

**ОТЧЕТ О ВЫПОЛНЕНИИ**

**практической работы №15**

**Тема: «Создание Windows-приложений. Создание графического калькулятора».**

Выполнил: ст.гр. Алирзаева З.Н

Специальность: 09.02.07

Информационные системы и программирование

Проверил: преподаватель Боклач Б. И.

Москва

2023

**Цель работы**:

1. Понять основы разработки графического интерфейса в приложениях Windows Forms.
2. Изучить стандартные элементы управления, обеспечивающие требуемый графический интерфейс пользователя.
3. Научиться создавать графический калькулятор на языке C# с помощью Windows Forms.

**Вариант 1**.

**Задание:** Cоздать графический калькулятор на языке C# с помощью Windows Forms.

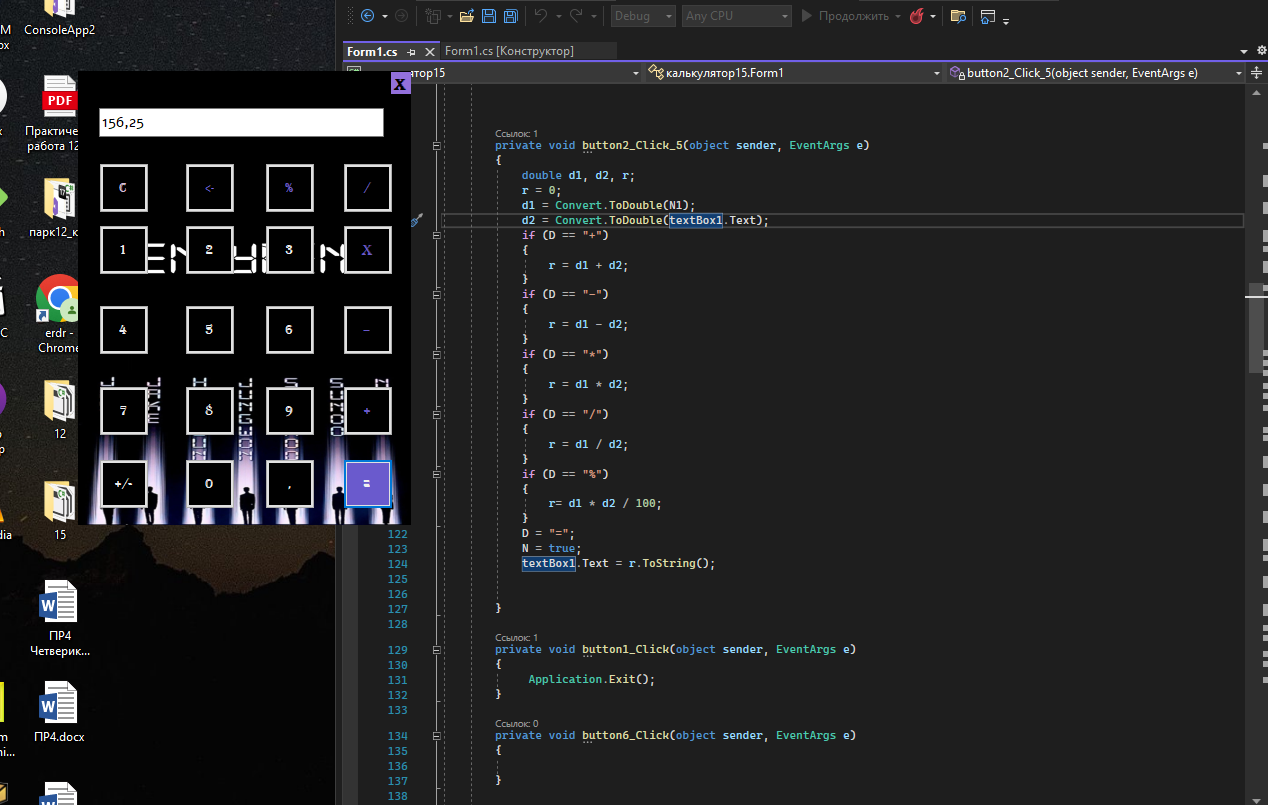


Рис 1.

**Вывод:**

В данной практической работе я познакомилась с основами разработки графического интерфейса в приложениях **Windows Forms** и изучили стандартные элементы управления, обеспечивающие требуемый графический интерфейс пользователя.

**Контрольные вопросы:**

1. **Какое свойство определяет положение компонента относительно клиентской области родителя? #?**

**Anchor —** свойство позволяет определить, к какой стороне родительского компонента прикреплять компонент;

1. **Перечислите этапы разработки Windows-приложения.**

* Создание плана разработки;
* Создания **формы,** редактирование **формы;**
* Программирование кнопок **формы;**
* Тестирование приложения;
* Внесение изменений, оптимизация приложения, в случаи ошибок;

1. **Что такое событийная процедура?**

**Событийная процедура -** подпрограмма, которая начинает выполняться после достижения какого-то события. Событийная процедура всегда связана с каким-то ОБЪЕКТОМ. Чтобы вызвать событийную процедуру ОБЪЕКТА, достаточно по этому ОБЪЕКТУ сделать двойной левый щелчок. Событийные процедуры бывают и локальные, и глобальные.

1. **К какому классу относятся элементы управления?**

Класс **Control**

1. **Какие способы работы со свойствами компонента Вы знаете?**

**BackgroundColor —** цвет фона окна. Это свойство имеет тип класса Color. Если щелкнуть в поле напротив свойства, то появится кнопка выпадающего списка. Щелкните по ней, и вы увидите панель выбора цвета.

**Cursor —** позволяет задать тип курсора, который будет отображаться в момент, когда курсор мыши находится над поверхностью окна. При изменении этого свойства в визуальном дизайнере появляется выпадающий список, в котором перечислены названия типов курсоров и даже показаны их изображения;

**Enabled —** позволяет задать доступность окна. Если свойство равно false, то компонент не станет реагировать на действия пользователя, т. е. будет недоступен;

**Font —** шрифт, который будет использоваться для вывода текста поверх окна. Для окна это, на первый взгляд, не очень актуально.

**ForeColor** — цвет переднего плана, который чаще всего используется в качестве цвета текста. Если вы посмотрите на доступные свойства шрифта, то заметите, что среди них нет цвета. Цвет текста задается через свойство ForeColor;

**FormBorderStyle —** стиль контура формы. Это свойство является перечислением, и в редакторе свойств появляется выпадающий список для выбора из следующих элементов;

**None —** контура не будет;

**FixedSingle —** тоненький фиксированный контур, не позволяющий изменять размеры окна;

**Fixed3D —** контур фиксированный, не позволяющий изменять размеры окна, но трехмерный, создающий эффект заглубленного окна;

**FixedDialog —** похож на FixedSingle, только в заголовке окна не будет значка. Такой контур чаще всего используют для диалоговых окон;

1. **Какой тип данных имеет информация, которая вводится и выводится с помощью компонента «текстовое поле»?**

**string**