Департамент образования города Москвы

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ОБРАЗОВАТЕЛЬНЫЙ КОМПЛЕКС

(ГБПОУ МГОК)

|  |  |
| --- | --- |
| УДК 004.467 | УТВЕРЖДАЮ  Заведующий кафедрой  информационных технологий  Полубабкин В.П.\_\_\_\_\_\_\_\_\_\_ «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 2019 г. |

КУРСОВОЙ ПРОЕКТ

ПМ.04 Разработка, администрирование и защита баз данных

по специальности 09.02.07 Информационные системы и программирование

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БАЗ ДАННЫХ ПРИЕМНОЙ КОМИССИИ ТЕХНИКУМА

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гладковская О.А.

(подпись, дата)

Студент группы ИС-217/1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Катков Н.С.

(подпись, дата)

Москва 2019РЕФЕРАТ

Пояснительная записка 22 с., 10 рис., 8 источн., 3 прил.

ПРОГРАММА, КУРСОВОЙ ПРОЕКТ, РАЗРАБОТКА, ПРИЛОЖЕНИЕ, РАБОТА, БАЗА ДАННЫХ, ПРИЕМНАЯ КОМИССИЯ C#, ЯЗЫК, ПРОГРАММИРОВАНИЕ, ТЕХНОЛОГИЯ.

Целью работы является разработка программного продукта «Запись на подготовительные курсы по ЕГЭ».

Разработка происходила на языке C#. Пояснительная записка реализована в Microsoft Office Word 2016.

Предмет исследования – язык программирования С#.

Объект исследования – Техникум.

Результатом работы стала база данных «Запись\_на\_курсы» и одноименный программный продукт.

Степень внедрения – разработанное приложение может использоваться в предприятии проводящей подготовку к сдаче ЕГЭ.

Автор проекта подтверждает, что приведённый в курсовом проекте расчётно-аналитический материал правильно и объективно отражает состояние разрабатываемого объекта, все заимствованные из литературных и других источников теоретические и методологические положения и концепции сопровождаются ссылками на их авторов

**Содержание**

[ВВЕДЕНИЕ 4](#_Toc9154194)

[1. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 6](#_Toc9154195)

[1.1 Описание предметной области и постановка задачи 6](#_Toc9154196)

[1.2 Инфологическое и даталогическое проектирование 7](#_Toc9154197)

[2. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ 12](#_Toc9154198)

[2.1 Реализация физической схемы базы данных 12](#_Toc9154199)

[ЗАКЛЮЧЕНИЕ 14](#_Toc9154200)

[СПИСОК ЛИТЕРАТУРЫ 15](#_Toc9154201)

# ВВЕДЕНИЕ

В наше время во всем требуется удобство, простота и порядок. Без этих трех компонентов невозможно представить наш мир. В программировании порядок важен как нигде так как хаос не может существовать в программе основанной на логике и порядке. Сама суть программирования заключается в создании разноплановых как в сложности так и в назначении программ, которые должны соответствовать неким нормам порядка для того чтобы программу могли использовать не только сами программисты но и пользователи для которых эти программы и разрабатываются. Удобство и простота – это основа грамотного и эффективного использования программных продуктов.[5]

Сейчас нельзя представить сложные компьютерные системы в разработке которых принимает участие всего лишь один человек. Поэтому для того чтобы программисты могли работать в команде требуется использование стандартизированных норм программирования. [2]

Данный программный продукт предназначен для использования администраторами и пользователями. Его актуальность обусловлена тем, что каждый год тысячи российских школьников и просто желающие сдают экзамены ЕГЭ по всем предметам. Организаторы ЕГЭ всегда имеют дело с очень большими объемами данных которые необходимо структурировать, администрировать и максимально автоматизировать. В ходе выполнения данного курсового проекта необходимо создание соответствующего программного продукта и базы данных с которой он будет работать. [6]

Современная база данных обладает определенными свойствами: независимостью данных от использующих их прикладных программ, целостностью, восстанавливаемостью, безопасностью, эффективностью, предельными размерами и эксплуатационными ограничениями. Следовательно, создание базы данных, обладающей такими свойствами, задача достаточно актуальная и полезная. [8]

Объектом являетсяпредметная область «Запись на курсы по сдаче ЕГЭ».

Предметом являетсяавтоматизация информационных процессов предметной области «Курсы по сдаче ЕГЭ».

Цель - разработка базы данных и программного продукта предметной области «Запись на курсы по сдаче ЕГЭ».

Задачи:

* Описание предметной области;
* Инфологическое и даталогическое проектирование;
* Реализация физической схемы базы данных;
* Создание конечного программного продукта;
* Отладка программного продукта и его максимальная автоматизация

Структура работы:курсовая работа состоит из введения, двух глав, заключения, списка литературы. Во введении обоснована актуальность темы исследования. В первой главе описаны предметная область, построены инфологическая и даталогическая модели базы данных. Вторая глава посвящена описанию процесса разработки программного продукта для выбранной предметной области.

# 1. Проектирование базы данных

## Описание предметной области и постановка задачи

В данном курсовом проекте, в соответствии с темой, необходимо разработать программный продукт для записи на курсы по подготовке к ЕГЭ.

Единый Государственный Экзамен – это экзамен сдающийся школьниками по окончанию 11 класса и всеми желающими. Экзамен представлен целым набором дисциплин которые проходят в школе и могут понадобиться при поступлении в институт. Ежегодно тысячи людей сдают ЕГЭ и каждому необходимо зарегистрироваться на сдачу определенного или определенных экзаменов. Экзамены представляют собой проверку знаний по выбранным учеником предметам за весь период обучения в школе, разумеется сдать экзамен такого уровня без предварительной подготовки невероятно сложно и многие ученики нанимают репетиторов для подготовки или вступают в соответственные группы по подготовке к экзаменам. Данная программа предназначена для облегчения записи на курсы по подготовке к сдаче ЕГЭ, учеников и всех желающих вспомнить или выучить предмет.

Готовый программный продукт, то есть база данных с программным продуктом, ориентирована на эксплуатацию абитуриентами при записи в группу по подготовке к ЕГЭ и администратором ответственным за использование данного программного продукта.

## 1.2 Инфологическое и даталогическое проектирование

Инфологическая модель (информационно-логи­ческая модель) — ориентированная на человека и не зависимая от типа СУБД модель предметной области, определяющая совокупности информационных объектов, их атрибутов и отношений между объектами, динамику изменений предметной области, а также харак­тер информационных потребностей пользователей. Инфологическая модель предметной области может быть описана моделью "сущность—связь" (моделью Чена), в основе которой лежит деление реального мира на отдель­ные различимые сущности, находящиеся в определенных связях друг с другом, причем обе категории — сущность и связь полагаются первичными, неопределенными по­нятиями.

Даталогические модели — модели данных, ориентированные на выбранный тип СУБД: внутренняя, концептуальная, внешняя.

Внутренняя модель — модель данных низшего (физического) уровня в архитектуре СУБД, отражающая представление данных во внешней памяти и методы до­ступа к ним.

Внешняя модель — модель данных внешнего уровня в архитектуре СУБД, отражающая представление пользователя о базе данных (подсхема базы данных и ее описание).

Концептуальная модель — информационная модель предметной области в терминах конкретной СУБД, содержащая полный набор данных и связей между ними. В архитектуре СУБД представляет промежуточный между внешним и внутренним уровнями.

Схема базы данных — описание даталогических моделей в терминах СУБД (часто используется как си­ноним модели данных).

Основные этапы, на которые разбивается процесс проектирования базы данных информационной системы:

Концептуальное проектирование - сбор, анализ и редактирование требований к данным. Для этого осуществляются следующие мероприятия:

- обследование предметной области, изучение ее информационной структуры

- выявление всех фрагментов, каждый из которых характеризуется пользовательским представлением, информационными объектами и связями между ними, процессами над информационными объектами

- моделирование и интеграция всех представлений

По окончании данного этапа получаем концептуальную модель, инвариантную к структуре базы данных. Часто она представляется в виде модели "сущность-связь". Представление аналитика (сущности, атрибуты, связи).

Логическое проектирование - преобразование требований к данным в структуры данных. На выходе получаем СУБД-ориентированную структуру базы данных и спецификации прикладных программ. На этом этапе часто моделируют базы данных применительно к различным СУБД и проводят сравнительный анализ моделей.

Представление программиста (записи, элементы данных, связи между записями).

Физическое проектирование - определение особенностей хранения данных, методов доступа и т.д.

Представление администратора (группирование данных, индексы, методы доступа).

Конкретный вид и содержание концептуальной модели базы данных определяется выбранным для этого формальным аппаратом. Обычно инфологическая модель строится мной в онлайн - программе dia.

Чаще всего концептуальная модель базы данных включает в себя:

* описание информационных объектов или понятий предметной области и связей между ними.
* описание ограничений целостности, т.е. требований к допустимым значениям данных и к связям между ними. [6]

Нормализация отношений - процесс преобразования отношений базы данных к виду, отвечающему нормальным формам, называется нормализацией.

Нормализация отношений (таблиц) — одна из основополагающих частей теории реляционных баз данных. Нормализация имеет своей целью избавиться от избыточности в отношениях и модифицировать их структуру таким образом, чтобы процесс работы с ними не был обременён различными посторонними сложностями. При игнорировании такого подхода эффективность проектирования стремительно снижается, что вкупе с прочими подобными вольностями может привести к критическим последствиям.  
Любому специалисту, по роду своей деятельности так или иначе связанному с проектированием реляционных баз данных, полезно понимать и уметь осуществить нормализацию отношений.

Существует несколько нормальных форм, но не всегда специалисту требуются, исключительно, все. Разберём некоторые из них.

Первая нормальная форма

Отношение находится в первой нормальной форме (далее 1НФ), если все его атрибуты атомарны, то есть, если ни один из его атрибутов нельзя разделить на более простые атрибуты, которые соответствуют каким-то другим свойствам описываемой сущности.

Будем называть исходное отношение основным, а значение неатомарного атрибута — подчинённым.

Для того, чтобы нормализовать исходное отношение, атрибуты которого неатомарны, необходимо объединить схемы основного и подчинённого отношений. Кроме того, если, например, таблица, соответствующая, ненормализованному отношению уже содержится в БД и заполнена информацией, задача усложняется тем, что значение неатомарного атрибута может в свою очередь содержать несколько кортежей.

Вторая нормальная форма

Ясно, что отношение, находящееся в 1НФ, также может обладать избыточностью. Для её устранения предназначена вторая нормальная форма. Но прежде чем приступить к её описанию, сначала следует выявить недостатки первой.

Далее представлена нормализация отношений предметной области «Приемной комиссии техникума».

Третья нормальная форма  ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) Third normal form; сокращённо 3НФ)  — одна из возможных [нормальных форм](https://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%84%D0%BE%D1%80%D0%BC%D0%B0) [отношения](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BD%D0%BE%D1%88%D0%B5%D0%BD%D0%B8%D0%B5_(%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C)) в [реляционной базе данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). 3NF была изначально сформулирована [Э. Ф. Коддом](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%B4%D0%B4,_%D0%AD%D0%B4%D0%B3%D0%B0%D1%80) в [1971 году](https://ru.wikipedia.org/wiki/1971_%D0%B3%D0%BE%D0%B4).

Переменная отношения R находится в 3NF тогда и только тогда, когда выполняются следующие условия:

R находится во [второй нормальной форме](https://ru.wikipedia.org/wiki/%D0%92%D1%82%D0%BE%D1%80%D0%B0%D1%8F_%D0%BD%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%84%D0%BE%D1%80%D0%BC%D0%B0).

Ни один не ключевой атрибут R не находится в транзитивной [функциональной зависимости](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) от [потенциального ключа](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D0%B5%D0%BD%D1%86%D0%B8%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BB%D1%8E%D1%87) R.

Пояснения к определению:

Не ключевой атрибут отношения R — это атрибут, который не принадлежит ни одному из потенциальных ключей R.

Даталогическое проектирование — создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных даталогическая модель — набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

Преобразование инфологической модели в даталогическую модель, как правило, осуществляется по формальным правилам. Этот этап может быть в значительной степени автоматизирован.

На этапе даталогического проектирования учитывается специфика конкретной модели данных, но может не учитываться специфика конкретной СУБД. [4]

В соответствии с изображением 1 приведена даталогическая модель базы данных записи на курсы по подготовке к ЕГЭ.

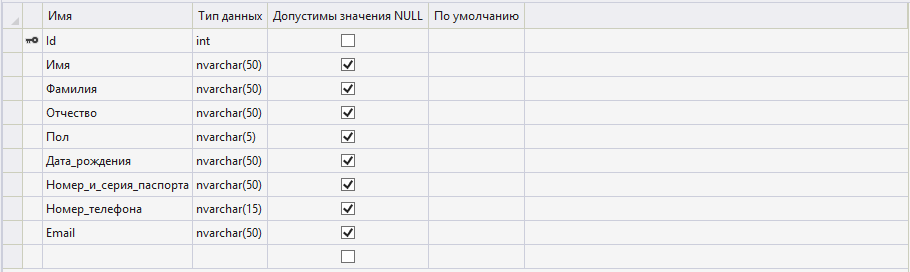


Рисунок 1.1 – Даталогическая модель базы данных

В первой главе были рассмотрены предметная область и описание программы, необходимой для ее реализации. Также для создания СУБД были выполнены инфологическая и даталогическая модели БД предметной области «Запись на курсы по подготовке к ЕГЭ».

# 2. Программная реализация базы данных

## 2.1 Реализация физической схемы базы данных

Любая база данных состоит из таблиц. Таблица – элемент базы данных, состоящий из отдельных строк и столбцов. Строки называются записями, а столбцы – полями. В программе MS SQL Server можно создать таблицу в режиме конструктора. Туда необходимо ввести название поля и тип данных. Чтобы изменить записи, необходимо перейти к таблице.

Для того, чтобы базу данных реализовать в формате приложения, необходимо создать базу данных в одной из СУБД.

В соответствии с изображением 2.1 представлена таблица базы данных «Запись на курсы по подготовке к ЕГЭ».

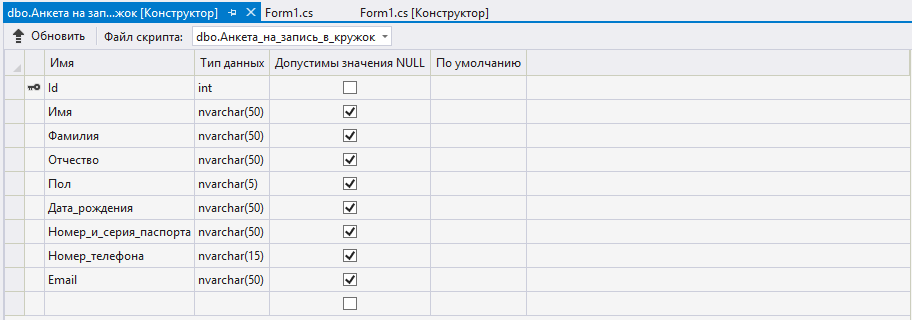


Рисунок 2.1 – Таблица базы данных «Анкета на запись на курсы по подготовке к ЕГЭ»

Создание первичных и внешних ключей

Таблица обычно имеет столбец или комбинацию столбцов, значения которых уникально идентифицируют каждую строку в таблице. Этот столбец (или столбцы) называется первичным ключом таблицы и нужен для обеспечения ее целостности. Если в первичный ключ входит более одного столбца, то значения в пределах одного столбца могут дублироваться, но любая комбинация значений всех столбцов первичного ключа должна быть уникальна.

Ограничение внешнего ключа - это основной механизм для поддержания ссылочной целостности между таблицами реляционной базы данных. Столбец дочерней таблицы, определенный в качестве внешнего ключа в параметре Внешний ключ, применяется для ссылки на столбец родительской таблицы, являющийся в ней первичным ключом.

Данные в столбцах, определенных в качестве внешнего ключа, могут принимать только такие же значения, какие находятся в связанных с ним столбцах первичного ключа родительской таблицы. Совпадение имен столбцов для связи дочерней и родительской таблиц необязательно.

Первичный ключ может быть определен для столбца с одним именем, в то время как столбец, на который наложено ограничение внешнего ключа, может иметь совершенно другое имя. Единственным требованием остается соответствие столбцов по типу и размеру данных.

В соответствии с изображением 2.2 представлена таблица, которую я присоединил к Microsoft Visual Studio 2017 Community.

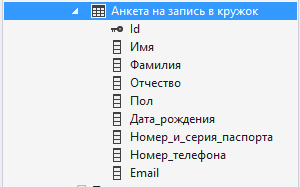


Рисунок 2.2 – Структура таблицы «Приемная комиссия техникума»

Во второй главе были разработаны и реализованы схемы базы данных, таблицы базы данных и т.д. Также были определены ключи таблицы, структура и макет программного проекта.

# ЗАКЛЮЧЕНИЕ

Выполнение данной курсовой работы было нацелено на разработку базы данных и программного продукта для предметной области «Запись на курсы по подготовке к ЕГЭ».

Для этого было необходимо изучить предметную область «Запись на курсы по подготовке к ЕГЭ», провести анализ СУБД, выполнить логическое проектирование базы данных, создать физическую реализацию данной базы данных и программного продукта.

В первой части курсового проекта была рассмотрена предметная область и выполнены элементы, необходимые для создания БД.

Во второй части была реализована схема базы данных и необходимые компоненты для облегчения работы рядового пользователя.

База данных была разработана в [Microsoft Visual Studio 2017 Community](https://info-comp.ru/programmirovanie/479-visual-studio-2015-community.html)

на основе интерфейса. Microsoft Visual Studio — линейка продуктов компании [Microsoft](https://ru.wikipedia.org/wiki/Microsoft), включающих [интегрированную среду разработки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) программного обеспечения и ряд других инструментальных средств.

Из этого следует, что поставленные в начале работы цели были выполнены в полном объеме, представлена рабочая версия программного продукта и присоединенная к нему база данных.

# СПИСОК ЛИТЕРАТУРЫ

1. Агальцов В.П. Базы данных. В 2-х т.Т. 2. Распределенные и удаленные базы данных.— М.: ФОРУМ: ИНФРА-М, 2016
2. [Бьюли А.](javascript:__doPostBack('ctl00$cph$ucGoodCard$AuthorSpecializedSearch$lbt_Search','')) Изучаем SQL.— СПб.: [Символ-Плюс](javascript:__doPostBack('ctl00$cph$ucGoodCard$PublisherSpecializedSearch$lbt_Search','')), 2017
3. Васильев А.А., Избачков Ю.С., Петров В.Н., Телина И.С. Информационные системы.— СПб.: Питер, 2017
4. Гагарина Л.Г., Киселев Д.В., Федотова Е.Л. Разработка и эксплуатация автоматизированных информационных систем — М.: Издательский Дом "Форум", 2016.
5. Емельянова Н.З., Партыка Т.Л., Попов И.И. Проектирование информационных систем — М.: ФОРУМ: ИНФРА-М, 2017.
6. Кумскова И.А. Базы данных. – Издательство: [Кнорус](http://www.labirint.ru/pubhouse/1446/), 2017
7. Рудаков А. В. Технология разработки программных продуктов: учебное пособие. — М.: Академия, 2016
8. [Шустова Л.И.](http://bookradar.org/search/?q=%D0%A8%D1%83%D1%81%D1%82%D0%BE%D0%B2%D0%B0%20%D0%9B.%D0%98.) [Базы данных. Учебник](http://bookradar.org/books/%D0%91%D0%B0%D0%B7%D1%8B%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85.%20%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D0%B8%D0%BA). — М.: ФОРУМ: ИНФРА-М, 2016.

ПРИЛОЖЕНИЕ А

(обязательное)

Техническое задание

Основным назначением разработки является создание приложения для приемной комиссии техникума (ГБПОУ МГОК), на языке программирования C#. Также, для создания базы данных, используется система управления базами данных (СУБД) MS SQL Server.

Целью создания приложения является: приложение для записи в группу для подготовки к ЕГЭ, на языке программирования C#.

Целевой аудиторией приложения являются:

* школьники;
* студенты;
* преподаватели;
* любой желающий вспомнить школьную программу выбранного предмета.

Первичная разработка и добавление контента (содержимого) приложения должна производиться силами исполнителя при непосредственном участии Заказчика. Заказчик предоставляет все необходимые исполнителю текстовые и графические материалы, а также комментарии, касающиеся их содержания, объема, оформления и размещения.

Приложение должно иметь русскоязычную версию. Все страницы приложения должны иметь соответствующие ссылки.

Система управления контентом приложения должна обеспечить администратору программы возможность выполнения следующих действий:

* добавление и удаление;
* редактирование;
* редактирование настроек и пунктов;

Изменение дизайна и структуры приложения, а также доработка существующего и создание нового функционала должны происходить в рамках процедур поддержки приложения исполнителем либо в соответствии с отдельными договорами на указанные виды работ. Компоновка страниц приложения должна обеспечивать автоматическое масштабирование страниц.

Структура приложения:

* фамилия;
* имя;
* отчество;
* пол;
* номер телефона;
* E-mail;
* Серия и номер паспорта;
* Дата рождения;
* активная кнопка – добавить;
* Активная кнопка – обновить;
* Активная кнопка – удалить.

ПРИЛОЖЕНИЕ Б

(рекомендуемое)

Код программного продукта



Рисунок Б.1 – Код для запуска программного продукта

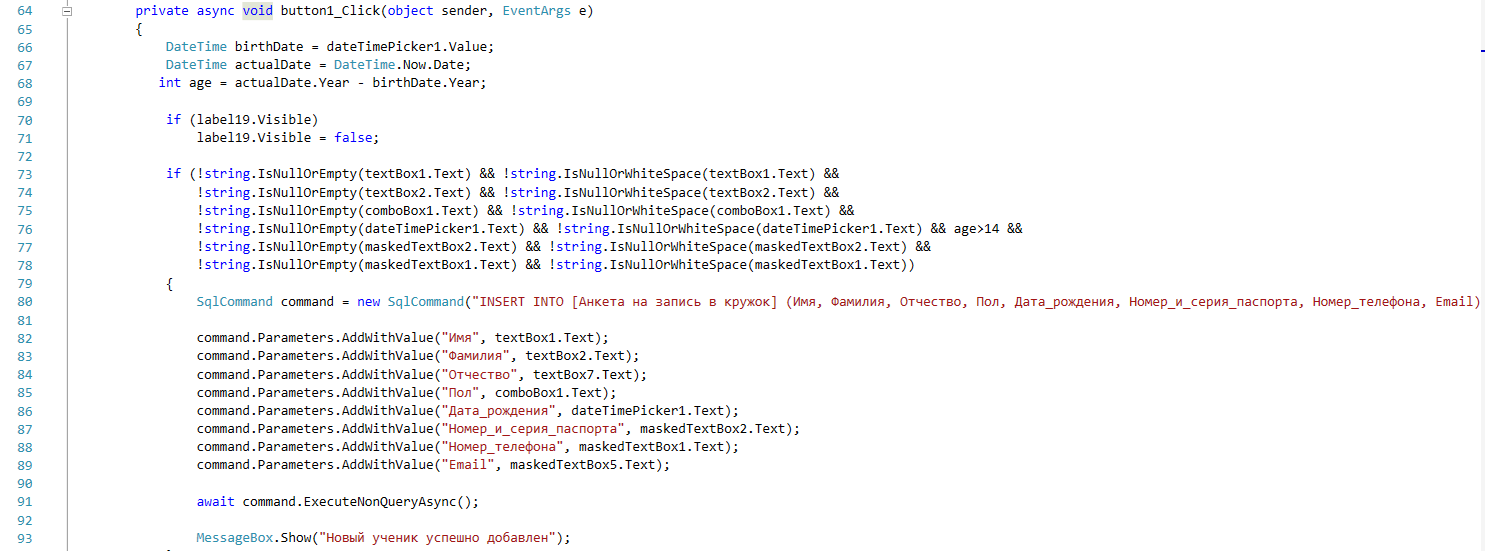


Рисунок Б.2 – Код кнопки «Добавить ученика»

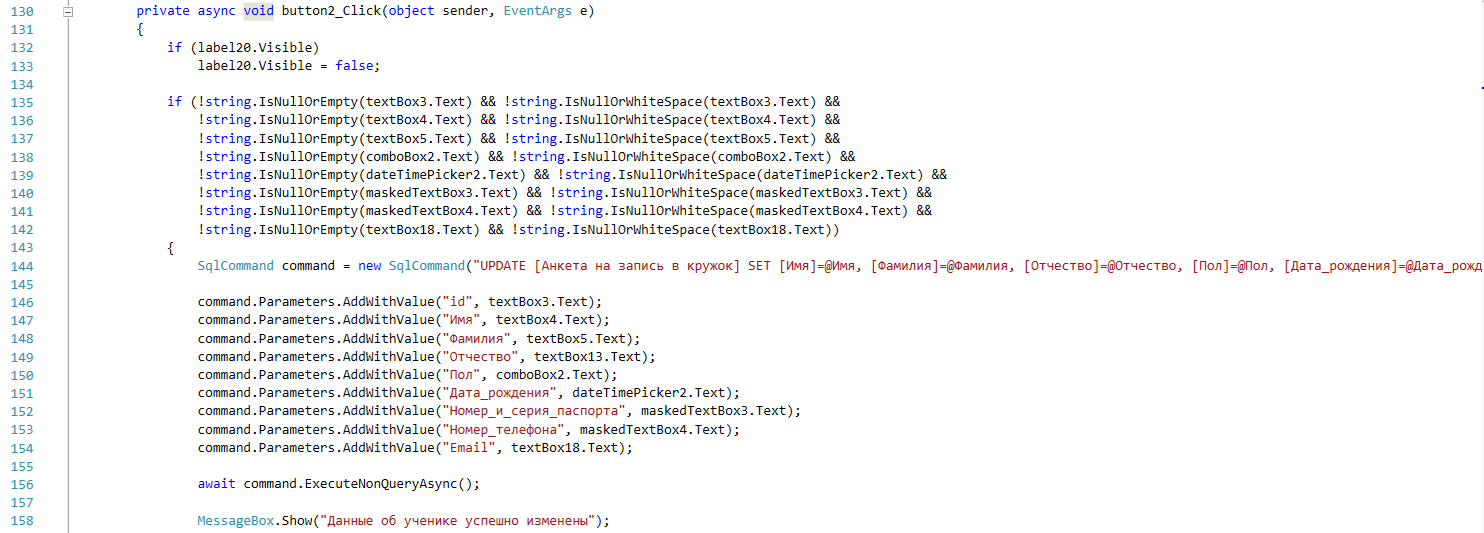


Рисунок Б.3 – Код кнопки «Изменить»

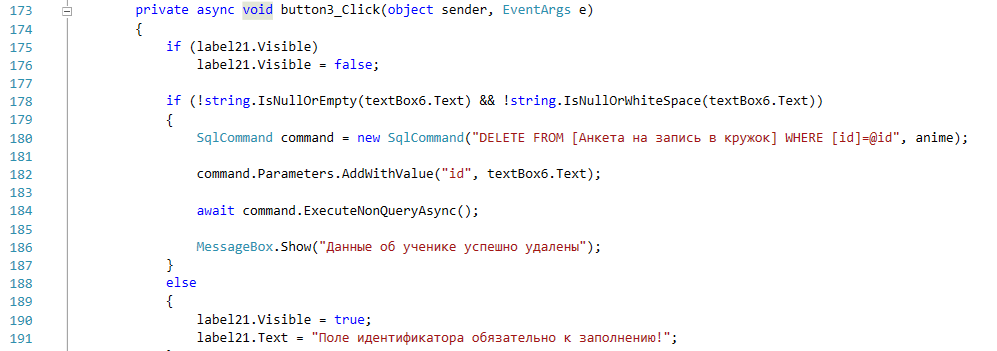


Рисунок Б.4 – Код кнопки «Удалить»

ПРИЛОЖЕНИЕ В

(рекомендуемое)

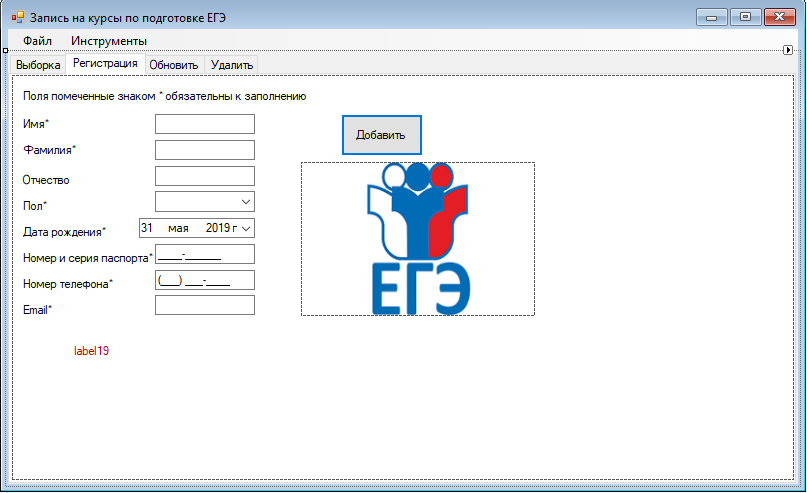


Рисунок В.1 – Макет приложения базы данных «Запись на курсы по подготовке к ЕГЭ»

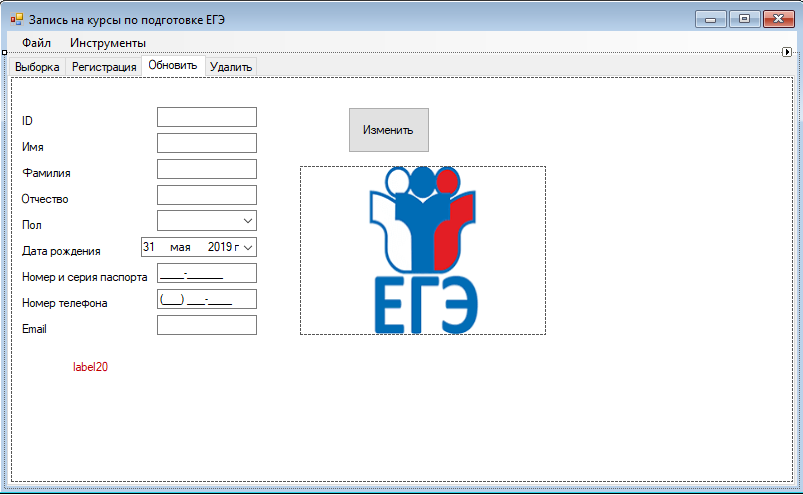


Рисунок В.3 – Готовый программный продукт