассмотрим возможность создания таблиц в режиме конструктора, как наиболее часто используемого.

После создания нашей базы данных в появившемся окне на вкладке «Главная» нажимаем «Режим» и выбираем соответственно «Конструктор» (Рисунок 11).

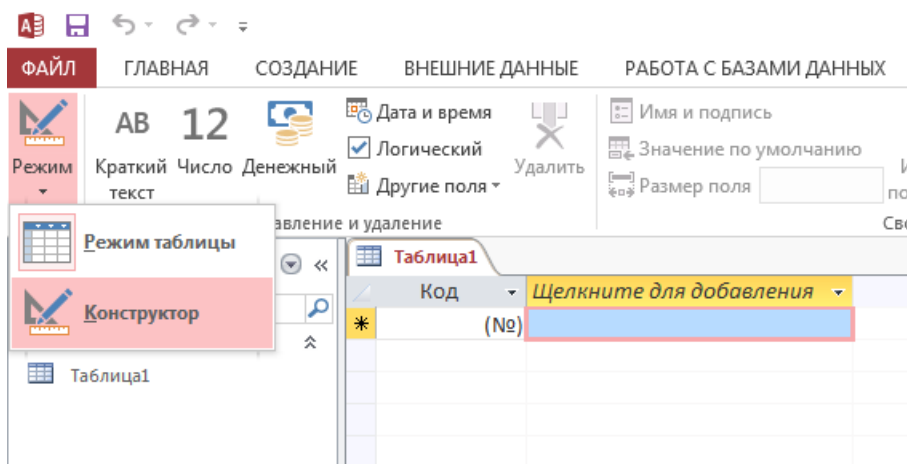


Рисунок 11 – Режим «Конструктор»

Но прежде чем произойдет переход таблицы в режим конструктора, появится окно, где необходимо указать имя таблицы и сохранить ее (Рисунок 12).

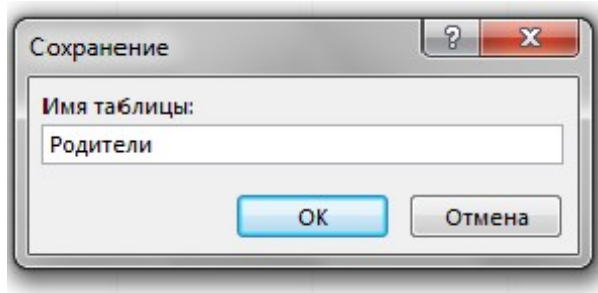


Рисунок 12 – Сохранение таблицы

В результате проделанных операций открывается окно для описания структуры таблицы и других ее характеристик (Рисунок 13).

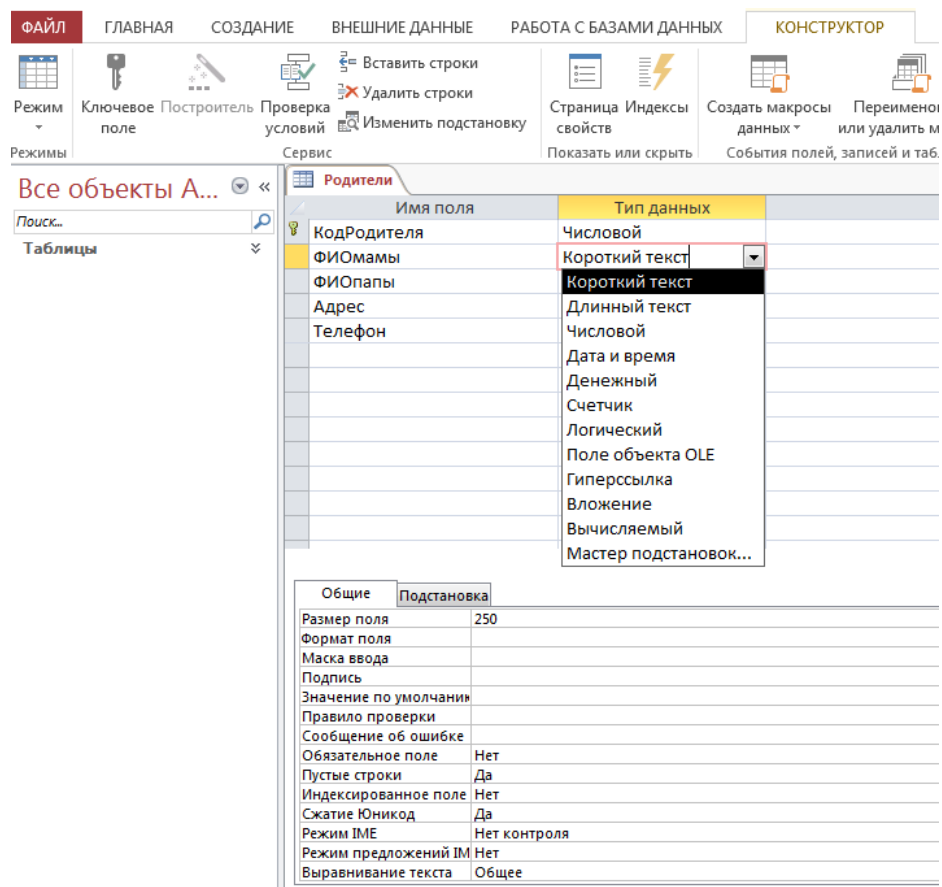


Рисунок 13 – Описание структуры таблицы

Описание структуры таблицы и других ее характеристик

В табличной форме надо последовательно описать все поля создаваемой таблицы. Сначала задается имя поля. Access допускает задание длинных имен с пробелами на русском языке.

В Microsoft Access действуют следующие ограничения на имена полей:

- имя должно содержать не более 64 символов;
- имя может включать любую комбинацию букв, цифр, пробелов и специальных символов за исключением точки (.), восклицательного знака (!), надстрочного символа (¢) и прямых скобок ([]);
- имя не должно начинаться с символа пробела;
- имя не должно включать управляющие символы (с кодами ASCII от 0 до 31).

При задании имен не допускайте их совпадения с зарезервированными словами. Например, не следует давать полю имя Count, Name и т. п.

После задания имени надо выбрать тип поля. Если щелкнуть мышкой по свободной ячейке графы «Тип поля», то высветится список допустимых типов полей, из которого и следует выбрать подходящий для описываемого поля тип. Имя и тип поля должны задаваться обязательно. Графа «Описание» может не заполняться. Эта графа используется в целях документирования проекта.

Определение ключа таблицы

Каждая реляционная таблица по определению имеет ключ. Access позволяет задавать ключ при описании таблицы, но также разрешает и отказаться от этой возможности. По ключу система автоматически выполняет индексирование, а также проверяет уникальность значений ключа при вводе новых записей или их корректировке.

Если Вы собираетесь в качестве ключа выбрать автоматически задаваемый системой код (т. е. поле типа «счетчик»), то можно это поле первоначально не описывать, а подтвердить необходимость его создания при завершении описания таблицы. Access создаст это поле автоматически (Рисунок 14).

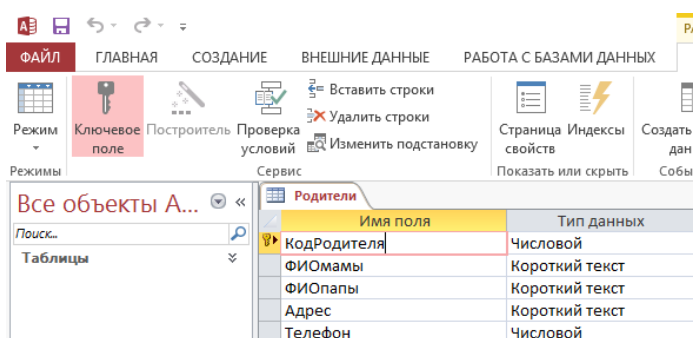


Рисунок 14 – Описание структуры таблицы с автоматически введенным системой ключевым полем Код

Аналогичные действия повторяются при создании остальных таблиц БД.

После того, как все необходимые таблицы созданы, можно задать их связанность. Чтобы создать связи между таблицами нужно:

- На вкладке «Работа с базами данных» нажать на соответствующую кнопку – «Схема данных» (Рисунок 15);

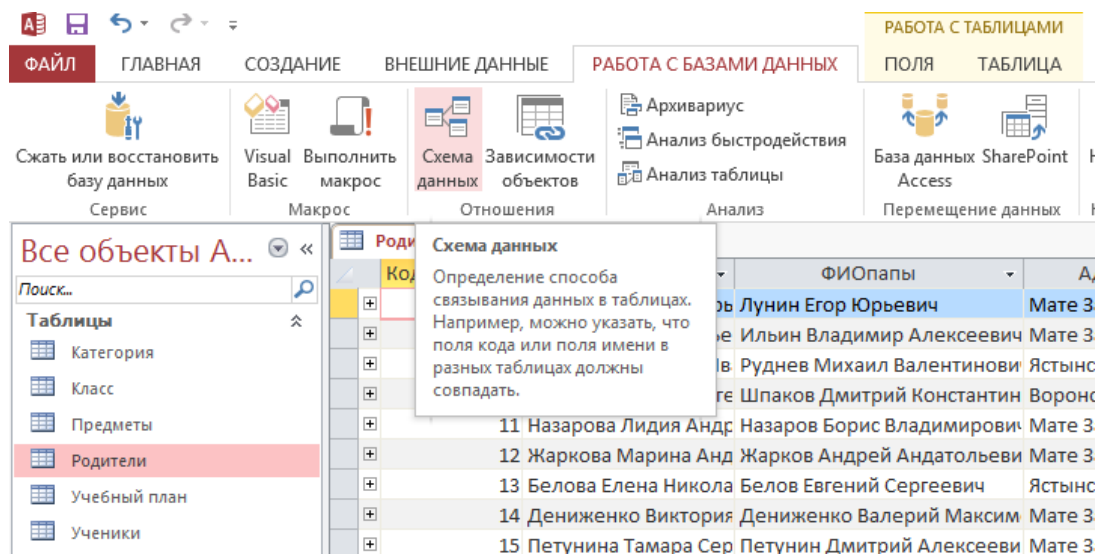


Рисунок 15 – Создание схемы данных

- Далее появляется окно для добавления таблиц и установки связей между ними (Рисунки 16 и 17).

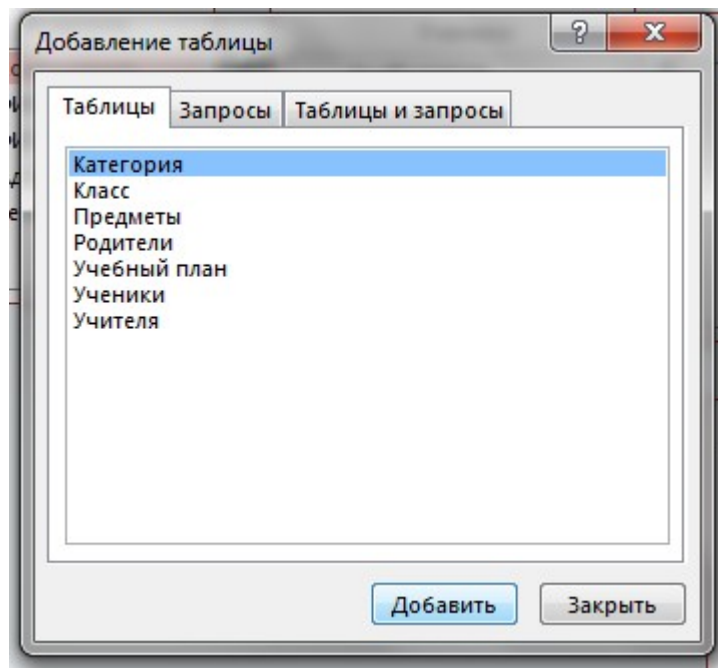


Рисунок 16 – Добавление таблиц

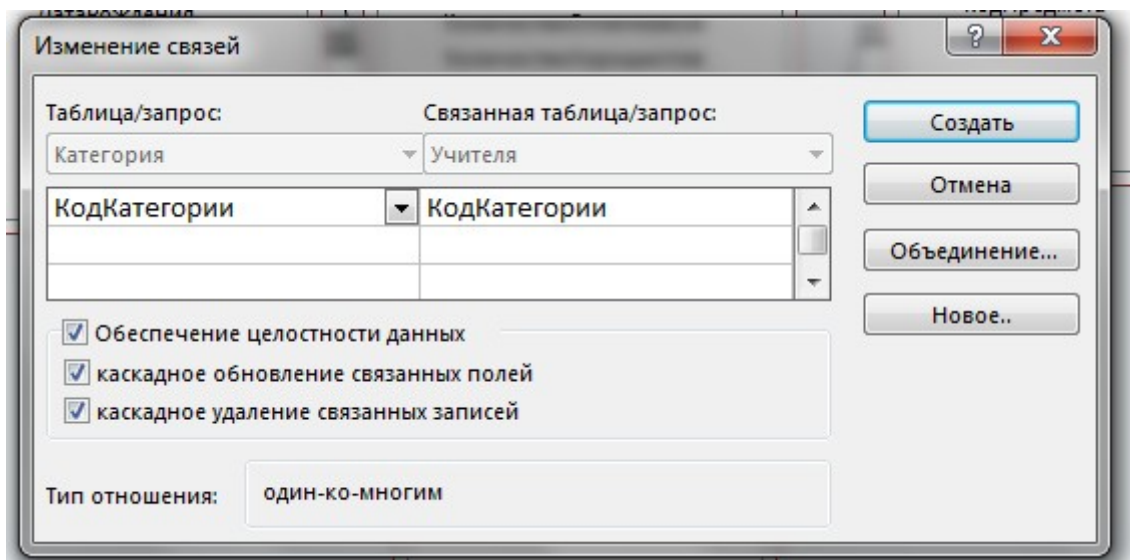


Рисунок 17 – Создание связей

В итоге получаем приведенную ниже схему данных (Рисунок 18).

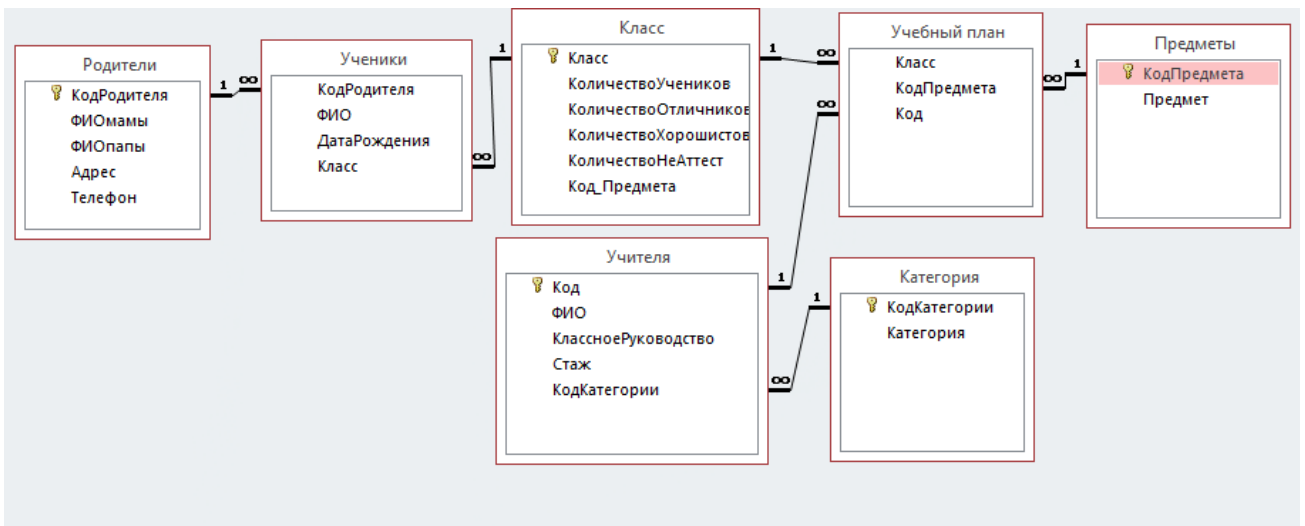


Рисунок 18 – Схема данных

ДАЛЕЕ Описание SQL-запросов (3-4 шт), среди них обязательны:

- ✓ 1 простой запрос;
- ✓ 1 запрос с использованием одной из агрегатных функций;
- ✓ 1 запрос с параметром;
- ✓ 1 запрос на добавление, или обновление, или удаление записи.