



МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ  
ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«ТОСНЕНСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА  
№3 ИМЕНИ ГЕРОЯ СОВЕТСКОГО СОЮЗА С.П.ТИМОФЕЕВА»

**ПРОЕКТНАЯ РАБОТА**

**ПО ИНФОРМАТИКЕ**

**«РАЗРАБОТКА ПРИЛОЖЕНИЙ С  
ИСПОЛЬЗОВАНИЕМ БАЗ ДАННЫХ»**

Выполнил:  
Ученик 10а класса

Руководитель проекта:  
Методист по информационным технологиям  
Вавилов Алексей Николаевич

г.Тосно

2023 г

# 1.

## ОГЛАВЛЕНИЕ

1.	Оглавление _____	2
2.	Введение _____	3
3.	Основная часть	
3.1	Теоретическая часть	
3.1.1	Программирование _____	4
3.1.2	C# _____	5
3.1.3	WPF _____	6
3.1.4	Базы данных _____	7
3.1.5	SQL _____	8
3.1.6	Microsoft SQL Server _____	9
3.1.7	Entity Framework Core _____	10
3.2	Практическая часть	
3.2.1	Программа «Путь к знаниям» _____	11
3.2.2.	Создание программы _____	11-13
4.	Заключение _____	14
5.	Список используемых источников _____	15
6.	Приложение _____	16-29

## **2. Введение.**

Программирование — процесс написания программ для компьютеров при помощи специальных языков программирования. Успех в области программирования по большей части определяется правильным выбором конкретных инструментов под определенную задачу.

**Актуальность** выбранной темы заключается в том, что значение тестовых технологий в образовательном процессе неуклонно растёт, поэтому разработка программы для создания и проведения тестов имеет важное значение как для учеников при подготовке к экзаменам, так и для учителей на своих уроках.

**Целью** моего проекта является изучить язык программирования C#, научиться пользоваться базой данных и создать приложение для проведения тестов.

### **Задачи проекта:**

1. Изучить возможности программирования.
2. Изучить язык программирования C#.
3. Научиться пользоваться СУБД Microsoft SQL Server.
4. Разработать программный продукт.

### **Этапы проекта:**

#### **1.Подготовительный этап.**

Выбор темы. Подбор материалов. Посещение библиотеки с целью изучения книг и журнал по выбранной теме, а также изучение интернет источников.

#### **2.Основной этап.**

Изучение подобранных материалов. Выбор наиболее важных фактов о языке программирования C# и создание программного продукта.

#### **3.Заключительный этап.**

Анализ, проверка и обобщение результатов проделанной работы.  
Литературное оформление результатов исследования.

### **3.1. Теоретическая часть.**

#### **3.1.1. Программирование**

Программирование – это процесс создания компьютерных программ. Программирование основывается на использовании языков программирования, на которых записываются исходные тексты программ.

Изначально программирование имело крайне примитивный вид и практически не имело отличий от упорядоченного бинарного кода с формализованным подходом. Очевидных и естественных удобств для программиста не существовало, он обязан был обладать знаниями числовых кодов для каждой команды машины. Даже распределение памяти для выполнения команд ложилось на специалиста.<sup>1</sup>

В 1949 был разработан язык программирования Ассемблер. Для отображения переменных стали использоваться символьные наименования. Вместо числовых операций человеку достаточно знать их названия.

Со временем программирование развивалось и появлялись новые языки программирования. В наше время языки программирования делятся на 3 типа: процедурные, декларативные и объектно-ориентированные (приложение № 1). Процедурные языки требуют от программиста детального описания того, как решать задачу. К ним можно отнести Basic, Pure Basic, Pascal, C. Декларативное программирование - это стиль программирования, в котором задаётся спецификация решения задачи: описывается, что представляет собой проблема и ожидаемый результат, но без описания способа достижения этого результата.<sup>2</sup> Примерами этого типа языков являются Haskell, F#, HTML, SQL. Объектно-ориентированное программирование (ООП) – концепция, которая призвана облегчить разработку сложных систем, за счет введения новых понятий, более

---

<sup>1</sup> Большая компьютерная энциклопедия. 2007, с. 205

<sup>2</sup> Декларативное программирование [tproger.ru]

приближенных к реальному миру. К этим языкам относятся : C++, C#, Python, Ruby, JavaScript, Swift, Java, Go, Kotlin, PHP.

### 3.1.2. C#

C# (произносится как "си шарп") - современный объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать разные типы безопасных и надежных приложений, выполняющихся в .NET. C# относится к известному семейству языков C.

C# был разработан в 1998 - 2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft .NET Framework и .NET Core.

Язык C# практически универсален. Его можно использовать для создания любого ПО: приложений для Windows (Windows Forms, WPF, UWP), мобильных приложений для Android и iOS (Xamarin Forms, .NET MAUI), веб-приложений и сайтов (ASP.NET Core), видеоигр (Unity) и т.д.

C# имеет много функций, которые позволяют создавать надежные и устойчивые приложения. Сборщик мусора .NET управляет выделением и освобождением памяти для приложения. Типы, допускающие значение null, обеспечивают защиту от переменных, которые не ссылаются на выделенные объекты. Обработка исключений предоставляет структурированный и расширяемый подход к обнаружению ошибок и восстановлению после них. Лямбда-выражения поддерживают приемы функционального программирования. Синтаксис LINQ создает общий шаблон для работы с данными из любого источника. Поддержка языков для асинхронных операций предоставляет синтаксис для создания распределенных систем. В C# имеется единая система типов. Все типы C#, включая типы-примитивы, такие как int и double, происходят от одного корневого типа object. Все типы используют

общий набор операций, а значения любого типа можно хранить, передавать и обрабатывать схожим образом (приложение № 2).<sup>3</sup>

### **3.1.3. WPF.**

Windows Presentation Foundation (WPF) — это платформа пользовательского интерфейса для создания клиентских приложений для настольных систем с помощью языка C#. Платформа разработки WPF поддерживает широкий набор компонентов для разработки приложений, включая модель, ресурсы, элементы управления, графику, макет, привязки данных, документы и безопасность (приложение № 3).

В основе WPF лежит мощная инфраструктура, основанная на DirectX — API-интерфейсе графики с аппаратным ускорением, который используется в современных компьютерных играх. Это означает возможность применения развитых графических эффектов, не платя за это производительностью.

Из преимуществ WPF можно выделить: веб-подобная модель компоновки, богатая модель рисования, развитая текстовая модель, анимация, поддержка аудио и видео, стили и шаблоны, команды и приложения на основе страниц.<sup>4</sup>

XAML (eXtensible Application Markup Language) - язык разметки, используемый для инициализации объектов в технологиях на платформе .NET. Применительно к WPF данный язык используется прежде всего для создания пользовательского интерфейса декларативным путем.

Использование XAML в своих проектах позволяет отделить графический интерфейс от логики приложения, а также XAML компактен, понятен, и код написанный на нем относительно легко поддерживать.

Основным паттерном программирования, используемым при разработке в среде WPF, является MVVM (приложение № 4). Паттерн MVVM (Model-View-ViewModel) позволяет отделить логику приложения от визуальной части

---

<sup>3</sup> Краткий обзор языка C#[[learn.microsoft.com](http://learn.microsoft.com)]

<sup>4</sup> Преимущества WPF[[professorweb.ru](http://professorweb.ru)]

(представления). Данный паттерн является архитектурным, то есть он задает общую архитектуру приложения.<sup>5</sup>

### **3.1.4. Базы данных.**

База данных — это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных (СУБД). Данные вместе с СУБД, а также приложения, которые с ними связаны, называются системой баз данных, или, для краткости, просто базой данных.<sup>6</sup>

Данные в наиболее распространенных типах современных баз данных обычно хранятся в виде строк и столбцов формирующих таблицу. Этими данными можно легко управлять, изменять, обновлять, контролировать и упорядочивать. В большинстве баз данных для записи и запросов данных используется язык структурированных запросов (SQL).

Существует и другие типы баз данных, например: key-value базы данных, документно-ориентированные базы данных, графовые базы данных, колоночные базы данных (приложение № 5).

Основными преимуществами баз данных являются:

1. В базе данных можно хранить очень огромное количество данных — миллиарды и триллионы записей;
2. Базы помогают защищать данные - они позволяют давать доступ к данным только определенному кругу лиц. При этом можно ставить ограничения, кому к каким данным можно давать доступ и какого типа доступ, только чтение или редактирование тоже;
3. Базы данных могут помогать следить за правильностью данных с помощью различного вида проверок;

---

<sup>5</sup> Паттерн MVVM[metanit.com]

<sup>6</sup> Что такое база данных[www.oracle.com]

4. Базы данных могут позволять большому количеству людей одновременно взаимодействовать с данными.<sup>7</sup>

### 3.1.5. SQL.

Язык SQL — это не процедурный язык, который является стандартным средством работы с данными во всех реляционных СУБД. Операторы (команды), написанные на этом языке, лишь указывают СУБД, какой результат должен быть получен, но не описывают процедуру получения этого результата. СУБД сама определяет способ выполнения команды пользователя.<sup>8</sup>

Команды SQL делятся на несколько типов (приложение № 6):

Язык определения данных (DDL) относится к командам SQL, которые проектируют структуру базы данных. Программисты используют DDL для создания и изменения объектов базы данных. Они применяют команду CREATE для создания объектов, таких как таблицы, представления и индексы.

Язык запросов данных (DQL) состоит из инструкций по извлечению данных, хранящихся в реляционных базах данных. Программные приложения используют команду SELECT для фильтрации и возврата определенных результатов из таблицы SQL.

В инструкциях языка управления данными (DML) записывается новая информация или изменяются существующие записи в реляционной базе данных. Например, команда INSERT используется для сохранения новой записи в таблице.

Администраторы используют язык управления данными (DCL) для управления или авторизации доступа к базе данных для других пользователей. Например, они могут использовать команду GRANT, чтобы разрешить определенным приложениям манипулировать одной или несколькими таблицами.

---

<sup>7</sup> Введение в базы данных[habr.com]

<sup>8</sup> PostgreSQL. Основы языка SQL. 2008, стр.18



Реляционный механизм использует язык управления транзакциями (TCL) для автоматического внесения изменений в базу данных. Например, база данных использует команду ROLLBACK для отмены ошибочной транзакции.<sup>9</sup>

### **3.1.6. Microsoft SQL Server.**

Microsoft SQL Server — система управления реляционными базами данных, разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия.

Преимущества SQL Server:

1. Тесная интеграция с операционной системой Windows.
2. Высокая производительность, отказоустойчивость.
3. Поддержка многопользовательской среды.
4. Расширенные функции резервирования данных.
5. Работа с удаленным подключением.<sup>10</sup>

Специфика работы сервера базы данных SQL Server заключается в транзакционной обработке данных. Это означает, что по каждому запросу от СУБД обрабатывается и сохраняется небольшое количество информации.

Для удобного управления базами данных и различными опциями и настройками в MS SQL Server существует специальное средство администрирования, которое называется SQL Server Management Studio (SSMS). Данную программу можно использовать для создания баз данных и их таблиц, написания и выполнения запросов, а также для много другого (приложение № 7).

При создании T-SQL язык SQL был расширен такими дополнительными возможностями как: управляющие операторы, локальные и глобальные

---

<sup>9</sup> Что такое SQL[aws.amazon.com]

<sup>10</sup> Что такое MS SQL Server и чем она отличается от других СУБД [timeweb.com]

переменные, различные дополнительные функции для обработки строк, дат, математики, поддержка аутентификации Microsoft Windows.<sup>11</sup>

### **3.1.7. Entity Framework Core.**

Entity Framework представляет ORM-технология (object-relational mapping - отображения данных на реальные объекты) от компании Microsoft для доступа к базе данных. Entity Framework Core позволяет абстрагироваться от самой базы данных и ее таблиц и работать с данными как с объектами классов независимо от типа хранилища. Если на физическом уровне мы оперируем таблицами, индексами, первичными и внешними ключами, то на концептуальном уровне, который нам предлагает Entity Framework, мы уже работаем с объектами (приложение № 8).

Любая сущность, как и любой объект из реального мира, обладает рядом свойств. Например, если она описывает человека, то мы можем выделить такие свойства, как имя, фамилия, рост, возраст. И у каждой сущности может быть одно или несколько свойств, которые будут отличать эту её от других и будут уникально определять эту сущность. Подобные свойства называют ключами.

При этом сущности могут быть связаны ассоциативной связью один-ко-многим, один-ко-одному и многие-ко-многим, подобно тому, как в реальной базе данных происходит связь через внешние ключи.

Технология доступа к данным Entity Framework Core работает поверх платформы .NET, поэтому может использоваться на различных платформах стека .NET. Это и стандартные платформы типа Windows Forms, консольные приложения, WPF, UWP и ASP.NET Core. При этом кроссплатформенная природа EF Core позволяет задействовать ее не только на ОС Windows, но и на Linux и Mac OS.<sup>12</sup>

---

<sup>11</sup>Transact-SQL [ru.wikipedia.org]

<sup>12</sup> Что такое Entity Framework Core [metanit.com]

Для работы с Entity Framework Core используется технология LINQ to Entities. LINQ (Language Integrated Query) — это технология, представляющая собой набор функций в .NET, которые позволяют писать структурированные запросы к базе данных. LINQ использует похожие на SQL выражения языка C# для получения данных из базы данных.

## **3.2 Практическая часть.**

### **3.2.1 Программа «Путь к знаниям».**

В ходе реализации моей практической работы я создал приложение для проведения тестов, а также для предоставления ученикам возможности тренироваться решать задачи по различным предметам.

При запуске приложения пользователь попадает на главную страницу. На ней он может получить информацию о приложении, зарегистрироваться как ученик или учитель и войти в свой аккаунт.

Ученики на своей странице могут увидеть и изменить информацию о себе, а также просмотреть свои группы и своих учителей. Также ученики могут зайти на вкладку с заданиями и решить их, чтобы подготовиться к тестам или просто потренироваться по нужной им теме. На странице тестов отображаются уже решенные задания и их результаты, а также тесты, которые ученик должен выполнить.

На своей странице учителя также могут редактировать данные о себе. На этой вкладке учителя могут увидеть свои классы, изменить информацию о них, просмотреть и добавить учеников, удалить ненужный класс. На второй странице можно создать группу, введя её название и номер класса. На последней вкладке учителя создают тесты для своих классов, при этом либо сами выбирают задания для теста, либо вопросы генерируются автоматически.

### **3.2.2 Создание программы.**

Создание приложения я начал с проектирования базы данных. Она состоит из таблиц, содержащих информацию о пользователях, тестах и их

результатах, заданиях и ответах. После проектирования таблиц я продумал связи между ними. Базу данных я реализовал через фреймворк Entity Framework Core. В С# я написал классы, параметрами которых стали колонки таблиц. Также для каждой этой сущности я реализовал интерфейс INotifyPropertyChanged для будущего связывания данных базы с формами. Затем я создал класс ApplicationContext, реализовав DbContext, где прописал каждую сущность как таблицу и с помощью Fluent Api я прописал первичные ключи, их автоматическую генерацию и установил связи между таблицами. После создания базы данных я приступил к логике программы.

Для дальнейшей разработки я использовал паттерн MVVM. Так как модели уже были созданы, я приступил к написанию моделей представлений, которые получали данные из базы и привязывали их к представлениям, и самих представлений.

Следующим этапом разработки стало главное окно. Главное окно состоит из «шапки программы», фрейма меню, в котором отображаются все вкладки, и фрейма главной части. При переходе на разные вкладки во фреймы загружаются определенные страницы.

Для основной страницы я создал представления для информации о программе, регистрации и входа в аккаунт. Окно регистрации состоит из трех панелей для ученика, учителя и экзаменатора. Обязательными данными являются логин и пароль, а остальные данные можно будет ввести после регистрации в личном кабинете (приложение № 9). Еще при регистрации для каждого пользователя создается уникальный идентификатор. Регистрация аккаунтов происходит с помощью команд. Класс команд RelayCommand также создается для реализации MVVM. При условии того, что логин и пароль введены и логин уникален, в соответствующую таблицу через объект контекста базы данных добавляется новый пользователь. Окно входа в аккаунт состоит из полей для логина и пароля. При нажатии кнопки входа выполняется команда, в которой логин и пароль проверяются, и при успехе пользователь отправляется

на следующий страницу, где используются модели представлений отдельно для учеников, учителей и экзаменаторов.

Далее я занялся формами для учеников. Сначала я создал модель представления, куда вписывал все команды и функции. Затем я сделал представления для просмотра, изменения данных о ученике и команды для этих действий (приложение № 10). Следующей частью стала форма решения задач. Ученик может выбрать нужный ему класс, предмет и тему для закрепления знаний и подготовки к тестам (приложение № 11). Для вывода вопросов я создал класс QuestionView, который объединяет информацию из нескольких таблиц. Он имеет методы для показа дополнительных материалов и выведения правильного ответа на вопрос. Дальше я приступил к созданию формы тестов. В этом представлении есть данные об уже завершенных тестах, идущих прямо сейчас и еще предстоящих. Если работа доступна, ученик может пройти её. При прохождении теста человек попадает на страницу с заданиями (приложение № 12). После решения всех задач тест отправляется на проверку учителю.

После разработки интерфейса учеников я приступил к созданию форм для учителей. Начал я также с модели представления и страниц для просмотра и изменения данных (приложения № 13). Затем я написал форму для создания классов. Обязательными данными для создания группы являются поля названия группы и номера класса. Добавлять, просматривать и удалять учеников можно на основной странице, выбрав нужный вам класс. Далее я занялся страницей создания тестов. Для начала учитель устанавливает время начала и окончания работы, её длительность и из списка своих классов выбирает нужный. Если учитель решает, что тест будет сгенерирован компьютером, то он ставит темы работы, её сложность и программа создает тест. В ином случае учитель после нажатия кнопки «Создать тест» отправляется на страницу с вопросами, которая похожа на форму с вопросами у учеников, и выбирает вопросы, после чего работа отправляется ученикам (приложение № 14). Еще одной формой стала проверка тестов. На этой странице учитель может просмотреть задания с

полными ответами и выставить баллы. После проверки результаты тестов отправляются ученикам.

Следующим этапом разработки стала стилизация. Программа выполнена в бело-зелёном цвете. Все стили я создавал в файле приложения «App.xaml». В наборах стилей я указывал какими должны быть шрифты, отступы, цвета и т.д. Все эти параметры я указывал с помощью тегов `style` и `setters`. Созданные стили я добавил ко всем элементам управления.

#### **4. Заключение.**

При работе над проектом я убедился, что изучение языков программирования открывает новые возможности и перспективы для каждого человека. К 2023 году программирование настолько глубоко вошло в быт, что люди перестали замечать, как изменилась жизнь. Сотни тысяч привычных вещей не существовали бы без программирования или были бы гораздо менее удобными в использовании.

Нынешнее программирование многогранно используется в таких важных сферах как строительство, бизнес и экономика, медицина, биология и физика. Большой процент физического труда в промышленности заменен на машинный и роботизированный труд, который управляется посредством программного обеспечения, что обеспечивает существенный прирост скорости, точности операций и эффективности производства. Такое богатство разнообразия применений обеспечивается солидным выбором языков программирования, у каждого из которых есть свои плюсы и минусы.

Освоение механизма программирования на языке C# открывает пользователям возможность использования полученных знаний и навыков при работе с десятками и сотнями других программ. Начав с составления простейших приложений, при желании можно в рамках одного инструментария стать профессионалом, разрабатывающим программы любой сложности.

## **5. Список используемых источников.**

### **Книги:**

1. Большая компьютерная энциклопедия – М.: Эксмо, 2007. – 480 с.
2. Моргунов Е.П. PostgreSQL. Основы языка SQL: учеб. пособие – СПб: БХВ-Петербург, 2022. – 336 с.: ил.
3. Пахомов Б. И. С# для начинающих. – СПб.: БХВ-Петербург, 2014. – 432 с.: ил.
4. Фаронов В.В. Создание приложений с помощью С#. Руководство программиста. – М.: Эксмо, 2008. - 576 с.: ил.

### **Интернет источники:**

5. Введение в базы данных [электронный ресурс]. - [habr.com](https://habr.com)
6. Декларативное программирование [электронный ресурс]. - [tproger.ru](https://tproger.ru)
7. Краткий обзор языка С# [электронный ресурс]. - [learn.microsoft.com](https://learn.microsoft.com)
8. Краткое содержание по MVVM: ключевые моменты и примеры [электронный ресурс]. - [highload.today](https://highload.today)
9. Основы языка запросов SQL [электронный ресурс]. - [htmlacademy.ru](https://htmlacademy.ru)
10. Паттерн MVVM [электронный ресурс]. - [metanit.com](https://metanit.com)
11. Преимущества WPF [электронный ресурс]. - [professorweb.ru](https://professorweb.ru)
12. Что такое база данных [электронный ресурс]. - [www.oracle.com](https://www.oracle.com)
13. Что такое Entity Framework и как его использовать [электронный ресурс]. - [skillbox.ru](https://skillbox.ru)

14. Что такое Entity Framework Core [электронный ресурс]. - metanit.com
15. Что такое MS SQL Server и чем она отличается от других СУБД [электронный ресурс]. - timeweb.com
16. Что такое SQL [электронный ресурс]. - aws.amazon.com
17. SQL [электронный ресурс]. - blog.skillfactory.ru
18. Transact-SQL [электронный ресурс]. - ru.wikipedia.org

Приложение № 1

## Классификация и обзор языков программирования





## Правила языка C#.

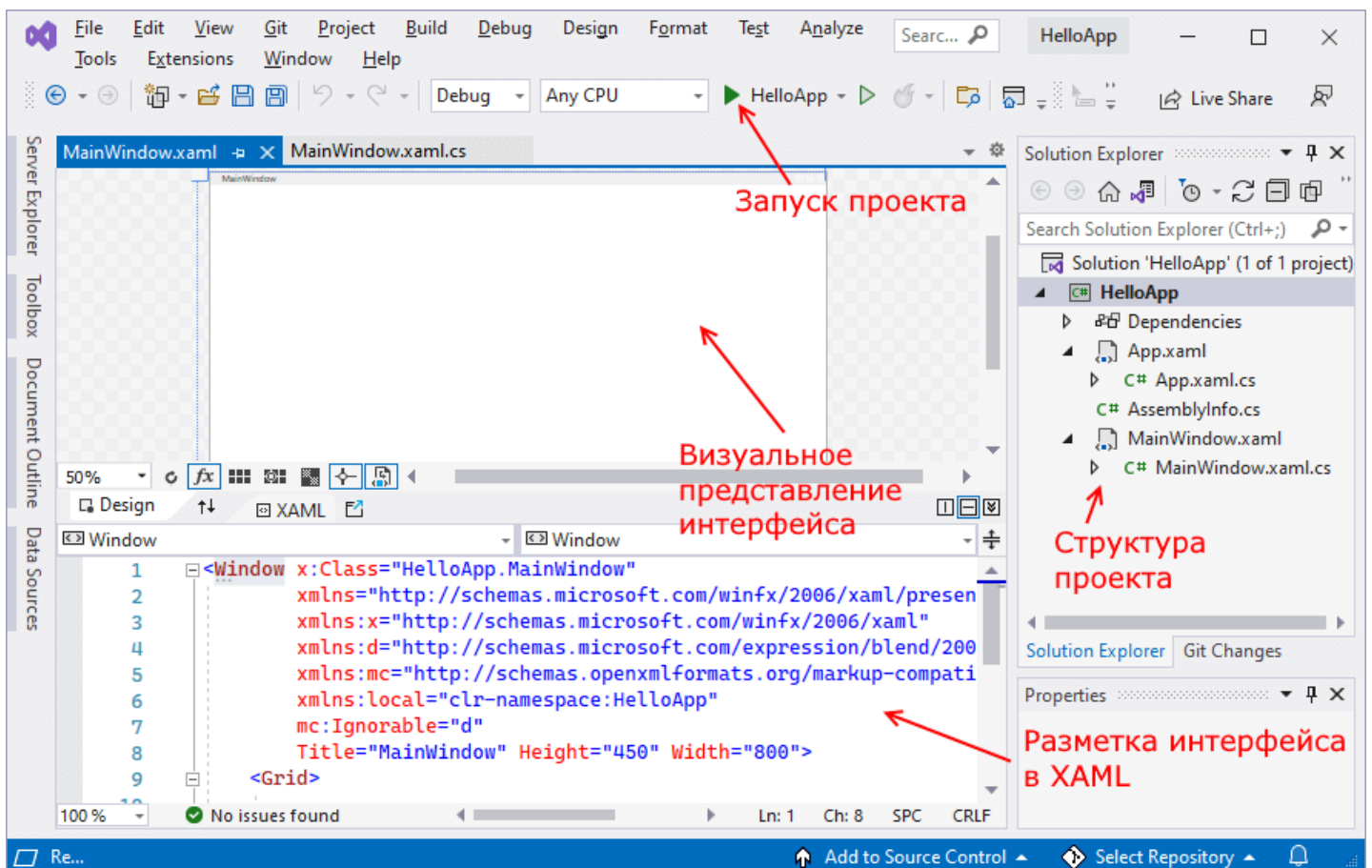
Язык C# - ЯП со строгим контролем типов данных.

В C# две основные категории встроенных типов данных - простые типы и ссылочные типы.

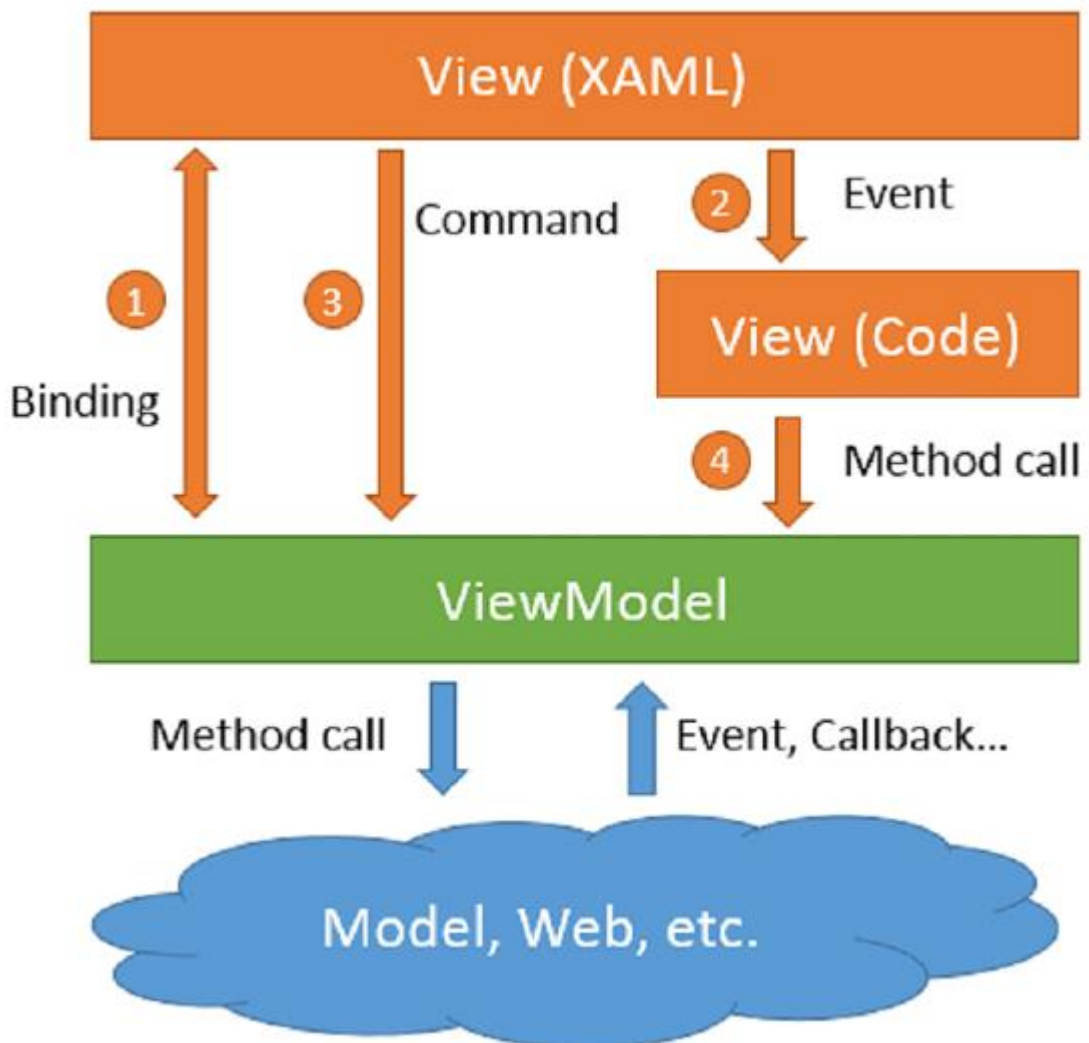
Область видимости переменной в C# - блок кода, заключенный в фигурные скобки { }.

Создание переменных осуществляется при входе управления программой в область видимости; уничтожение - при выходе из нее.

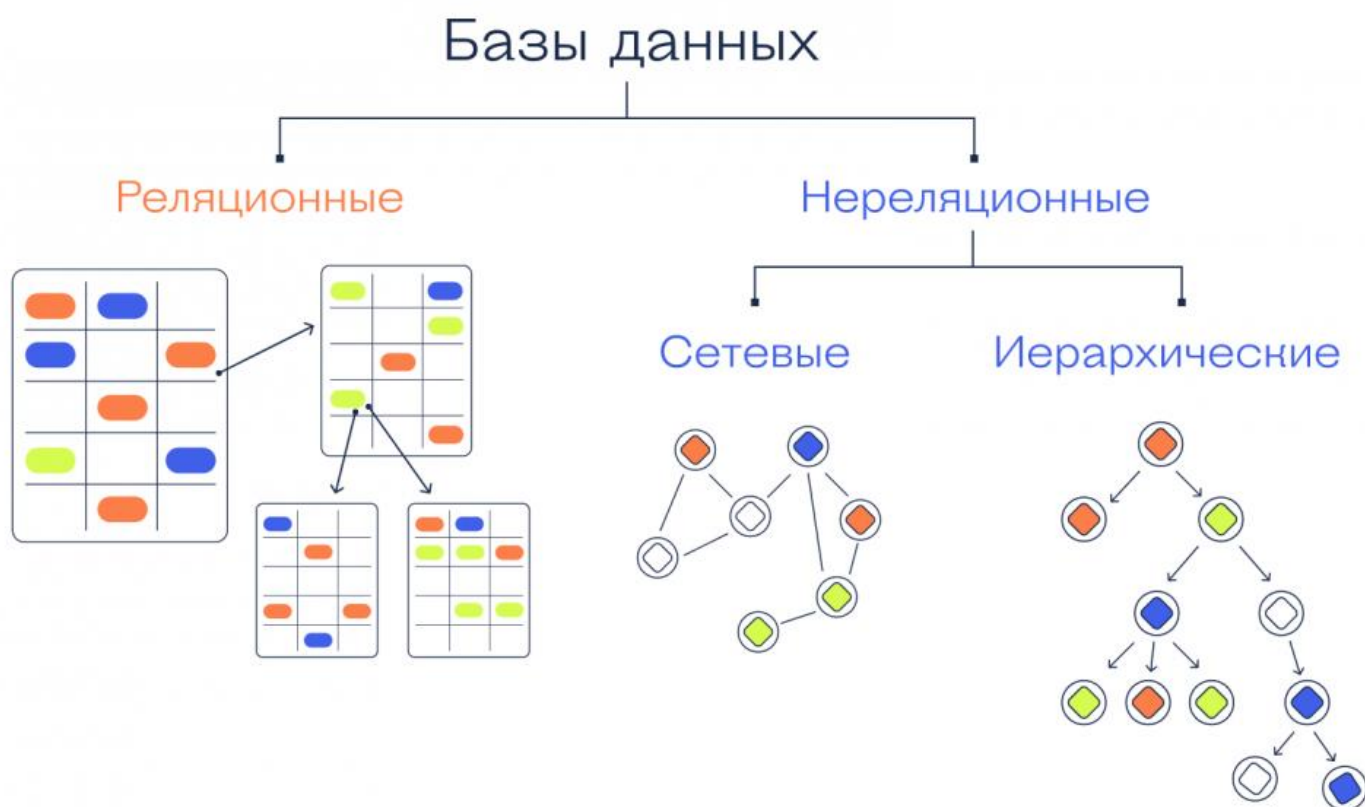
## Среда разработки WPF.



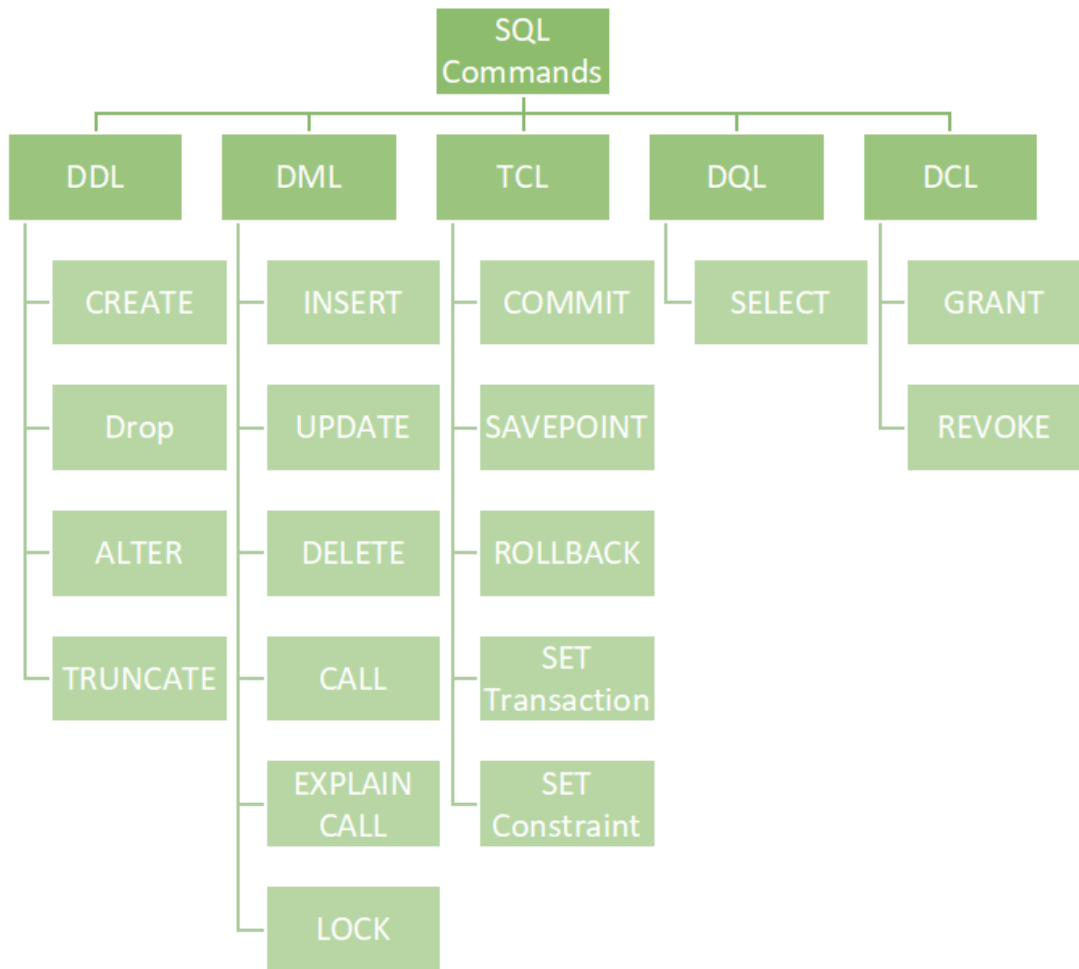
# MVVM.



## Типы базы данных.



## **Типы SQL команд.**



Приложение № 7

# Sql Server Management Studio.

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the 'Northwind' database structure, including tables like 'Products' and 'Suppliers'. A vertical red label 'Структура базы данных' (Database structure) is positioned next to it. The main window shows a SQL query in the 'SQLQuery3.sql' file:

```
/****** Script for SelectTopNRows command from SSMS ******/
SELECT TOP 1000 [ProductID]
      ,[ProductName]
      ,[SupplierID]
      ,[CategoryID]
      ,[QuantityPerUnit]
      ,[UnitPrice]
      ,[UnitsInStock]
      ,[UnitsOnOrder]
      ,[ReorderLevel]
      ,[Discontinued]
FROM [Northwind].[dbo].[Products]
```

Red arrows point from the text 'Код SQL' (SQL code) to the query text and from 'Результаты запроса' (Query results) to the results grid below. The results grid shows the following data:

	ProductID	ProductName	SupplierID	CategoryID	QuantityP
1	1	Chai	1	1	10 boxes :
2	2	Chang	1	1	24 - 12 oz
3	3	Aniseed Syrup	1	2	12 - 550 m
4	4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz j
5	5	Chef Anton's Gumbo Mix	2	2	36 boxes

The status bar at the bottom indicates 'Query executed successfully.' and shows '77 rows'.

Приложение № 8

# Entity Framework Core.



### EDM



Приложение № 9



# Путь к знаниям

О приложении

Регистрация

Вход в аккаунт

## Зарегистрировать аккаунт

Ученик  Учитель  Проверяющий

### Регистрация ученика

Фамилия

Юрченко

Имя

Артём

Отчество

Юрьевич

Дата рождения

15.06.2006

Класс

10

Логин

Asteruss

Пароль

\*\*\*\*\*

Зарегистрироваться

Приложение № 10

# Путь к знаниям

Юрченко Артем Юрьевич

Выйти

Личный кабинет

Задачи

Тесты

## Мой профиль

ID профиля st-08-48-532038-15  
Логин Asteruss  
Фамилия Юрченко  
Имя Артем  
Отчество Юрьевич  
Дата рождения    
Класс 10

Редактировать данные

## Мои группы

Название группы	Учитель
10 класс	Иванова Ольга Петровна

## Мои учителя

Логин	Имя	Фамилия	Отчество
Olga55	Иванова	Ольга	Петровна

Приложение № 11

# Путь к знаниям

Юрченко Артем Юрьевич

Выйти

Личный кабинет

Задачи

Тесты

## Решение задач

Предмет: Математика  
Тип задач: Задача с параметром  
Темы:

Найдите все значения параметра  $k$ , при каждом из которых уравнение  $\frac{1 + (2 - 2k) \sin t}{\cos t - \sin t} = 2k$  имеет хотя бы одно решение на интервале  $(0, \frac{\pi}{2})$ .

Показать правильный ответ

Найдите все значения  $k$ , при каждом из которых уравнение

$$\frac{6k - (2 - 3k) \cos t}{\sin t - \cos t} = 2$$

имеет хотя бы одно решение на отрезке  $[0, \frac{\pi}{2}]$ .

Показать правильный ответ

Определите, при каких значениях параметра  $a$  уравнение

$$|x - 2| = a \log_2 |x - 2|$$

имеет ровно два решения

# Путь к знаниям

Юрченко Артем Юрьевич

Выйти

Личный кабинет

Задачи

Тесты

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Дана правильная треугольная призма  $ABC A_1 B_1 C_1$ , все ребра основания которой равны 2. Сечение, проходящее через боковое ребро  $AA_1$  и середину  $M$  ребра  $B_1 C_1$ , является квадратом.

- Докажите, что расстояние между прямыми  $A_1 B$  и  $AM$  равно длине перпендикуляра, опущенного из центра этого квадрата на прямую  $A_1 B$ .
- Найдите это расстояние.

Ответ:

Ответить

Приложение №13

# Путь к знаниям

Иванова Ольга Петровна

Выйти

Личный кабинет

Добавить класс

Тесты

## Мой профиль

ID профиля tc-08-40-809741-40  
Логин Olga55  
Фамилия Иванова  
Имя Ольга  
Отчество Петровна

Редактировать данные

## Мои классы и группы

Название группы	Учитель	Класс
10а класс	Иванова Ольга Петровна	10 класс

Информация о группе    Просмотреть учеников    Добавить ученика    Удалить класс

Приложение №14

# Путь к знаниям

Иванова Ольга Петровна

Выйти

Личный кабинет

Добавить класс

Тесты

## Создание теста

Предмет

Время начала теста

Время окончания теста

Время написание теста

Выберите тип задач

Сложные задачи с параметром  
 Сложные неравенства  
 Сложные практические задачи  
 **Стереометрическая задача**  
 Уравнения, системы уравнений  
 Числа и их свойства

Выберите тему(необязательно)

Круглые тела: цилиндр, конус, шар  
 Объёмы многогранников  
 Расстояние между прямыми и плоскостями  
 Расстояние от точки до прямой и до плоскости  
 Сечения многогранников  
 Угол между плоскостями

Количество вопросов

Создать тест