

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**  
(НИУ «БелГУ»)

**ФАКУЛЬТЕТ МАТЕМАТИКИ И ЕСТЕСТВЕННОНАУЧНОГО ОБРАЗОВАНИЯ  
ПЕДАГОГИЧЕСКОГО ИНСТИТУТА**

**КАФЕДРА ИНФОРМАТИКИ, ЕСТЕСТВЕННОНАУЧНЫХ ДИСЦИПЛИН И  
МЕТОДИК ПРЕПОДАВАНИЯ**

**РАЗРАБОТКА БАЗЫ ДАННЫХ «РАСПИСАНИЕ УРОКОВ В  
ШКОЛЕ»**

Выпускная квалификационная работа  
обучающегося по направлению подготовки 44.03.01 Педагогическое  
образование профиль Информатика  
очной формы обучения, группы 02041304  
Саляевой Натальи Сергеевны

Научный руководитель  
к.т.н. доцент Сатлер О. Н.

БЕЛГОРОД 2017

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ РАЗРАБОТКИ.....	5
1.1 Деятельность заведующего учебной частью в школе.....	5
1.2 Свойства разработки баз данных.....	9
1.3 Разработка приложений средствами Microsoft Access.....	12
2 РАЗРАБОТКА ПРОЕКТА .....	14
2.1 Разработка БД.....	14
2.2 Концептуальная модель.....	15
2.3 Логическая модель.....	20
2.4 Создание кнопочной формы и макросов .....	22
2.5 Процесс создания запросов, форм и отчетов .....	24
3 РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ.....	27
3.1 Руководство пользователя .....	27
3.2 Описание запросов.....	28
3.3 Описание форм.....	31
3.4 Описание отчетов .....	33
3.5 Рекомендации по администрированию и защите базы данных .....	36
ЗАКЛЮЧЕНИЕ .....	40
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	42

## **ВВЕДЕНИЕ**

Расписание уроков в школе – это главный организационный документ в школе, определяющий работу учеников и учителей, администрации и всего учреждения в целом, а так же он является важнейшим видом планирования процесса образования.

Составление школьного расписание является одним из сложнейших и трудоемких задач заведующего учебной частью. Он знает, сколько времени, сил и терпения уходит на подготовку и обновление расписания, так же сколько различных факторов необходимо при этом учитывать.

Чтобы облегчить задачу для составления расписания стали использовать базу данных, так как на сегодняшний момент они получили огромное распространение в различных сферах деятельности. БД используют в разных учреждениях и предприятиях, фирмах и заводах, офисах и так далее. Это обусловлено тем, что данная отрасль информационных технологий не заменима не только в практических ситуациях, но она так же может применяться даже в простых, конкретных случаях. Тем более что появляются различные реализации баз данных и системы управления базами данных. Что дает возможность выбора наиболее подходящей СУБД, как для узкопрофильного, так и для широкопрофильного специализированного предприятия.

СУБД приобрели свою популярность посредством возможности обработки огромных объемов информации и стали лучшим инструментальным средством для проектирования.

Данная работа была направлена на анализ, сравнение и разработку базы данных. В качестве средства построения БД был использован Microsoft Access. Эта реляционная система управления базами данных, которая имеет простой и графический интерфейс, позволяющий создавать не только базы данных но и разрабатывать кнопочные формы. За счет большого количества мастеров создание БД становится доступным,

интересным, занимательным, не требующим знаний языков программирования.

Access не единственная программа для создания баз данных, их точное количество не известно, так как существует несколько типов БД. Они делятся по принципу хранения на иерархические, сетевые и реляционные. Каждый тип имеет свою сферу применения к которой он наиболее подходит.

Все БД хранят в себе большие объемы систематизированных, упорядоченных данных. Благодаря этому можно легко найти нужную информацию, достаточно лишь знать что конкретно необходимо найти.

Все данные в базе приведены к определенному виду, для того чтобы информация была разделена по смыслу и значению и при нахождении необходимых знаний человек мог с легкостью обработать и получить нужный результат.

В данной выпускной квалификационной работе были представлены основные модели, типы баз данных. А так же была разработана БД «Расписание уроков в школе».

**Объект исследования** деятельность заведующего учебной частью в школе.

**Предмет исследования** - база данных «Расписание уроков в школе».

**Целью выпускной квалификационной работы** – является разработка базы данных «Расписание уроков в школе».

**Задачи:**

1. Ознакомление с деятельностью заведующего учебной частью в школе.
2. Разработка базы данных «Расписание уроков в школе».
3. Создание запросов, форм и отчетов.
4. Создание макросов и кнопочной формы.
5. Представление наиболее эффективных способов защиты и администрирования БД.

## **1 Теоретическая часть разработки**

### **1.1 Деятельность заведующего учебной частью в школе**

Школа – это учебное заведение, которое не может существовать без управления. Четкое деление функциональных обязанностей помогает лучше понимать порученное дело. Человек, отвечающий за деятельность всей школы это заведующий учебной частью, а так же заместитель директора[3].

Он отвечает за организацию учебного процесса в школе, за соблюдение выполнения учебных программ, качество преподавания, контролирует постановку учебно-воспитательной работы, следит за успеваемостью и посещаемостью учеников, регулирует нагрузку учителя, организует методическую работу в учебном заведении, составляет расписание уроков и отчетов о состоянии учебно-воспитательной работы. Контролирует состояние и результаты работы в школе. Но так же заведующий учебной частью получает и такие полномочия, в пределах которых может, принимает решения и дает соответствующие указания.

Педагогическому коллективу необходима строгая согласованность действий всех учителей. Ведь учительский коллектив обязан выступать перед детьми как единое целое, не допуская некорректных действий[7].

Каждый заведующий учебной частью должен иметь соответствующие качества, такие как умение организовывать работу коллектива и поддерживать целостный педагогический процесс. Во главе учителей необходим человек, который может объединить вокруг себя коллектив единомышленников, способных на сотрудничество и взаимопонимание. Заведующий учебной частью является “двигателем”, который поддерживает рабочий ритм в коллективе, создает положительный микроклимат для продуктивного взаимодействия педагогического коллектива в школе[31].

Чтобы правильно организовать учебный процесс в школе, необходимо составить корректное расписание, а это сложный и трудоемкий процесс, в котором необходимо учесть множество условий, не допустив при этом грубых ошибок. Это довольно сложно сделать, так как он требует от составителя большой подготовительной работы, глубокого понимания, а так же соответствующих знаний учебных планов и программ.

При составлении расписания необходимо учитывать таблицу, показывающую ранги (баллы) на сколько сложен предмет и на каком уроке учащиеся лучше усвоят новые знания. Ранг предмета зависит от сложности усвоения ребенком того или иного урока. Чем больше внимания и концентрации нужно для усвоения предмета, тем более высокий ранг. Самыми высокоранговыми предметами считаются: алгебра, геометрия, а также русский язык они оцениваются высшим рангом, то есть 11 баллами[13].

Для каждого класса существует своя таблица рангов, потому что в каждый период взросления ребенка наилучшее время для усвоения знаний меняется. Так же в разные дни работоспособность учащихся неодинакова. Её уровень увеличивается к середине недели, а в начале и в конце считается низким, поэтому распределение нагрузки устанавливается таким образом, чтобы была большая усвояемость во вторник и четверг у 1-7 классов (среда облегченный день), в 8-11 это вторник, среда и четверг.

В эти дни недели расписание уроков содержит либо более трудные, либо средние и легкие по трудности предметы, но их должно быть больше, чем в оставшиеся дни.

Интервал с 10-12 часов, считается наиболее благоприятным временем для усвоения знаний, при наименьших психофизических затратах организма[24].

В расписании уроков необходимо чередовать различные по сложности уроки в течение дня и недели: для учащихся первой ступени образования основные предметы (математика, русский, иностранный

язык, природоведение, информатика) следует чередовать с уроками музыки, изобразительного искусства, труда, физической культуры; для учащихся второй и третьей ступени предметы с математическим профилем нужно чередовать с гуманитарием[18].

Так же существует правило, что предметы, с трудной и длительной домашней подготовкой не могут ставиться в один день[2].

Рекомендуется при составлении расписания использовать таблицу И.Г. Сивкова (1975г.), разработанную сотрудниками НИИ гигиены и охраны здоровья детей и подростков, в которой трудность каждого предмета ранжируется в баллах (смотреть таблицу 1).

Таблица 1 - Шкала трудности учебных предметов

№	Общеобразовательные предметы	Количество баллов								
		Классы								
		1-4	5	6	7	8	9	10	11	
I	II	III	IV	V	VI	VII	VIII	IX	X	
1	Математика	8	10	13	-					
2	Русский язык, иностраный язык	7	8 9	12 11	11 10	7 8	6 9	9 8	9 8	
3	Природоведение, информатика	6	5 4	8 10	- 4	- 7		7	6	6
4	Русская литература	5	4	6	4	4	7	8	8	
5	История	4	5	8	6	8	10	5	5	
6	Рисование музыка	3	3 2	3 1	1 1	3 1	-			
7	Труд	2	4	3	2	1	4			
8	Физическая культура	1	3	4	2	2	2	1	1	

Продолжение Таблицы 1

I	II	III	IV	V	VI	VII	VIII	IX	X
9	Алгебра				10	9	7	10	10
10	Геометрия				12	10	8	11	11
11	Химия				13	10	12	11	11
12	География	-		7	6	6	5	3	3
13	Биология		10	8	7	7	7	7	7
14	Физика,				8	9	13	12	12
15	Черчение	3	-	-	-	5	4	-	-
16	МХК				8	5	5	5	5
17	Краеведение		7	9	5	5	-	2	2
18	ОБЖ		1	2	3	3	3	2	2

Знание этой таблицы помогает заведующему учебной частью более рационально составлять расписание для учащихся, что обеспечивает ученикам наибольшую усвояемость предметов и повышения уровня знаний в школе. При корректном распределении нагрузки знания, даваемые учителями, будут быстрее и качественнее усваиваться, что приведет к большей успеваемости учеников. Заведующий учебной частью помогает достичь наиболее высокого уровня знаний в школе.

А теперь представьте одиннадцать классов, хотя бы по одной параллели и каждому классу необходимо расписание, подобранное для всех учащихся с наиболее подходящим временем для лучшего усвоения предмета.

## 1.2 Свойства разработки баз данных

База данных (БД) — это информация, организованная по определенным правилам, которая имеет сформировавшуюся структуру и способна систематизировать большие объемы данных[10].

Можно так же сказать, что БД это хранилище для огромных объемов данных.

Базы данных имеют много разновидностей, которые отличаются своими свойствами и критериями. Основные типы БД:

- Иерархическая;
- Сетевая;
- Объектно-ориентированная;
- Реляционная.

Для создания базы данных администратор определяет: какие таблицы будут составлять его БД и какие данные будут входить в каждую таблицу, сколько всего их будет и как связать таблицы. Все эти вопросы должны быть решены на этапе проектирования БД[4].

В процессе проектирования разработчик определяет логическую, физическую и концептуальную модель структуры БД. Определяет состав реляционных таблиц, их структуру и связи между ними.

Так же перед созданием БД необходимо хорошо изучить предметную область, знать все источники информации, а так же актуальные данные, чтобы пользователи могли находить нужную им информацию.

Если на этапе проектирования БД выполнены все эти действия: определен состав, структура данных, предметная область, то все это поможет пользователю при нахождении нужной информации, а так же при решении поставленных задач. Структуру данных предметной области может показать информационно-логическая модель, на базе которой можно легко создать реляционную[8].

Этапы проектирования при создании БД можно представить в такой последовательности:

- Определение темы и информации, которая будет содержать БД;
- Построение концептуальной модели;
- Построение логической модели данных;
- Построение физической модели БД;
- Конструирование таблиц БД;
- Создание связей между таблицами;
- Создание схемы данных;
- Ввод данных в таблицы (создание записей);
- Создание необходимых форм, запросов, макросов, модулей, отчетов;
- Создание пользовательского интерфейса[27].

В процессе создания модели данных необходимо выделить информационные объекты, которые соответствуют требованиям нормализации данных, а так же определить связи между ними. Такая модель позволяет сделать реляционную БД, не применяя дублирование. Которое позволяет однократно ввести данные при первоначальной загрузке и корректировке, а также сохранить целостность данных при внесении изменений в таблицы.

Существует два подхода для разработки модели данных. Первый подход: сначала определяются главные задачи, при решении которых строится БД и выявляются потребности задач в данных. Во втором подходе сразу устанавливаются типовые объекты предметной области. Но лучше рационально сочетать сразу два подхода. Потому что на первом этапе обычно, нет исчерпывающих сведений обо всех задачах. Применение такой технологии лучше тем, что гибкие средства создания реляционных БД дают возможность на любом из этапов разработки изменить её или улучшить структуру данных без потери ранее внесенной информации[21].

Рассмотрим формальные правила, которые используются для выделения информационных объектов.

- На базе описания предметной области нужно рассмотреть документы, а так же их атрибуты, подлежащие хранению в БД;
- Выявить функциональную зависимость между атрибутами;
- Выбрать все зависимые атрибуты и указать для них ключевые атрибуты, то есть те, от которых он зависит;
- Сгруппировать атрибуты, которые одинаково зависят от ключевых атрибутов.

Получившиеся группы зависимых атрибутов вместе с их ключевыми атрибутами образуют информационные объекты.

При создании логической структуры реляционной БД на базе модели каждый информационный объект нормально виден в реляционной таблице, а связи между таблицами соответствуют связям между информационными объектами[28].

В процессе создания сначала конструируются таблицы БД, которые соответствуют информационным объектам построенной модели данных. Потом может создаваться схема данных, которая фиксирует существующие логические связи между таблицами. А они в свое время соответствуют связям информационных объектов. В схеме данных бывают заданы параметры поддержания целостности БД, в случае если модель данных была разработана в соответствии с требованиями нормализации. Целостность данных означает, что в базе данных имеются правильные и корректные взаимосвязи между записями разных таблиц при загрузке, добавлении и удалении записей в связанных таблицах, как и при изменении значений ключевого поля[16].

После создания и редактирования схемы данных происходит ввод непротиворечивых данных из документов предметной области.

На базе созданной БД формируются нужный запрос, форма, макрос, модуль, отчет, производящие нужную обработку данных базы и их представление.

При помощи встроенных средств и инструментов в БД может быть создан пользовательский интерфейс, который позволяет управлять процессами ввода, хранения, обработки, обновления и представления информации в БД[5].

### **1.3 Разработка приложений средствами Microsoft Access**

Для создания приложения в СУБД Microsoft Access существует инструментарий, который позволяет создавать приложения. Есть два способа их создания. Первый и простейший способ это написание макросов, второй и более сложный это написание с использованием языка программирование. Хотя макросы и являются достаточно хорошим средством для создания, автоматизирующее работу с БД, но все же есть задачи, которые не разрешить с помощью них, либо те для решения которых более подходит язык программирования[25].

Работая в среде Microsoft Access, пользователь часто сталкивается с проблемой многократного повторения одних и тех же действий. Для того чтобы упростить или ускорить работу можно использовать макросы. Макросы – это группа команд, оформленная специальным образом и представляющая собой единое целое. При этом в макросе нужно определить не только команду, но и ее параметры[14].

Для программирования в MS Access используется процедурный язык Visual Basic for Applications (VBA) с добавлением объектных расширений и элементов SQL. Язык программирования VBA есть общий инструмент для всех приложений Microsoft Office, который позволяет решать разные задачи программирования, начиная от автоматизации многократных

действий разработчика и заканчивая разработкой полномасштабных приложений, использующих Microsoft Access в качестве среды разработки[17].

Код VBA в приложении Microsoft Access содержится в модулях. Модули являются такими же объектами Microsoft Access, как таблицы, запросы, формы, отчеты и макросы. Основное содержание модулей – это процедуры на языке VBA. Применение модулей требует от пользователя знания языка программирования и основных принципов объектно-ориентированного программирования[1].

Если в БД имеются макросы, то они могут быть преобразованы в процедуры на VBA. Это касается как макросов, связанных с событиями в формах и отчетах, так и общих макросов, не связанных с формами и отчетами. Все они преобразуются в отдельные модули, причем макросы, выполняемые при возникновении событий, преобразуются в процедуры обработки событий[29].

Процесс создания программных расширений в среде Microsoft Access предполагает активное использование технологии объектно-ориентированного программирования (ООП). Каждый объект базы данных предполагает набор свойств, методов и событий. Множество различных событий имеется для форм, отчетов и элементов управления, которые используются в них. При работе с приложениями могут запускаться специально разработанные программы – обработчики событий (событийные процедуры). Работа программиста, сводится к нахождению такого события, которое необходимо обработать программой, и программированию реакции на событие, а также к созданию новых объектов, описанию их свойств, методов и событий[22].

## **2 Разработка проекта**

### **2.1 Разработка БД**

При выполнении выпускной квалификационной работы была разработана база данных «Расписание уроков в школе» в программе Microsoft Access. Применение такой БД увеличивает уровень организации деятельности в образовательных учреждениях, таких как школа. Во всех учебных заведениях невозможен образовательный процесс без расписания занятий, а использование нашей базы данных позволит значительно облегчить процесс создания и редактирования расписания.

БД включает в себя таблицы:

1. Аудитории
2. Дни недели
3. Занятость аудиторий
4. Занятость класса
5. Занятость учителей
6. Классы
7. Период проведения уроков
8. Предметы
9. Расписание
10. Учителя

Первая таблица включает в себя код предмета, номер аудитории, количество мест в аудиториях и названия кабинетов. Вторая состоит из номера дня недели и названия дня недели. В третьей таблице находится номер урока, аудитории, дни недели, признак занятости. Четвертая таблица включает в себя номер класса, урока, день недели и признак занятости. Пятая показывает номер урока, день недели, код учителя и признак занятости. Шестая состоит из номера класса, количества человек в классе. Седьмая таблица включает в себя номер, время начала и время окончания урока. Восьмая состоит из наименования предмета и кода предмета.

Девятая таблица показывает номер урока, номер аудитории, номер дня недели, код предмета, код учителя, номер класса. В десятой таблице находятся код учителя, его фамилия, имя и отчество, а так же код предмета, который он ведет.

## **2.2 Концептуальная модель**

Концептуальное проектирование основывается на анализе предметной области, которое включает в себя анализ концептуальных требований и информационных потребностей, выявляет информационные объекты и связи между ними.

Концептуальная модель содержит описание объектов и их взаимосвязей в интересующей предметной области, а так же она важна при выявлении результата и анализе данных. Модель применяется для структурирования предметной области с учетом информационного интереса пользователя системы, с помощью которого появляется возможность систематизировать информационное содержание предметной области, позволяет увидеть отдельные элементы программного обеспечения. При этом, уровень детализации зависит от выбранной модели[11].

Концептуальная модель является представлением точки зрения пользователя на предметную область и не зависит ни от программного обеспечения системы управления базами данных, ни от технических решений.

Одной из распространенных моделей концептуальной схемы является модель «сущность - связь». Основными конструкциями данной модели являются сущности и связи. Под сущностью понимают некоторую принятую в конкретной постановке задачи абстракцию реального мира, процесса или явления, о котором необходимо хранить информацию в

системе. В качестве сущности могут выступать места, вещи, личности или явления. А экземпляром сущности - является конкретный объект[15].

Основными элементами концептуальной модели данных являются объекты и отношения. Объекты представляют собой любой конкретный объект в нужной области. Объекты в каждый момент времени характеризуются определенным состоянием, которое должно описываться каким-то набором свойств и отношений с другими объектами.

Характеристика, определяющая свойство сущности, которое можно сформулировать и записать, называется атрибутом. Атрибут, который однозначно определяет сущность, называется идентификатором.

Каждый объект предметной области может иметь некоторый набор атрибутов, отображающих свойства объекта. Атрибуты используются для определения того, какая информация должна быть собрана об объекте. Примерами атрибутов для объекта «Учителя» служат Фамилия, Имя, Отчество и т.д.

Связь определяет отношения между сущностями. Типы связей: один к одному, один ко многим, многие ко многим.

При построении модели «сущность - связь» используют графические диаграммы. При этом обозначают: сущности - прямоугольниками, атрибуты - овалами, связи – ромбами.

После выбора сущностей, задания атрибутов и анализа связей можно перейти к проектированию информационной (концептуальной) схемы БД.

Каждая сущность имеет свой набор атрибутов, один из которых является ключевым, однозначно характеризующим её. Например в таблице «Аудитории» главным атрибутом является «Номер аудитории» (Смотреть рисунки 1-10). В некоторых сущностях было затруднительно указать ключевой атрибут, поэтому мы вводили дополнительные атрибуты, чтобы однозначно идентифицировать их.

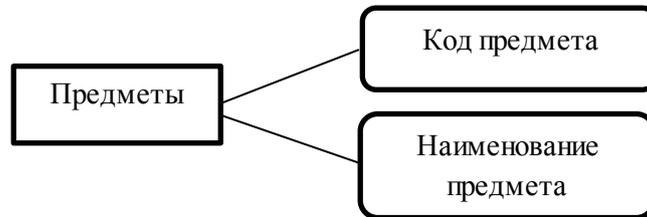


Рисунок 1 – Атрибуты сущности «Предметы»

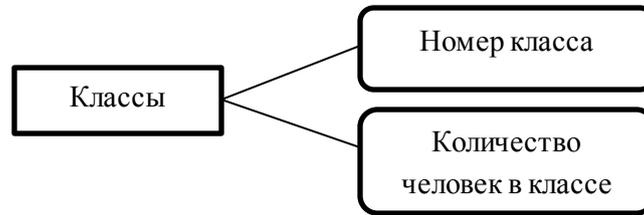


Рисунок 2 – Атрибуты сущности «Классы»

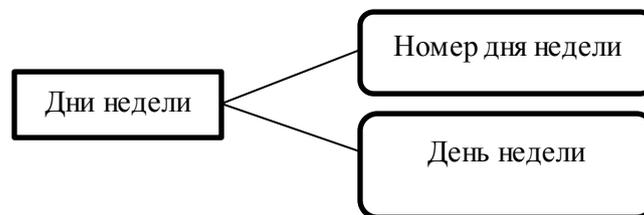


Рисунок 3 – Атрибуты сущности «Дни недели»

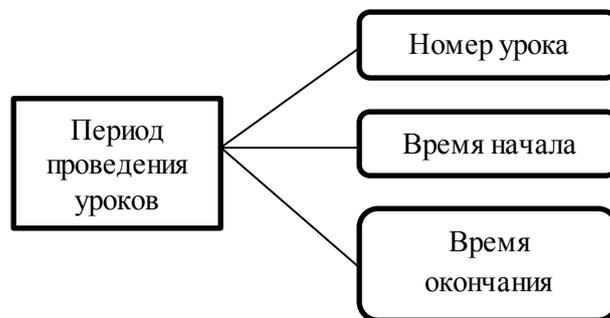


Рисунок 4 – Атрибуты сущности «Период проведения уроков»

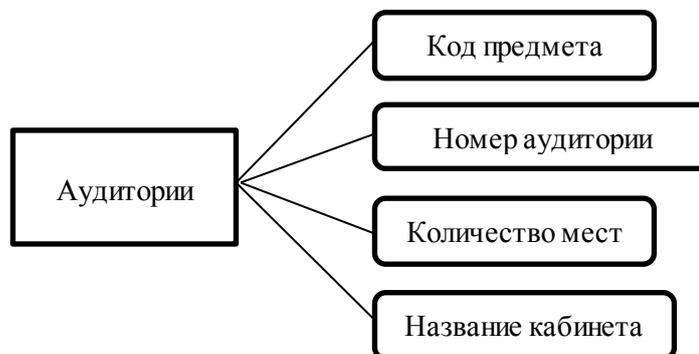


Рисунок 5 – Атрибуты сущности «Аудитории»

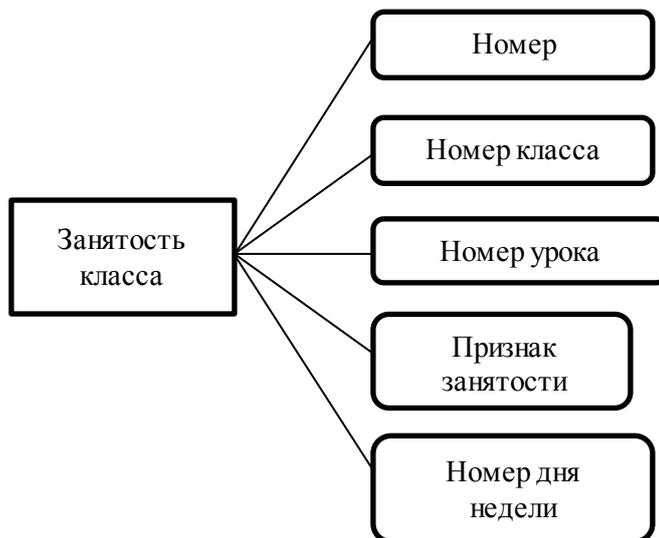


Рисунок 6 – Атрибуты сущности «Занятость класса»

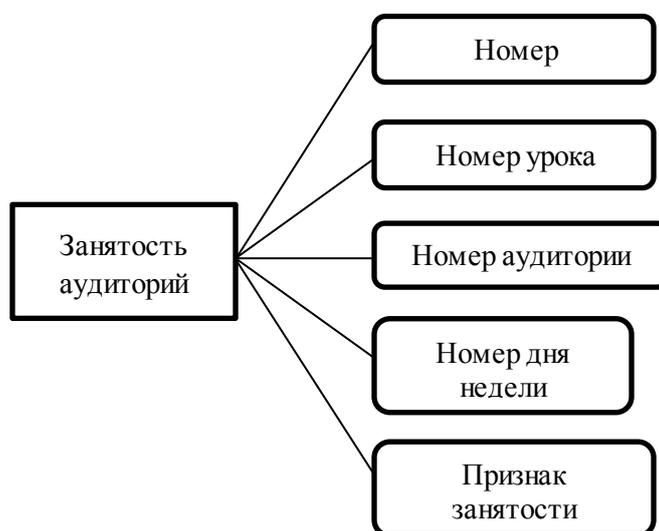


Рисунок 7 – Атрибуты сущности «Занятость аудиторий»



Рисунок 8 – Атрибуты сущности «Занятость преподавателей»

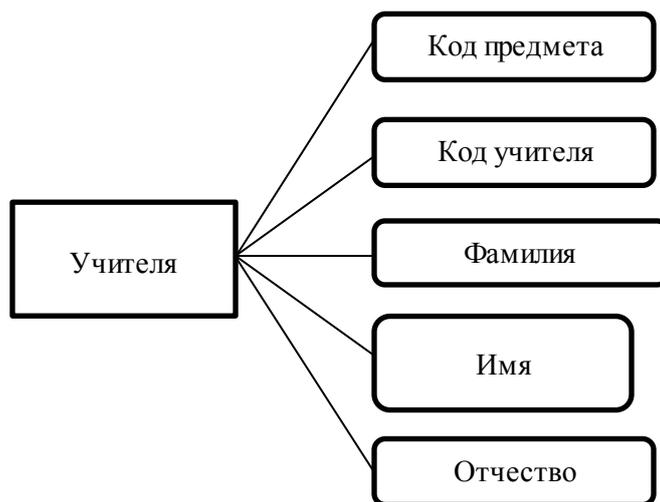


Рисунок 9 – Атрибуты сущности «Учителя»

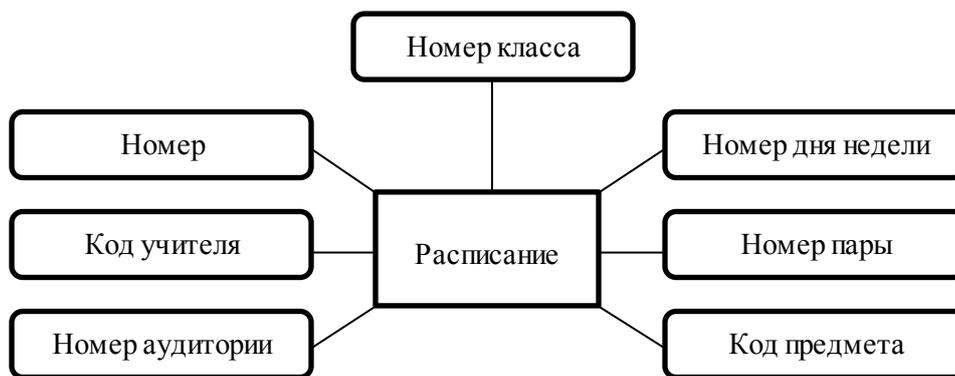


Рисунок 10 – Атрибуты сущности «Расписание»

Созданная нами концептуальная модель базы данных «Расписание уроков в школе» включает в себя структурированное описание предметной области БД, которая является простой и понятной для любого пользователя. Так же инфологическая модель отражает связи между сущностями и ключевые атрибуты.

## 2.3 Логическая модель

Следующим этапом построения БД является логическое или дата-логическое проектирование.

Это создание схемы базы данных на основе конкретной модели данных. Например: преобразование концептуальной модели в логическую. Она строится в терминах информационных единиц, но без привязки к конкретной СУБД. Более того, логическая модель данных обязательно должна быть выражена средствами именно реляционной модели данных[12].

Первым шагом является построение ER-диаграмм, которые состоят из трех частей: сущностей (существительные), атрибутов (прилагательные) и взаимосвязей (глаголы). ER-диаграмма позволяет рассмотреть нужную тему целиком и выяснить требования необходимые для ее разработки (смотрите рисунок 11).

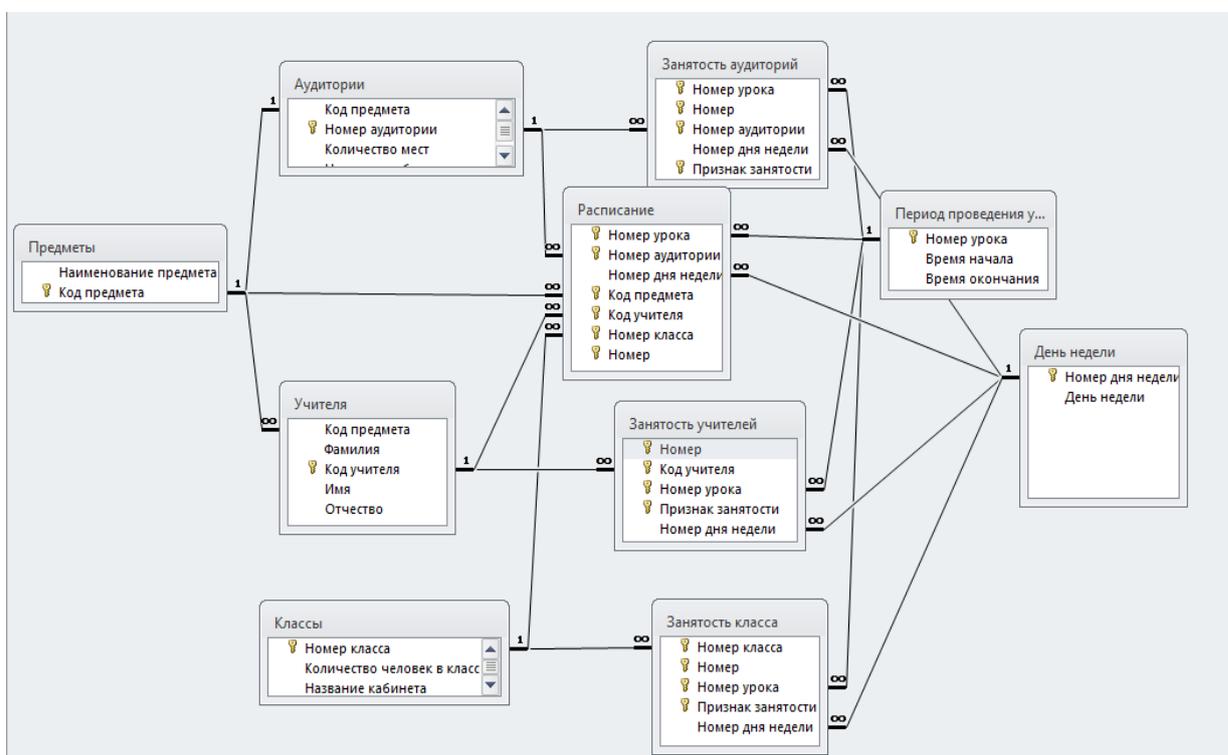


Рисунок 11 – Логическая модель БД «Расписание занятий в школе»

Одну и ту же ER-модель можно преобразовать как в реляционную модель данных, так и в модель данных для иерархических и сетевых СУБД, или в постреляционную модель данных. Однако так как мы рассматриваем именно реляционные системы управления базами данных, то можно считать, что логическая модель данных для нас формулируется в терминах реляционной модели данных[6].

На этом этапе проектирования учитывается специфика конкретной модели базы данных, но может не учитываться специфика конкретной СУБД.

Нормализация отношений информационной модели предметной области является механизмом создания логической модели реляционной базы данных.

Математическая точка зрения рассматривает задачу построения как информационной модели предметной области, так и логической модели реляционной базы данных является результатом решения следующих комбинаторных задач:

1. группировкой атрибутов в отношении предметной области;
2. распределением атрибутов по отношениям базы данных.

Эти задачи имеют большое число вариантов решений, и приводят к проблеме подбора оптимального варианта из множества альтернативных схем отношений. Выбор наиболее рационального варианта обуславливается соблюдением различных соглашений и требований, таких как:

1. Первичные ключи отношений должны быть минимальными.
2. Число отношений базы данных должно по возможности давать наименьшую избыточность данных.
3. Число отношений базы данных не должно приводить к потере производительности системы.

4. Данные не должны быть противоречивыми, т.е. при выполнении операций включения, удаления и обновления данных их потенциальная противоречивость должна быть сведена к минимуму.

5. Схема отношений базы данных должна быть устойчивой, способной адаптироваться к изменениям при ее расширении дополнительными атрибутами.

6. Разброс времени реакции на различные запросы к базе данных не должен быть большим.

7. Данные должны правильно отражать состояние предметной области базы данных в каждый конкретный момент времени.

Для создания системы, которая могла бы комплексно удовлетворить все вышеперечисленные требования, необходимо проделать оптимизацию всех параметров, многих из которых не имеют однозначного решения. Большинство требований противоречивы. Так, например, требование производительности противоречит требованию гибкости. Требование минимизировать число отношений в БД противоречит требованию надежности данных.

## **2.4 Создание кнопочной формы и макросов**

Главная кнопочная форма создается с целью навигации по БД. Такая форма может быть представлена в качестве главного меню БД. Главная кнопочная форма состоит из элементов, таких как таблицы, запросы, формы и отчёты.

Формы - это объекты БД, обеспечивающие просмотр данных из таблиц и запросов, а так же выполняющие функцию ввода данных в базу, корректирования имеющейся информации и выполнение определенных действий. В форме могут содержаться графики, рисунки и другие внедренные объекты[9].

При внесении данных можно не использовать формы. Но существует некоторые причины, которые делают формы незаменимым средством ввода или изменения информации в БД:

- при работе с формами ограничен доступ к таблицам(то есть сохраняется изначальная информация);
- различные пользователи должны иметь определенные права доступа к информации, хранящейся в БД. При внесении изменений в предоставленные им формы, данных могут поступать только в одну таблицу;
- осуществление ввода данных в форму проще, чем в таблицу, так как в окне формы отображается, как правило, одна запись таблицы;
- для большинства случаев информация для БД берется из бумажных бланков. Экранные формы можно сделать точной копией бумажных бланков, что ведет к уменьшению количества ошибок при внесении данных и снижению утомляемости персонала.

Для создания формы можно использовать основы нескольких таблиц или запросов с помощью Мастера, используя средство автоформы, «вручную» в режиме конструктора, сохраняя таблицу или запрос как форму. Чтобы изменить интерфейс формы необходимо поменять режим, выбрав конструктор.

Запросы и таблицы не являются элементами главной кнопочной формы. Поэтому при создании кнопок Запросы или Таблицы на кнопочной форме необходимо использовать макросы. Для этого в окне БД создаются макросы «Открыть Запрос» или «Открыть Таблицу» с уникальными именами. Далее на кнопочной форме создают кнопки для вызова этих макросов[19].

Макрос представляет собой набор макрокоманд, таких как управление, элементами БД (таблицы, формы, запросы, отчеты), сохраненных под одним общим именем. Выполнение макроса – это последовательное выполнение команд, составляющих его. Макросы

обычно предназначены для автоматизации простых, но часто выполняемых действий.

Для одной БД можно создать несколько кнопочных форм. Кнопки необходимо группировать на страницах кнопочной формы таким образом, чтобы пользователю было легче понять, в каких кнопочных формах можно выполнять определенные команды, например редактирование данных. Следует отметить, что на подчиненных кнопочных формах необходимы кнопки возврата в главную кнопочную форму.

Технология создания кнопочных форм следующая:

1. создать страницу главной кнопочной формы;
2. создать необходимое количество страниц подчиненных кнопочных форм;
3. реализовать элементы главной кнопочной формы;
4. сделать элементы для кнопочных форм отчетов и форм ввода или изменения данных;
5. создать макросы для запросов или для таблиц с уникальными именами;
6. создать элементы для кнопочных форм запросов или таблиц.

## **2.5 Процесс создания запросов, форм и отчетов**

При создании базы данных мы сделали такие запросы как: «Незадействованные аудитории по дням недели», «Расписание «окон» у учителей на неделю », «Найти на каком уроке и в какой день недели аудитории под номерами 105 и 108 не задействованы», «Составить расписание по определенному учителю и дню недели», «Узнать расписание Агеевой у «2А» и «2Б» классов », «Количество уроков в каждой аудитории по каждому классу».

Запрос с условием. С помощью конструктора запросов выбираем необходимые таблицы. Затем поля которые будут использоваться и условия, по которым будет происходить отбор. Нажимаем на кнопку выполнить, проверяем результат и сохраняем запрос.

Перекрестный запрос. С помощью конструктора запросов выбираем необходимые таблицы. Далее на панели инструментов необходимо нажать на тип запроса «Перекрестный» и ввести поля и имя таблиц, которые будем использовать. Затем в каждом столбце «групповые операции» выбираем группировку. Далее в столбце «перекрестная таблица» выбираем заголовки строк, столбцов и значения.

При создании базы данных мы сделали формы: «Расписание уроков по каждому класса», «Расписания учителя», «Сведения о каждом учителе». Рассмотрим процессы создания форм.

Подчиненная форма. Чтобы создать такую форму нам необходимо сначала сделать запрос по нужным полям с помощью конструктора.

Выбираем необходимые поля и условия отбора. Нажимаем на кнопку выполнить, проверяем результат и сохраняем запрос, присвоив ему имя. После создания запроса можно приступать к созданию подчиненной формы.

В основном меню окна базы данных выберем команду Создание и далее режим мастера форм. В окне Создание форм в списке Таблицы и запросы выбрать нужную таблицу или запрос, в списке доступных полей выбрать все поля таблицы и нажать кнопку Далее. Выбрать внешний вид формы «В один столбец» и нажать кнопку Далее. Присвоить имя форме и нажать кнопку Готово. Просмотреть информацию таблицы с помощью созданной формы.

Потом переходим в режим конструктора и настраиваем форму так, чтобы она была читаемой. Затем нам необходимо создать кнопки для перехода по записям в главной форме. Для этого в главной панели выбираем элементы управления и нажимаем на элемент под названием

кнопка. Далее определив область, в которой хотели бы разместить кнопку и нажимаем на нее, растягивая чтобы придать кнопку желаемый масштаб. В появившемся диалоговом окне выбираем категорию «переход по записям», а в действиях «Предыдущая запись» и нажимаем далее. Потом выбираем оформление кнопки, текст или рисунок и нажимаем готово, эту операцию нужно повторить для кнопки перехода к следующей записи. Затем можно перейти к режиму формы и по пробовать полученную форму.

Так же можно создать форму на основе запросов, выполнив те же действия и простую форму, не создавая подчиненную.

При создании базы данных мы сделали отчеты: «На каких уроках каждый учитель имеет «окно» или у него свободный урок», «Незанятые аудитории по дням недели», простой отчет «Предметы» и «Расписание «i» учителя по «j» дню».

Простой отчет. Чтобы создать такой отчет нам необходимо выбрать таблицу, затем на главной панели нажать кнопку отчет. Далее придаем нужную форму отчету и просматриваем информацию таблицы с помощью созданного отчета.

Отчет с параметром. Для начала на панели управления выбрать мастер форм, затем выбираем созданный ранее запрос с параметром, не забывая, выбрать все поля и нажимаем далее, выбираем вид представления данных, снова нажимаем далее, при необходимости выбираем уровни группировки и сортировку записей и нажимаем готово. Далее придаем нужную форму отчету, предварительно перейдя в конструктор. Потом просматриваем информацию таблицы, с помощью созданного отчета выбирая необходимый параметр.

Так же можно создавать формы по нескольким таблицам, имеющим желательную связь один, ко многим.

## 3 Реализация базы данных

### 3.1 Руководство пользователя

База данных «Расписание уроков в школе» содержит в себе информацию об учителях, классах, кабинетах, предметах, занятости аудиторий, а так же само расписание. Это позволяет находить необходимую информацию достаточно быстро, сокращая и убирая большой объем ненужной, дополнительной информации.

При запуске БД открывается главное окно интерфейса, содержащее в себе навигационное меню, имеющее 4 основных раздела: таблицы, запросы, формы, отчеты и кнопку выход (Смотреть рисунок 12).

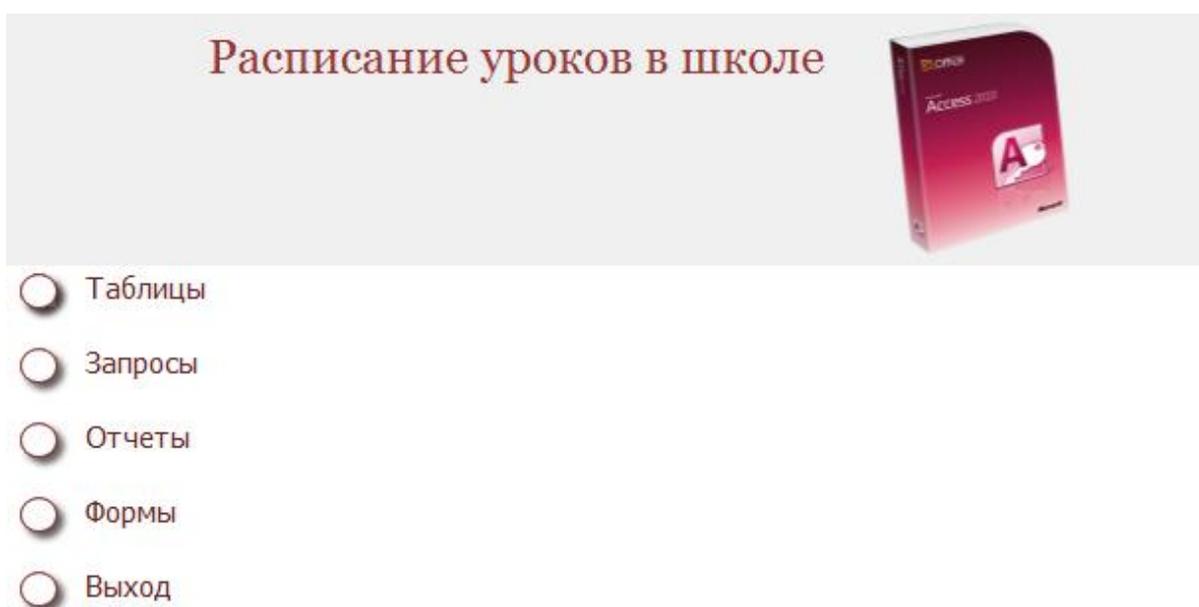


Рисунок 12 – Главное интерфейсное окно базы данных

Для удобства восприятия информации в БД имеются формы и отчеты, благодаря которым можно просмотреть интересующие Вас данные. Так же можно сформировать запрос, при этом выбрать например класс и день недели, чтобы посмотреть расписание и многое другое.

При редактировании базы данных можно изменять или вносить новую информацию, такую как:

- новый учитель, класс, кабинет и т. д.;
- удалять запись о классе;
- редактировать занятость аудиторий, учителей или классов;
- изменять наименование уроков, кабинетов и др.

Благодаря поиску по заданным критериям можно просмотреть интересующую информацию не изменяя данных, для этого необходимо выбрать нужные критерии и нажать на кнопку «Применить фильтр», которая находится на панели инструментов MS Access.

### 3.2 Описание запросов

Запрос «Незадействованные аудитории по дням недели». Результат запроса (смотрите рисунки 13 и 14).

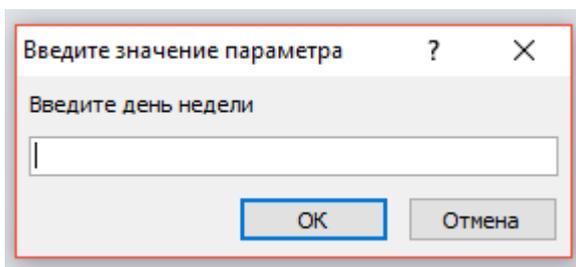


Рисунок 13 – Параметр для запроса 1

Номер дня не	Номер урока	Номер аудитории	Признак занятос
1	2	103	<input type="checkbox"/>
1	5	103	<input type="checkbox"/>
1	2	104	<input type="checkbox"/>
1	5	104	<input type="checkbox"/>
1	3	105	<input type="checkbox"/>
1	5	101	<input type="checkbox"/>
1	3	109	<input type="checkbox"/>
1	5	109	<input type="checkbox"/>

Рисунок 14 – Запрос 1

Этот запрос показывает, какие аудитории свободны в понедельник.

Запрос «Расписание «окон» у учителей на неделю ». Результат запроса (смотрите рисунок 15).

Расписание «окон» у учителей на неделю			
День недел	Номер урока	Фамилия	Признак занятости
Понедельник	5	Агеева	<input type="checkbox"/>
Вторник	5	Агеева	<input type="checkbox"/>
Среда	1	Агеева	<input type="checkbox"/>
Среда	5	Агеева	<input type="checkbox"/>
Четверг	3	Агеева	<input type="checkbox"/>
Четверг	4	Агеева	<input type="checkbox"/>
Четверг	5	Агеева	<input type="checkbox"/>
Пятница	5	Агеева	<input type="checkbox"/>
Понедельник	5	Гумбина	<input type="checkbox"/>
Среда	5	Гумбина	<input type="checkbox"/>
Пятница	5	Гумбина	<input type="checkbox"/>
Понедельник	2	Федоров	<input type="checkbox"/>
Понедельник	5	Федоров	<input type="checkbox"/>
Вторник	1	Федоров	<input type="checkbox"/>
Вторник	2	Федоров	<input type="checkbox"/>
Среда	1	Федоров	<input type="checkbox"/>
Среда	3	Федоров	<input type="checkbox"/>
Среда	5	Федоров	<input type="checkbox"/>
Четверг	5	Федоров	<input type="checkbox"/>
Пятница	5	Федоров	<input type="checkbox"/>
Вторник	5	Иванова	<input type="checkbox"/>
Пятница	4	Иванова	<input type="checkbox"/>
Пятница	5	Иванова	<input type="checkbox"/>
Среда	5	Шаталова	<input type="checkbox"/>
Пятница	4	Шаталова	<input type="checkbox"/>
Пятница	5	Шаталова	<input type="checkbox"/>
Понедельник	3	Александрова	<input type="checkbox"/>
Вторник	4	Александрова	<input type="checkbox"/>
Вторник	5	Александрова	<input type="checkbox"/>
Четверг	5	Александрова	<input type="checkbox"/>
Пятница	1	Александрова	<input type="checkbox"/>
Пятница	2	Александрова	<input type="checkbox"/>
Пятница	3	Александрова	<input type="checkbox"/>
Пятница	5	Александрова	<input type="checkbox"/>
Среда	4	Никитина	<input type="checkbox"/>
Среда	5	Никитина	<input type="checkbox"/>
Пятница	5	Никитина	<input type="checkbox"/>
Вторник	5	Кузнецова	<input type="checkbox"/>
Среда	5	Кузнецова	<input type="checkbox"/>
Четверг	5	Кузнецова	<input type="checkbox"/>

Рисунок 15 – Запрос 2

Этот запрос показывает, на каких уроках, в какие дни учителя имеют «окна».

Запрос «Найти на каком уроке и в какой день недели аудитории под номерами 105 и 108 не задействованы». Результат запроса (смотрите рисунок 16).

Незанятость аудиторий 105 и 108						
День недел	Номер урока	Номер аудитории	Признак занятос	Код предмета	Количество мест	
Понедельник	3	105	<input type="checkbox"/>	9	20	
Вторник	4	105	<input type="checkbox"/>	9	20	
Вторник	5	105	<input type="checkbox"/>	9	20	
Среда	4	108	<input type="checkbox"/>	6	10	
Среда	5	108	<input type="checkbox"/>	6	10	
Четверг	5	105	<input type="checkbox"/>	9	20	
Пятница	1	105	<input type="checkbox"/>	9	20	
Пятница	2	105	<input type="checkbox"/>	9	20	
Пятница	3	105	<input type="checkbox"/>	9	20	
Пятница	5	105	<input type="checkbox"/>	9	20	
Пятница	5	108	<input type="checkbox"/>	6	10	

Рисунок 16 – Запрос 3

Этот запрос показывает, свободные аудитории под номерами 105 и 108 на неделю.

Запрос «Составить расписание по определенному учителю и дню недели». Результат запроса (смотрите рисунок 17).

Введите значение параметра ? X

Введите код учителя

Введите значение параметра ? X

Введите номер дня недели

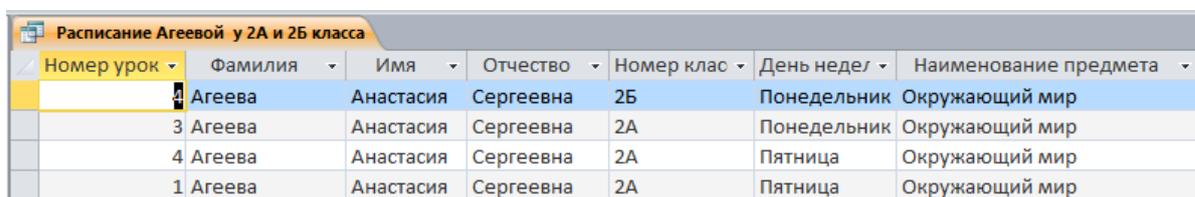
По дню недели и учителю				
Номер дня	Код учителя	Фамилия	Номер клас	День недел
2	4	Иванова	4А	Вторник
2	4	Иванова	4А	Вторник
2	4	Иванова	4Б	Вторник
2	4	Иванова	4В	Вторник

Рисунок 17 – Запрос 4

Этот запрос показывает расписание учителя Иванова на вторник.

Запрос «Узнать расписание Агеевой у «2А» и «2Б» классов».

Результат запроса (смотрите рисунок 18).



Номер урок	Фамилия	Имя	Отчество	Номер клас	День недел	Наименование предмета
1	Агеева	Анастасия	Сергеевна	2Б	Понедельник	Окружающий мир
3	Агеева	Анастасия	Сергеевна	2А	Понедельник	Окружающий мир
4	Агеева	Анастасия	Сергеевна	2А	Пятница	Окружающий мир
1	Агеева	Анастасия	Сергеевна	2А	Пятница	Окружающий мир

Рисунок 18 – Запрос 5

Этот запрос показывает расписание у учителя Агеевой у «2А» и «2Б» классов.

Запрос «Количество уроков в каждой аудитории по каждому классу». Результат запроса (смотрите рисунок 19).



Номер аудит	1А	1Б	2А	2Б	3А	3Б	4А	4Б	4В
101	1	1	3	1	1	3	2	3	3
102	4	2		2	2	2	3	3	3
103	2	4	2	2	2	2	2	4	3
104	2	3	4	3	4	2	3	2	3
105	2	1	3	1	5	4	3	2	3
106	3	2	1	2	3	3	4	2	2
107	2	2	3	3	3	2	2	3	2
108	2	3	2	3	2	4	2	2	3
109	2	2	2	3	2	2	3	3	2

Рисунок 19 – Запрос 6

Этот запрос показывает, какое количество уроков будет проводиться в определенной аудитории по классам.

### 3.3 Описание форм

Форма «Расписание уроков по каждому класса». Результат формы (смотрите рисунок 20).

Классы

### Классы

Номер класса

Количество человек в классе

Номер урока	День недели	Фамилия	Наименование предмета	Номер аудитории	Номер класса
1	Вторник	Никитина	Литературное чтение	108	1А
1	Понедельник	Иванова	Русский язык	106	1А
1	Пятница	Гумбина	Изобразительное искусство	103	1А
1	Среда	Иванова	Русский язык	106	1А
1	Четверг	Никитина	Литературное чтение	108	1А
2	Вторник	Кузнецова	Музыка	102	1А
2	Понедельник	Агеева	Окружающий мир	101	1А
2	Пятница	Шаталова	Математика	107	1А
2	Среда	Евсеева	Английский язык	109	1А
2	Четверг	Евсеева	Английский язык	109	1А
3	Вторник	Шаталова	Математика	107	1А
3	Понедельник	Федоров	Физическая культура	104	1А
3	Пятница	Иванова	Русский язык	106	1А
3	Среда	Александрова	Технология	105	1А
3	Четверг	Федоров	Физическая культура	104	1А
4	Вторник	Гумбина	Изобразительное искусство	103	1А
4	Понедельник	Кузнецова	Музыка	102	1А

Записи: 1 из 20

Рисунок 20 – Форма 1

Форма №1 показывает расписание уроков на неделю у 1А класса.

Форма «Расписания учителя». Результат формы (смотрите рисунок 21).

Расписание учителя

Код предмета

Фамилия

Код учителя

Имя

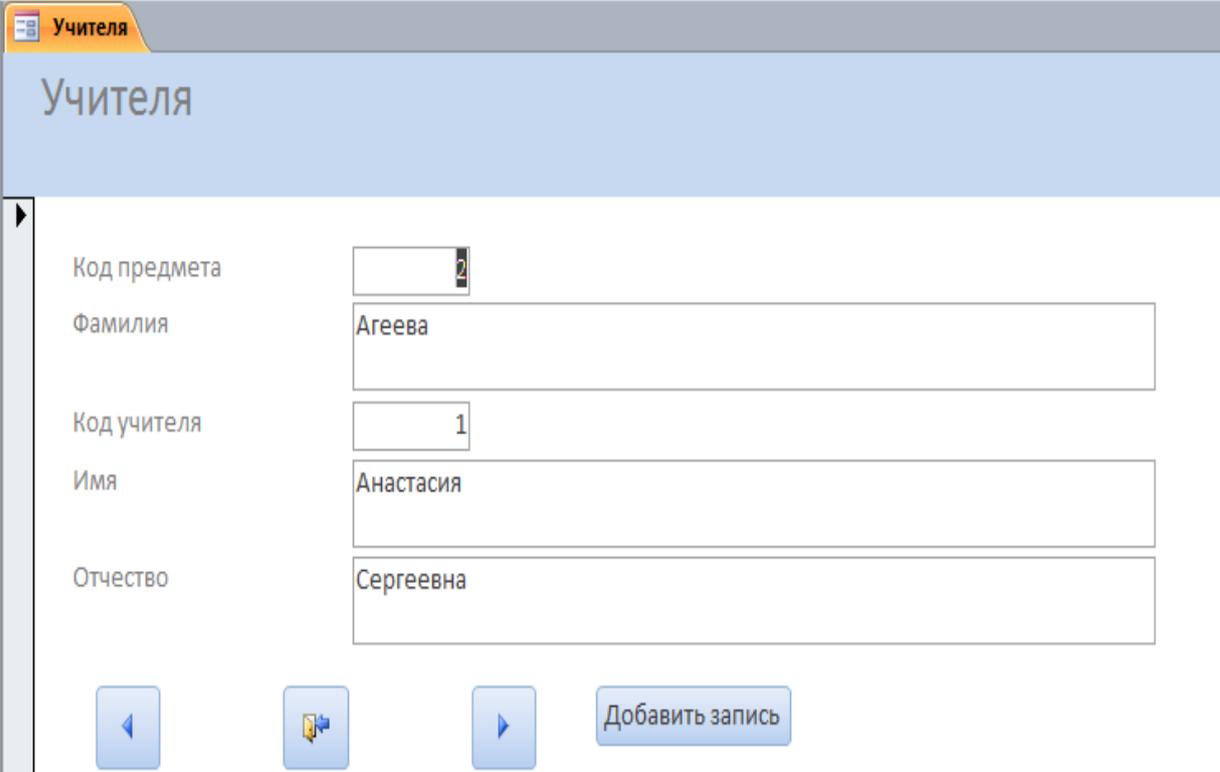
Отчество

День недели	Номер	Время на	Время оконч	Наименование предмета	Код учи	Номер класса
Пятница	1	8:30	9:15	Изобразительное искусство	2	1А
Вторник	1	8:30	9:15	Изобразительное искусство	2	3А
Понедельник	1	8:30	9:15	Изобразительное искусство	2	4А
Среда	1	8:30	9:15	Изобразительное искусство	2	4Б
Четверг	1	8:30	9:15	Изобразительное искусство	2	4В
Вторник	2	9:25	10:10	Изобразительное искусство	2	2А
Четверг	2	9:25	10:10	Изобразительное искусство	2	2А
Пятница	2	9:25	10:10	Изобразительное искусство	2	3А
Понедельник	2	9:25	10:10	Изобразительное искусство	2	3Б
Среда	2	9:25	10:10	Изобразительное искусство	2	4А

Рисунок 21 – Форма 2

Форма №2 показывает расписание Гумбиной Любовь Николаевны на неделю.

Форма «Сведения о каждом учителе». Результат формы (смотрите рисунок 22).



The screenshot shows a web-based form titled "Учителя" (Teachers). The form has a header bar with the title "Учителя" and a sub-header "Учителя". Below the header, there are five input fields with labels on the left: "Код предмета" (Subject code) with value "2", "Фамилия" (Family name) with value "Агеева", "Код учителя" (Teacher code) with value "1", "Имя" (Name) with value "Анастасия", and "Отчество" (Patronymic) with value "Сергеевна". At the bottom of the form, there are four buttons: a left arrow, a right arrow with a document icon, a right arrow, and a button labeled "Добавить запись" (Add record).

Рисунок 22 – Форма 3

Форма №3 показывает сведения об учителе Агеевой Анастасии Сергеевны.

### 3.4 Описание отчетов

Отчет, «На каких уроках каждый учитель имеет «окно» или у него свободный урок». Результат отчета (смотрите рисунок 23).

Зянятость учителей				
Фамилия	День недели	Номер урока	Признак зянятости	
Агеева	Вторник	5	<input type="checkbox"/>	
			Понедельник	<input type="checkbox"/>
				<input type="checkbox"/>
	Пятница	5	<input type="checkbox"/>	
		Среда	5	<input type="checkbox"/>
	Четверг	1	<input type="checkbox"/>	
		5	<input type="checkbox"/>	
		5	<input type="checkbox"/>	
		3	<input type="checkbox"/>	
Александрова		4	<input type="checkbox"/>	
			<input type="checkbox"/>	

Рисунок 23 – Отчет 1

Отчет №1 показывает свободные уроки каждого учителя на неделе.  
 Отчет «Незанятые аудитории по дням недели». Результат отчета (смотрите рисунок 24).

Незанятые аудитории по дням недели			
День недели	Номер урока	Номер аудитории	Признак зянятости
Вторник	5	101	<input type="checkbox"/>
	5	105	<input type="checkbox"/>
	4	105	<input type="checkbox"/>
	2	104	<input type="checkbox"/>
	1	104	<input type="checkbox"/>
	5	102	<input type="checkbox"/>
	5	106	<input type="checkbox"/>
	3	109	<input type="checkbox"/>
Понедельник			

Рисунок 24 – Отчет 2

Отчет №2 показывает, в какой день недели какие аудитории свободны и на каком уроке.

Отчет «Предметы». Результат отчета (смотрите рисунок 25).

Предметы	
Наименование предмета	Код предмета
Русский язык	1
Окружающий мир	2
Физическая культура	3
Математика	4
Музыка	5
Литературное чтение	6
Изобразительное искусство	7
Английский язык	8
Технология	9
	9

Рисунок 25 – Отчет 3

Отчет №3 показывает наименование и код предметов.

Отчет «Расписание «i» учителя по «j» дню». Результат отчета (смотрите рисунок 25).

Введите значение параметра ? X

Введите номер дня недели

Введите значение параметра ? X

Введите код учителя

Расписание				
Номер дня недели	Код учителя	Фамилия	Номер класса	День недели
4	5	Шаталова	2А	Четверг
4	5	Шаталова	2Б	Четверг
4	5	Шаталова	4А	Четверг
4	5	Шаталова	4Б	Четверг
4	5	Шаталова	4В	Четверг

Рисунок 26 – Отчет 4

Отчет №4 показывает расписание уроков на понедельник у учителя Шаталовой.

### **3.5 Рекомендации по администрированию и защите базы данных**

Информация, хранящаяся в БД имеет, огромную ценность и чтобы защитить её, необходимо сделать администрирование, то есть распределение уровня доступа к БД, которое является необходимым условием для доступа к данным, хранимым в ней. Безвозвратная потеря данных является серьезной опасностью. Потерянные вычислительные ресурсы подлежат восстановлению, но если не применять меры по восстановлению и защите потерянных данных, то восстановить их нельзя. Некоторые исследователи сделали статистику, которая показала что при безвозвратной потере корпоративных данных, половина не смогла продолжить свою деятельность [30].

Потеря данных в БД может быть вызвана рядом причин:

1. Сбои оборудования;
2. Физические воздействия на аппаратные средства БД;
3. Стихийные бедствия;
4. Ошибки пользователей;
5. Умышленные вредоносные действия несанкционированных программ или пользователей;
6. Ошибки операционной системы, СУБД или в прикладных программах;
7. Совместное выполнение конфликтных запросов пользователей и др.

База данных имеет объекты, подлежащие защите, например: представления, таблицы, а так же хранимые процедуры и триггеры. Каждый тип объектов имеет определенные действия. Поэтому для них

определены права доступа. На простейшем уровне имеется концепция обеспечения безопасности баз данных. При этом важно поддерживать основные принципы, такие как проверку полномочий и аутентификацию. Первый принцип заключается в том, что каждый пользователь или процесс информационной системы имеет набор действий. Он может выполнять их только лишь к определенным объектам. Аутентификация – это достоверное подтверждение пользователя или процесса, который пытается выполнить уполномоченное действие, только лишь на своем уровне доступа[20].

Система аутентификации имеет своеобразную иерархическую структуру. Всеми правами и полномочиями обладает системный администратор или администратор сервера баз данных. Обычно только этот пользователь создает других пользователей и назначает им определенные полномочия. Система управления базами данных в своих каталогах может хранить описание пользователей и описание их привилегий по отношению к объектам[23].

Схема определения уровня полномочий действует по принципу: в каждом объекте базы данных есть владелец, то есть пользователь, создавший определенный объект. Пользователь имеет определенные права-полномочия к данному объекту. Так же он имеет право предоставлять остальным пользователям полномочия по работе с этим объектом или забрать у них ранее имеющиеся полномочия.

В некоторых СУБД есть еще один уровень иерархии пользователей - это администратор базы данных. В этих системах управления базами данных один сервер может управлять остальными СУБД (например, MS SQL Server, Sybase). В системе управления базами данных Oracle используется однобазовая архитектура, в которую вводится понятие подсхемы - части общей схемы базы данных и вводится пользователь, который имеет доступ к подсхеме. В стандарте SQL не определена команда создания пользователя, но практически во всех коммерческих

СУБД создать пользователя можно не только в интерактивном режиме, но и программно с использованием специальных хранимых процедур. Но для выполнения такой операции пользователь должен иметь право на запуск соответствующей системной процедуры[26].

Существует еще такое понятие, как администратор безопасности, который занимается организацией и управлением системы разграничения доступа. В нее входят: доверительная характеристика (допуск) пользователя, конкретное назначения доступа, регистрация и формирование меток уровня доступа пользователей.

Доступ к массиву учетных записей пользователей имеет только администратор безопасности. Совмещение функций общего администрирования и администрирования безопасности одновременно одним пользователем не допускается, что объективно повышает надежность системы[33].

Авторизацией является предоставление пользователю только тех данных, на которые он имеет право, то есть разграничение прав доступа.

Надежность и сохранность данных - это главные обязанности администратора. Они включают, в свою очередь, решение технологических и профилактических задач:

1. Планирование, конфигурирование и поддержание системы использования устройств внешней памяти, в которых находятся файлы данных;
2. Проверка и поддержание целостности данных;
3. Архивирование и резервирование данных;
4. Восстановление данных после сбоев и повреждений.

Так же одним главных достоинств защиты является логин и пароль. Надежность парольной защиты основывается на хранении их в тайне. При использовании пароля обязательно нужно чтобы он состоял из комбинации букв и цифр или специальных знаков, а так же его длина может иметь не менее шести символов и в нем не должны содержаться пробелы[32].

На сегодняшний момент обеспечение информационной безопасности современных баз данных требует полноценного и комплексного подхода. Оно возможно только при применении широкого спектра защитных средств, которые объединены в продуманной архитектуре. Эти средства в основном не распространены. Часть из них находятся в стадии становления. Чтобы обеспечить реальную безопасность необходимо ежедневно работать над ней, постоянно ее модифицируя, тем самым уменьшая риск потери данных.

## ЗАКЛЮЧЕНИЕ

В данной выпускной квалификационной работе была разработана и реализована база данных «Расписание уроков в школе». При ее создании был проведен анализ деятельности заведующего учебной частью в школе. Так же были рассмотрены основные типы и модели баз данных, этапы проектирования и создания БД, подходы для разработки модели данных, а так же формальные правила для выделения информационных объектов. В качестве средства построения базы данных была использована среда Microsoft Access.

При написании выпускной квалификационной работы была разработана инструкция по эксплуатации базы данных, а так же создана кнопочная форма для наглядности и более быстрой навигации в БД. Для ее создания были разработаны запросы, формы, отчеты и макросы.

Для защиты базы данных были описаны рекомендации по аутентификации, администрированию и защите с помощью организации и управления системами разграничения доступа. Так же были представлены решения технологических и профилактических задач.

При решении поставленных задач были разработаны запросы: «Незадействованные аудитории по дням недели», «Расписание «окон» у учителей на неделю », «Найти на каком уроке и в какой день недели аудитории под номерами 105 и 108 не задействованы», «Составить расписание по определенному учителю и дню недели», «Узнать расписание Агеевой у «2А» и «2Б» классов », «Количество уроков в каждой аудитории по каждому классу».

При решении поставленных задач были разработаны формы: «Расписание уроков по каждому классу», «Расписания учителя», «Сведения о каждом учителе».

При решении поставленных задач были разработаны отчеты: «На каких уроках каждый учитель имеет «окно» или у него свободный урок»,

«Незанятые аудитории по дням недели», простой отчет «Предметы» и «Расписание «i» учителя по «j» дню».

При решении поставленных задач были разработаны макросы для открытия запросов, форм и отчетов.

Все задачи выпускной квалификационной работы были выполнены.  
Цель достигнута.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Агальцов, В.П. Базы данных. В 2-х т. Т. 2. Распределенные и удаленные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 272 с. с. 103-104
2. Власов А.И, Лыткин С.Л., Яковлев В.Л., СУБД ORACLE / МГТУ им. Н.Э.Баумана. – М., 2005г.-230с. с. 27-29
3. Власов А.И., Краткое практическое руководство разработчика информационных систем на базе СУБД Oracle Власов А.И., Лыткин С.Л., - СПб.: Питер, 2005.- 345 с. с. 162-167
4. Галкина Т. И. «Организация и содержание методической работы в современной школе. Книга современного завуча».- Феникс – 2008г.-384с. с.138-140
5. Галкина Т. И., Котельникова В. В. «Новая настольная книга современного завуча».- Феникс – 2009г.-416с. с.203-205
6. Голицына, О.Л. Базы данных: Учебное пособие/ О.Л. Голицына, Н.В. Максимов, И.И. Попов. - М.: Форум, 2012. - 400 с. с.162-163
7. Даммлер М. Microsoft SQL Server 2005 Обзор продукта. – М., 2005г.-548с. с. 245-256
8. Дейт, К. Дж. Введение в системы баз данных, 8-е издание.: Пер. с англ. М.: Издательский дом "Вильямс".-2005.-1328 с. с.87.
9. Дейт. К. Дж. Введение в системы баз данных - Introduction to Database Systems. -- 8-е изд. -- М.: «Вильямс», 2006.- 313с. с.44-45
10. Дереклеева Н.И. «Справочник завуча: Учебно-методическая работа. Воспитательная работа: 5-11 классы».- Вако – 2006 г. 352с. с.167-169
11. Елманова Н., Oracle и Microsoft SQL Server: прошлое, настоящее и будущее. Елманова Н./ Федоров А. М., 2003г.-245с. с. 203-205
12. Есипов А. «Информатика и информационные технологии». – С-П.: «БХВ-Петербург» - 2004 г.-646с. с. 456-465

13. Золотова С.И. Практикум по Access, Москва ФиС, 2010 г.
14. Карпов Б. «Microsoft Access 2000 Справочник» - СПб: Питер, 2012.
15. Карпова Т.С. Базы данных: модели, разработка, реализация. - СПб.: Питер, 2011. - 304с.
16. Карпова, И.П. Базы данных: Учебное пособие / И.П. Карпова. - СПб.: Питер, 2013. - 240 с. с,89-90
17. Кириллов, В.В. Введение в реляционные базы данных. Введение в реляционные базы данных / В.В. Кириллов, Г.Ю. Громов. - СПб.: БХВ-Петербург, 2012. - 464 с. с. 357-359
18. Кошелев, В.Е. Базы данных в ACCESS 2007: Эффективное использование / В.Е. Кошелев. - М.: Бином-Пресс, 2009. - 592 с.
19. Крылова, М.Б. Методический портфель заместителя директора. Из опыта работы / М.Б. Крылова. - Учитель, 2009. - 98 с. с. 61-62
20. Кузин, А.В. Базы данных: Учебное пособие для студ. высш. учеб. заведений / А.В. Кузин, С.В. Левонисова. - М.: ИЦ Академия, 2012. - 320 с. с. 56-58
21. Медведева, О. И. Организация работы школьного методического объединения. Нормативные и инструктивно-методические материалы / О. И. Медведева, Т. А. Кобзарева. - Учитель, 2013. - 136 с. с. 35-36
22. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. - М.: "Академия", 2007. - 586 с. с. 45-48
23. Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 2-е изд. /., Н.А. Олифер - СПб.: Питер, 2004. - 864 с. с. 523-546
24. Пирогов, В.Ю. Информационные системы и базы данных: организация и проектирование: Учебное пособие / В.Ю. Пирогов. - СПб.: БХВ-Петербург, 2009. - 528 с.
25. Роланд, Фред, Д. Основные концепции баз данных.: Пер. с англ.: - Издательский дом "Вильяме", 2002. 256 с.: ил. Парал. тит. англ. с.22

26. Сатлер, О.Н. Администрирование и защита баз данных / О.Н. Сатлер, Н.С. Саляева, А.С. Зубенко.- Виртуальное Моделирование, Прототипирование И Промышленный Дизайн.-Материалы III Международной научно-практической конференции: Электронный ресурс. Общая редакция: В. А. Немтинов.- Тамбов, 2016.- 134-137с.
27. Симоновича С.В Информатика: Базовый курс/Под ред.- СПб.:Питер,2002. 400 с. с. 351-365
28. Советов, Б.Я. Базы данных: теория и практика: Учебник для бакалавров / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской. - М.: Юрайт, 2013. - 463 с.
29. Степанчук Н. А. Модели экологического образования: программы, рекомендации, уроки/ Н. А. Степанчук .- Учитель .-2011.- 295 с. с. 149-150
30. Татарина, Л. П. Справочник руководителя методического объединения/ Л. П. Татарина, С. Д. Аюпова .- Учитель.- 2015.- 291 с. с. 263-264
31. Ульман Дж., Уидом Дж. Введение в системы баз данных. - М.: Лори, 2010. - 374с.
32. Фридланд А.Я. Информатика и компьютерные технологии. А.Я. Фридланд, Л.С. Ханамирова.- М.: Астрель. 2003.204 с. с.66-69
33. Хомоненко А.Д., Цыганков В.М., Мальцев М.Г. Базы данных: Учебник для высших учебных заведений / Под ред. проф. А.Д. Хомоненко. - СПб.: КОРОНА принт, 2009. - 416с.