

# Программирование на C++



| Минцифры  
РОССИИ

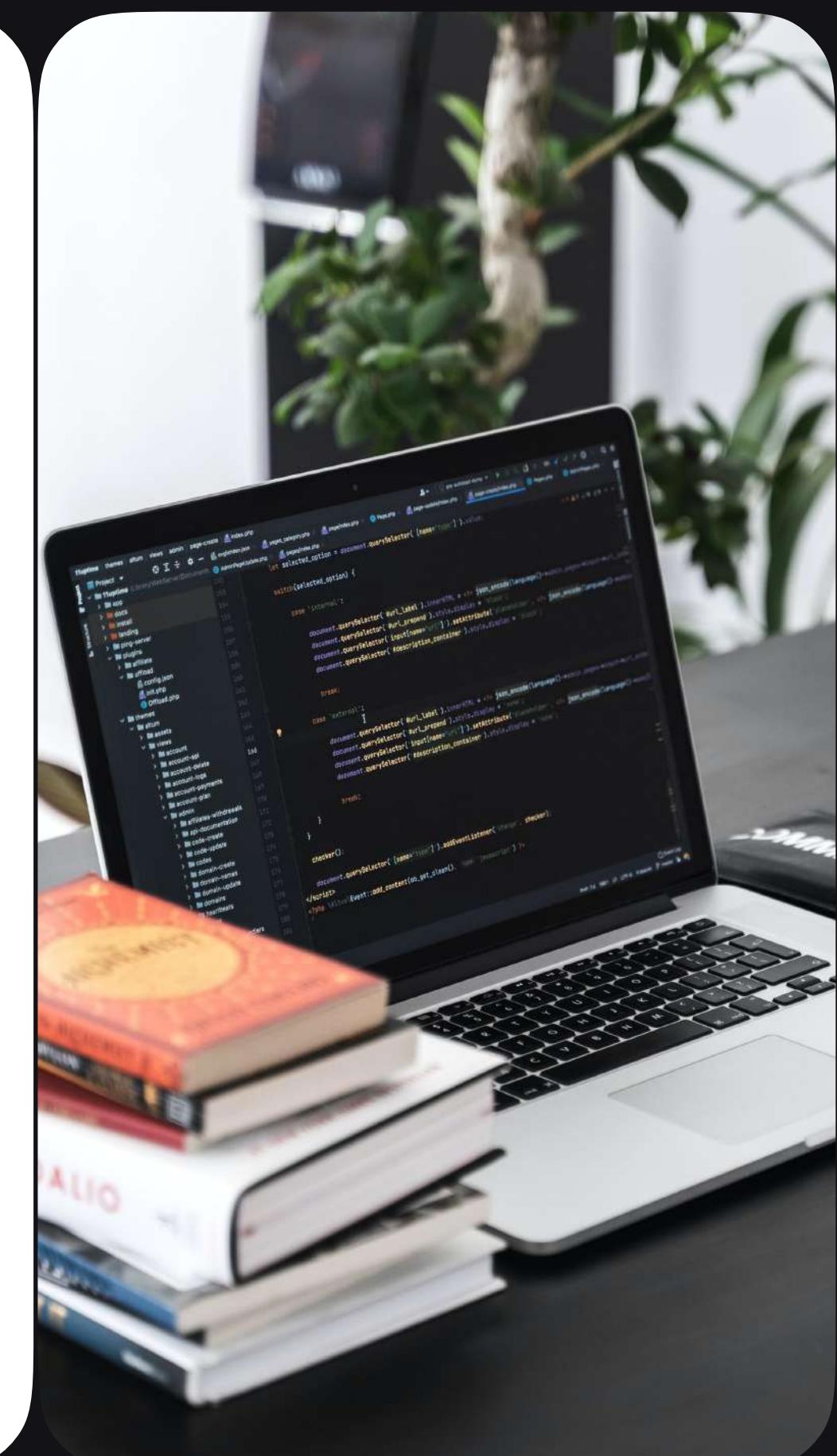
UCHI DOMA

**20.35**  
УНИВЕРСИТЕТ

Урок 12 Модуль 4

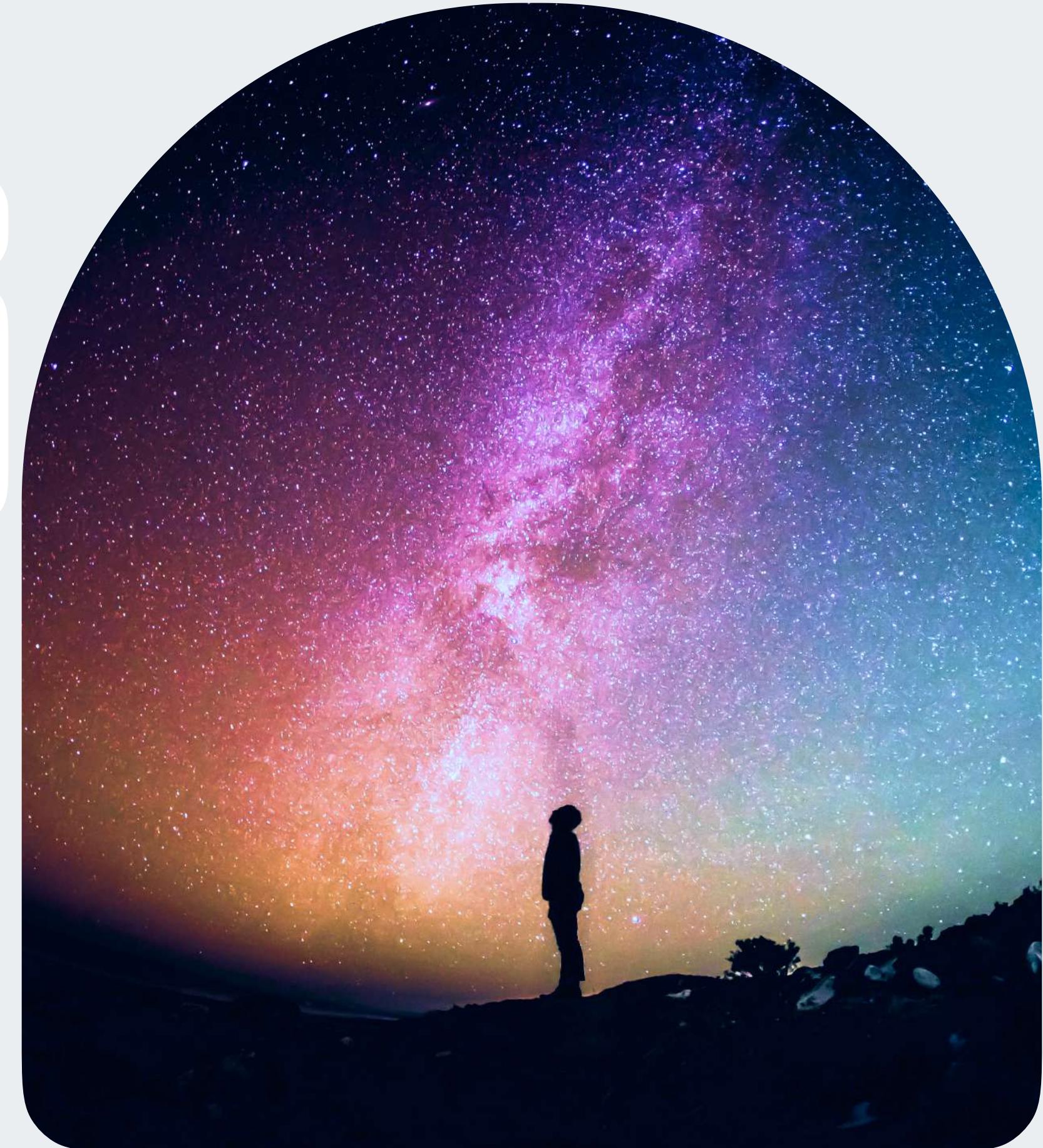
# Проект «Создание браузера»

Полезные материалы



# Цели урока

- ❖ Разработать web-браузер
- ❖ Отработать на практике написание приложений для работы в internet на C++



# Web-браузер



**Браузер, веб-обозреватель или веб-браузер**  
(от англ. *web browser*) – прикладное программное  
обеспечение для просмотра web-страниц.

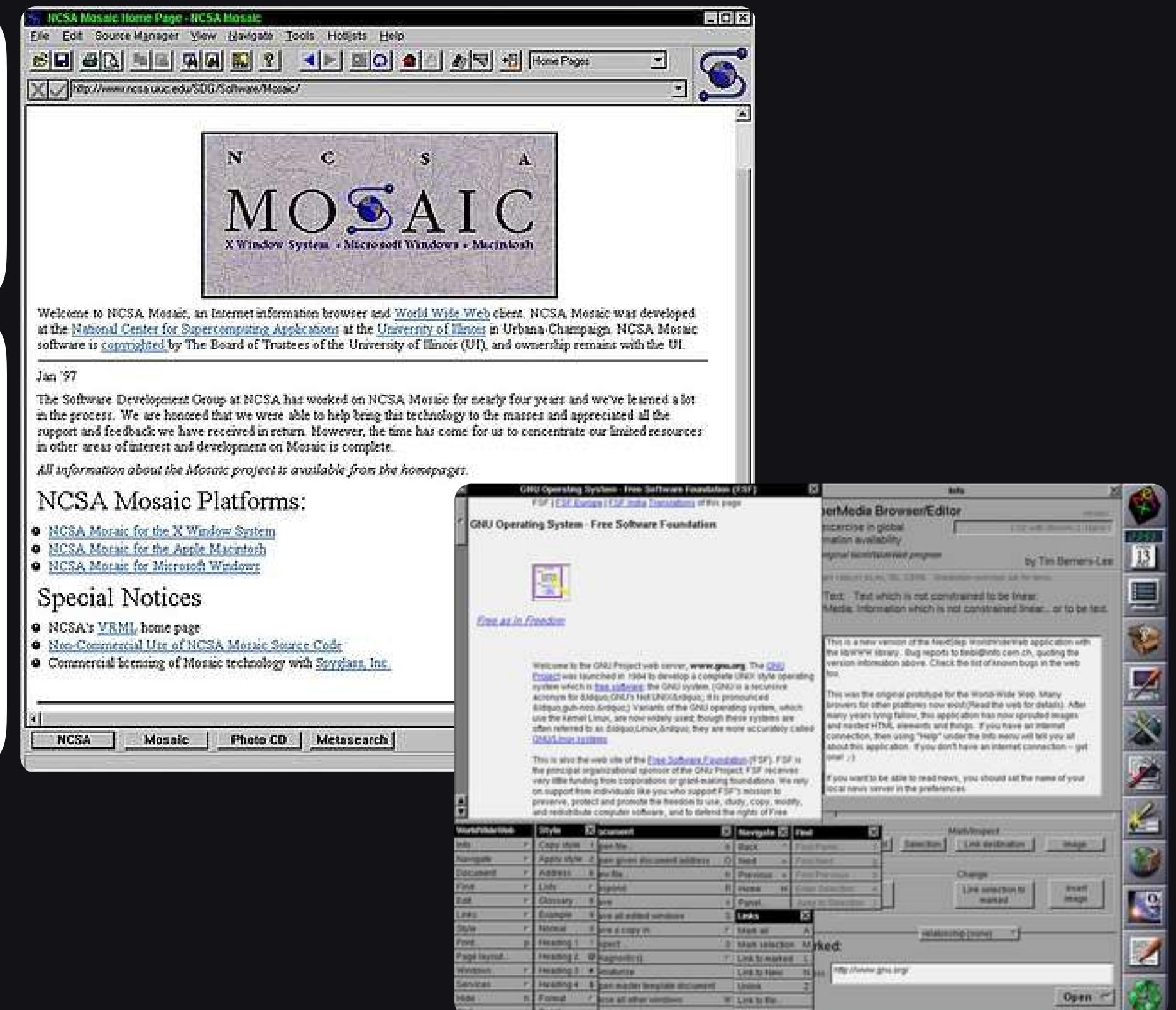
В глобальной сети браузеры используют для запроса, обработки,  
манипулирования и отображения содержания веб-сайтов.

Многие современные браузеры также могут использоваться для  
обмена файлами с серверами FTP, а также для непосредственного  
просмотра содержания файлов многих графических форматов  
(gif, jpeg, png, svg), аудио – и видеоформатов (mp3, mpeg),  
текстовых форматов (pdf, djvu) и других файлов.

# Web-браузер

Первый веб-браузер был создан в 1990 году сэром Тимом Бернерсом-Ли. Он назывался WorldWideWeb и позже был переименован в Nexus.

Первым распространённым браузером с графическим интерфейсом был NCSA Mosaic. Исходный код этого одного из первых браузеров был открыт, и некоторые другие браузеры (Netscape Navigator и Internet Explorer) взяли его за основу.



# Web-браузер

Борьба производителей браузеров за доминирование получила название “Война Браузеров”.

Первая Война Браузеров происходила в период с 1995 по 2001 год и сводилась к борьбе за долю рынка между **Netscape Navigator** и **Internet Explorer**.

На начало 1995 года Netscape Navigator обладал абсолютным доминированием с долей рынка порядка 80 %. Доля Internet Explorer составляла менее 5% .



VS



# Web-браузер



Война закончилась убедительной победой Internet Explorer, который к концу 2001 года контролировал около 96 % рынка.

Успех, по крайней мере отчасти, объясняется политикой компании Microsoft, которая предустановливала Internet Explorer в качестве бесплатного приложения на компьютеры в составе Windows.c



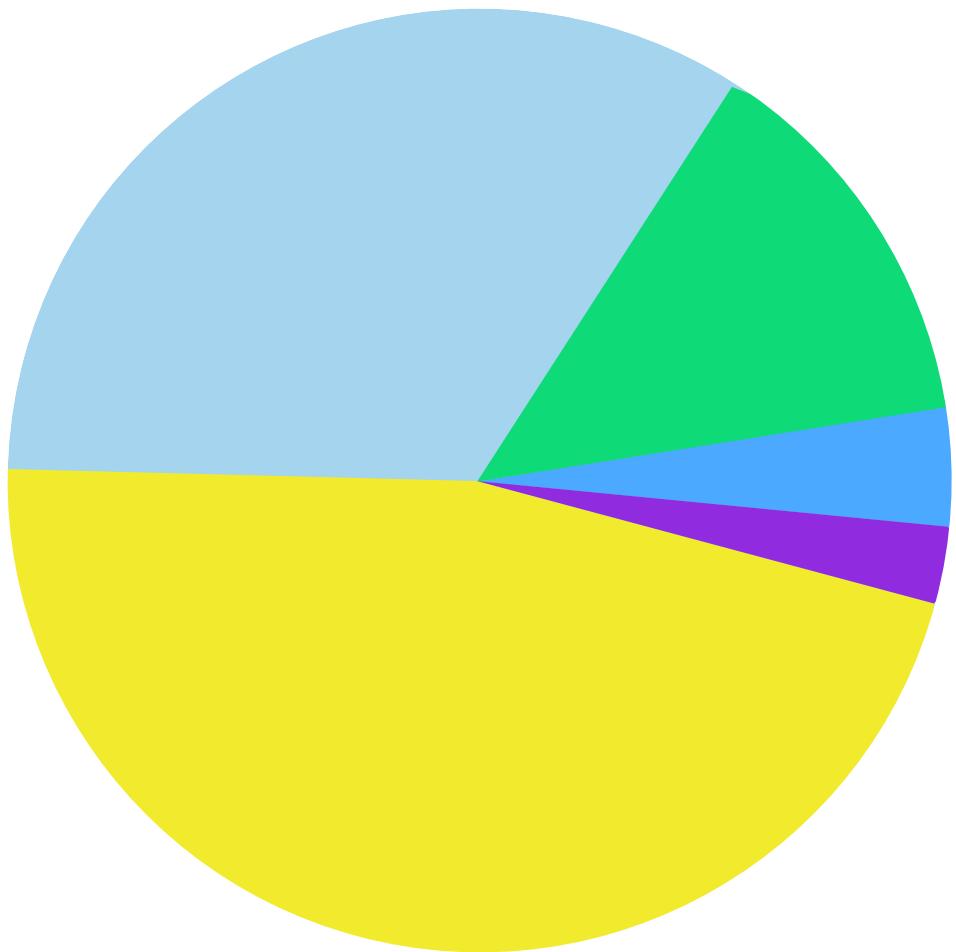
# Web-браузер

Вторая Война браузеров началась в 2004 году и начиналась как борьба, в первую очередь **Opera** и **Mozilla Foundation** с монополией IE.

Она привела к практически полному вытеснению Internet Explorer и доминированию Google Chrome, доля которого на конец 2017 года составляла около 60 %.



# Web-браузер



- Google Chrome (42.1 %)
- Яндекс.Браузер (30.54 %)
- Safari (12.23 %)
- Opera (3.66 %)
- Android Browser (2.41 %)

Статистика использования  
браузеров в зоне RU  
на апрель 2023 года.

Практика



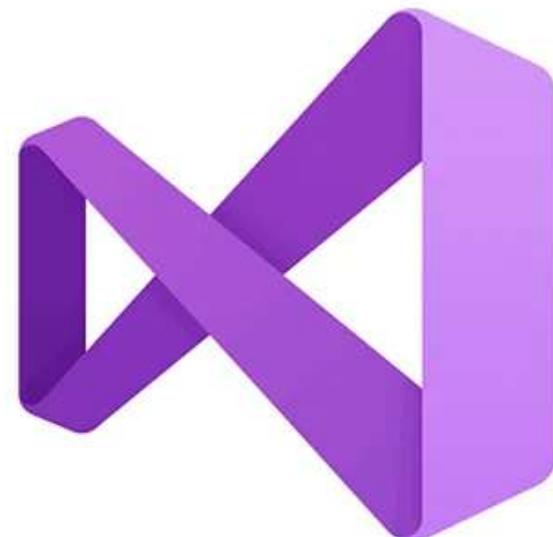
Создадим свой Web-браузер  
с помощью WinForms.

Практика



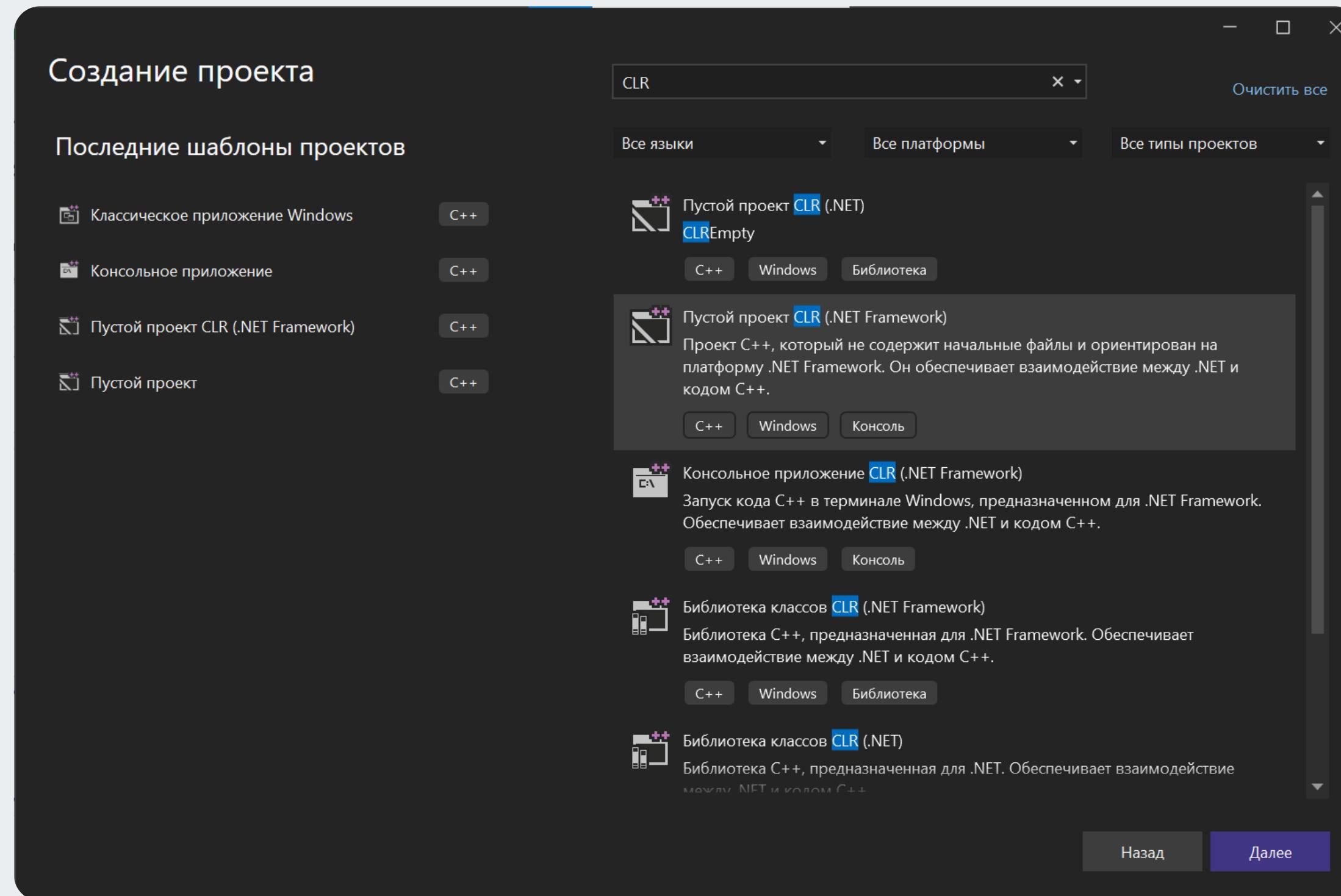
Будем работать в MS Visual Studio.

Запустите Visual Studio.

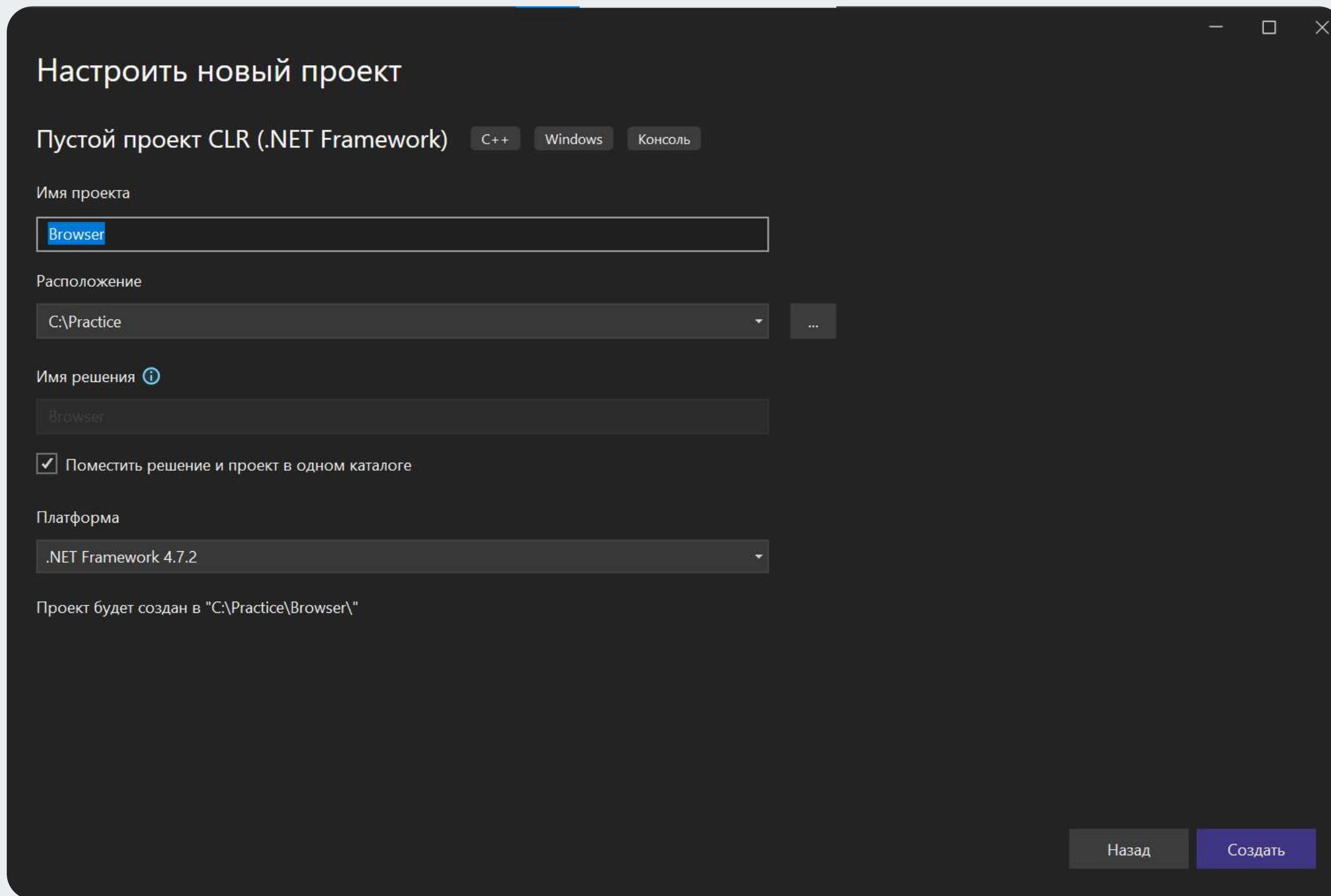


Visual  
Studio

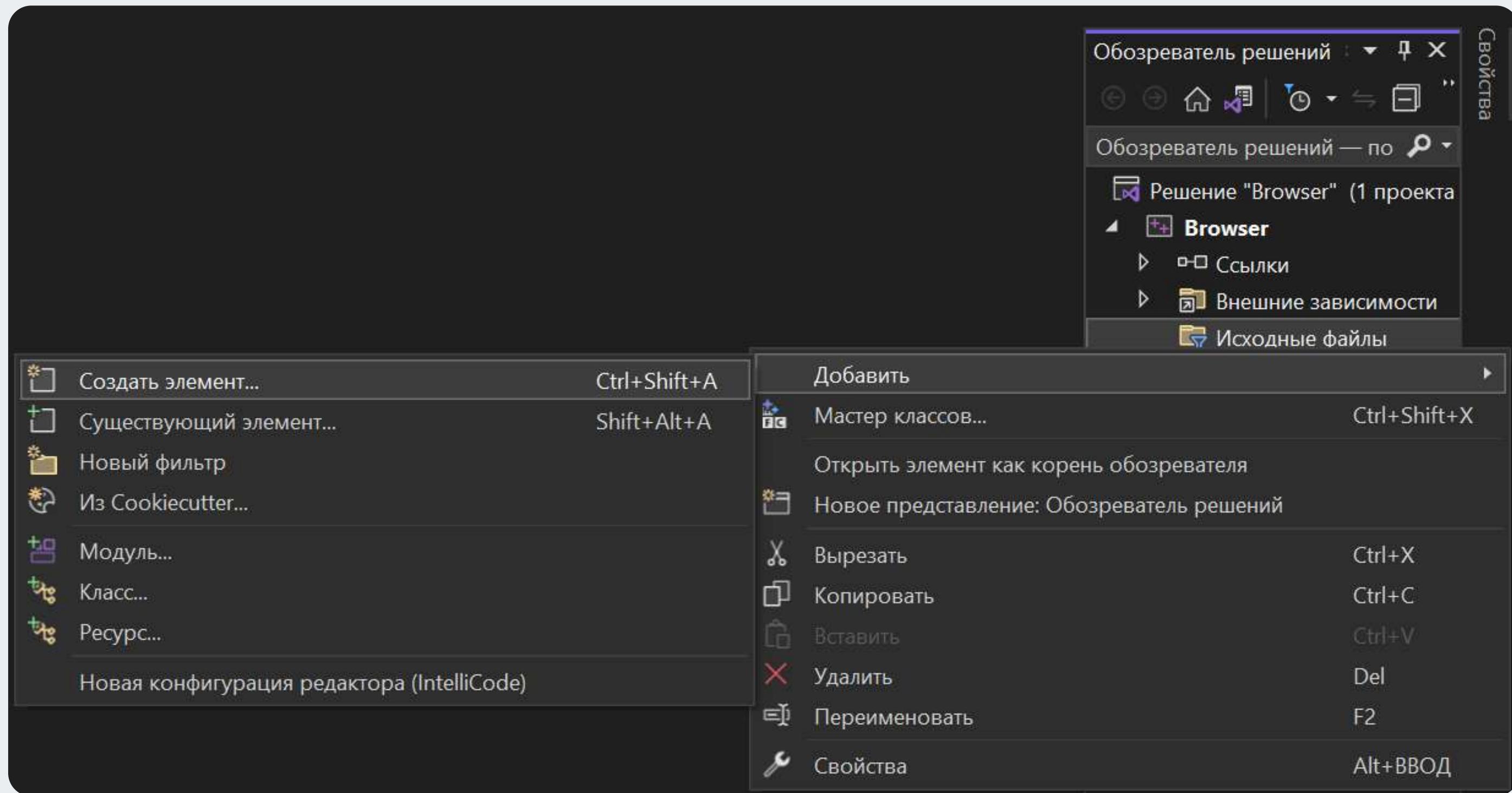
# Создайте новый проект



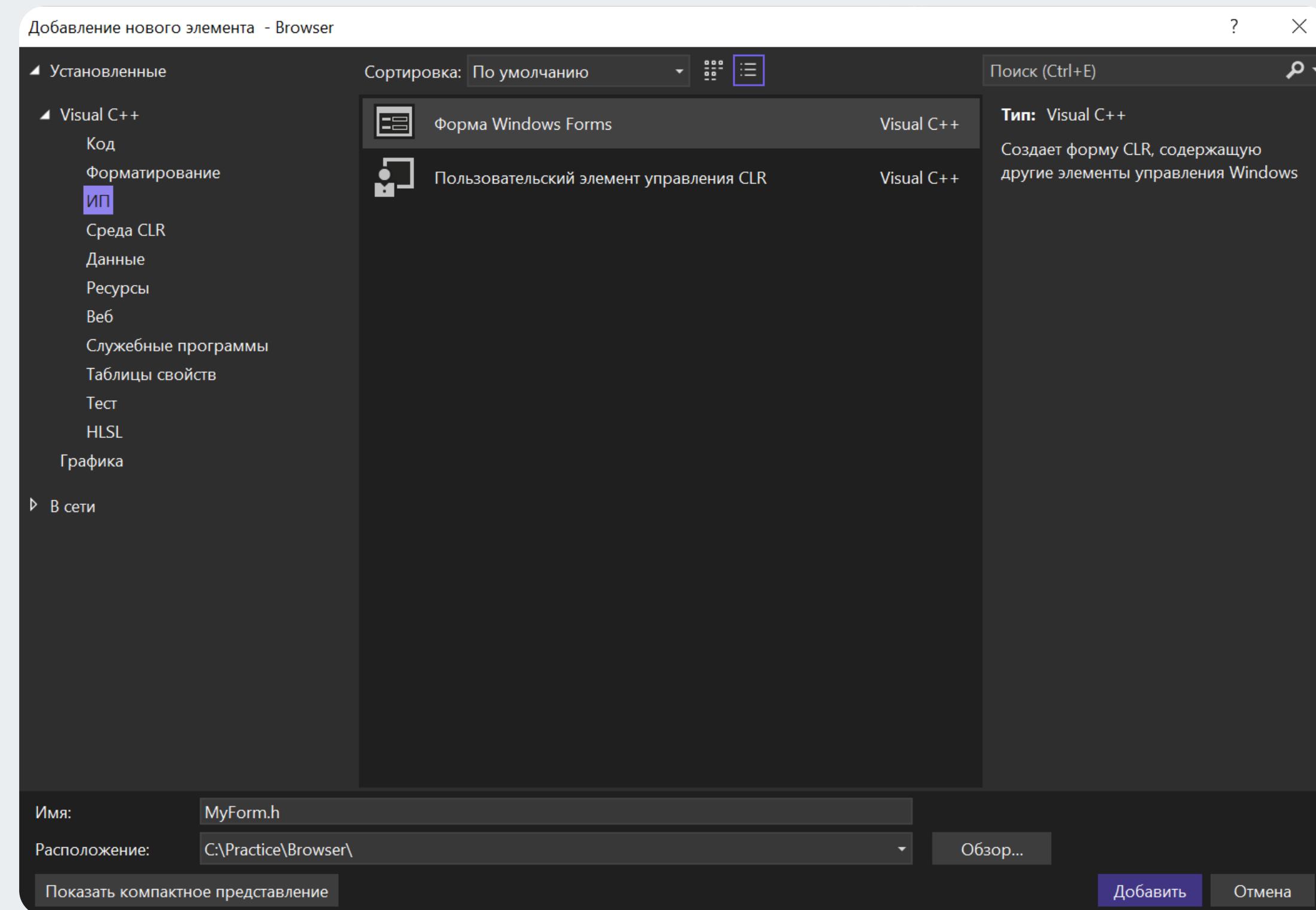
# Создайте новый проект



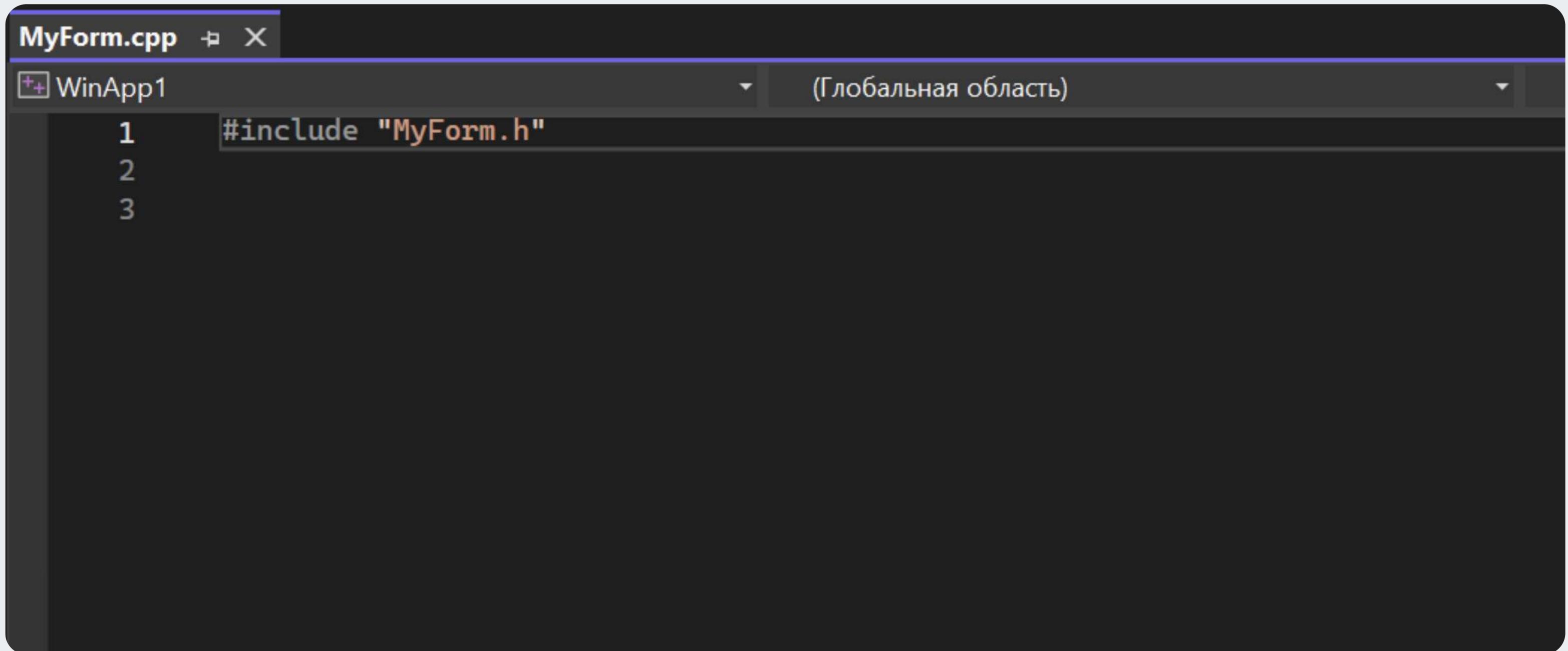
# Добавьте заголовочный файл



# Добавьте заголовочный файл



# Откроем файл MyForm.cpp



The screenshot shows a code editor window titled "MyForm.cpp". The window has a dark theme with a purple header bar. In the header, there is a close button (X) and a dropdown menu icon. Below the title, there is a tab labeled "WinApp1" with a plus sign icon. To the right of the tab, there is a dropdown menu set to "Глобальная область". The main editor area contains three lines of code:

```
1 #include "MyForm.h"  
2  
3
```



Мы видим, что подключен заголовочный файл MyForm.h, но нет функции main, поэтому проект не запустится.

# Добавляем пространства имен

```
1 #include "MyForm.h"  
2  
3 using namespace System;  
4 using namespace System::Windows::Forms;
```

# Добавим атрибут, чтобы функция main запускалась в отдельном потоке данных

```
1 #include "MyForm.h"  
2  
3 using namespace System;  
4 using namespace System::Windows::Forms;  
5  
6 [STAThreadAttribute]
```

# Объявим функцию main

```
1 #include "MyForm.h"  
2  
3 using namespace System;  
4 using namespace System::Windows::Forms;  
5  
6 [STAThreadAttribute]  
7  
8 int main(array<String^>^ args)  
9 {  
10 }  
11 }
```

массив в виде  
контейнера array

строковый тип из WinForms  
(обладает большим функционалом,  
чем обычный std::string)

динамическая очистка памяти после  
окончания использования объекта

# Функция main

```
int main(array<String^>^ args)
{
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
}
```

Подключаем визуальные стили

Необходимо для корректной обработки текста

# Создаем форму

```
int main(array<String^>^ args)
{
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
    Browser::MyForm form; ←
    Application::Run(% form);
}
```

Запускаем приложение и передаем в качестве параметра по ссылке нашу форму (по ссылке так как наш класс MyForm объявлен в файле MyForm.h как ссылочный класс)

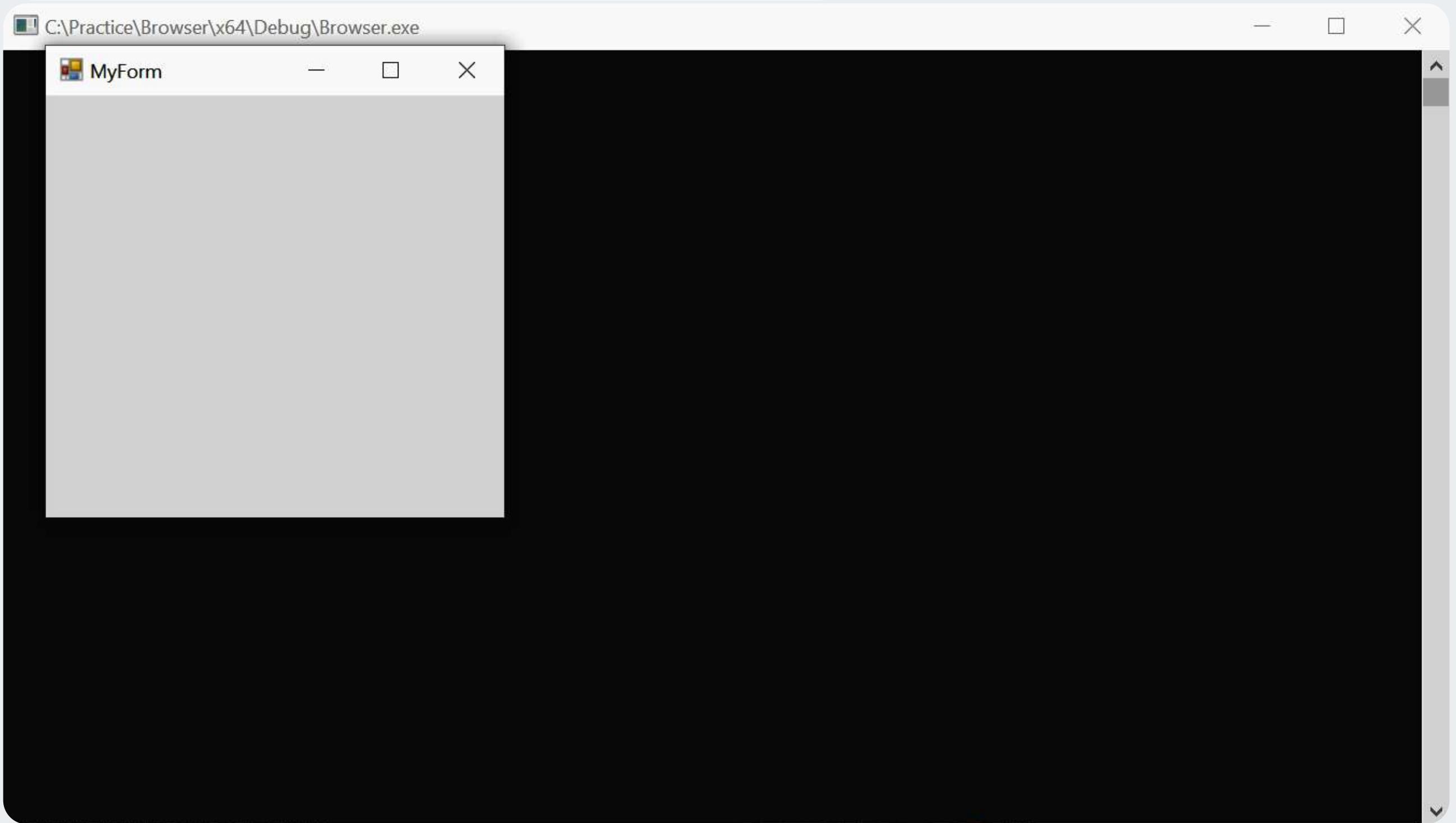
В пространстве имён Browser создаём объект с именем form класса MyForm (пространство имён и класс описаны в нашем файле MyForm.h)

# Файл MyForm.cpp

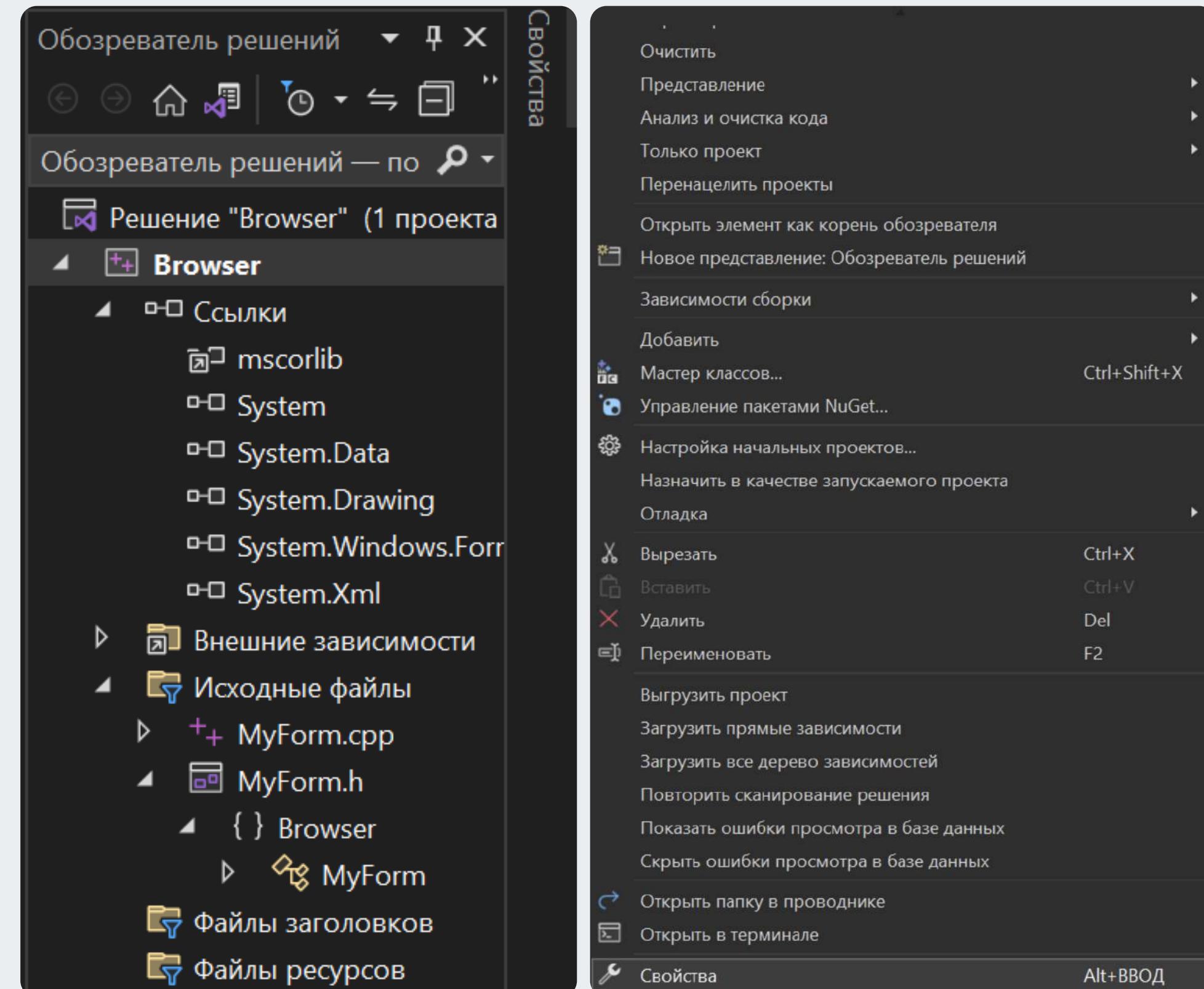
```
1 #include "MyForm.h"
2
3 using namespace System;
4 using namespace System::Windows::Forms;
5
6 [STAThreadAttribute]
7
8 int main(array<String^>^ args)
9 {
10     Application::SetCompatibleTextRenderingDefault(false);
11     Application::EnableVisualStyles();
12     Browser::MyForm form;
13     Application::Run(% form);
14
15 }
```

# Приложение готово – запускаем

Вместе с формой показывается консоль.

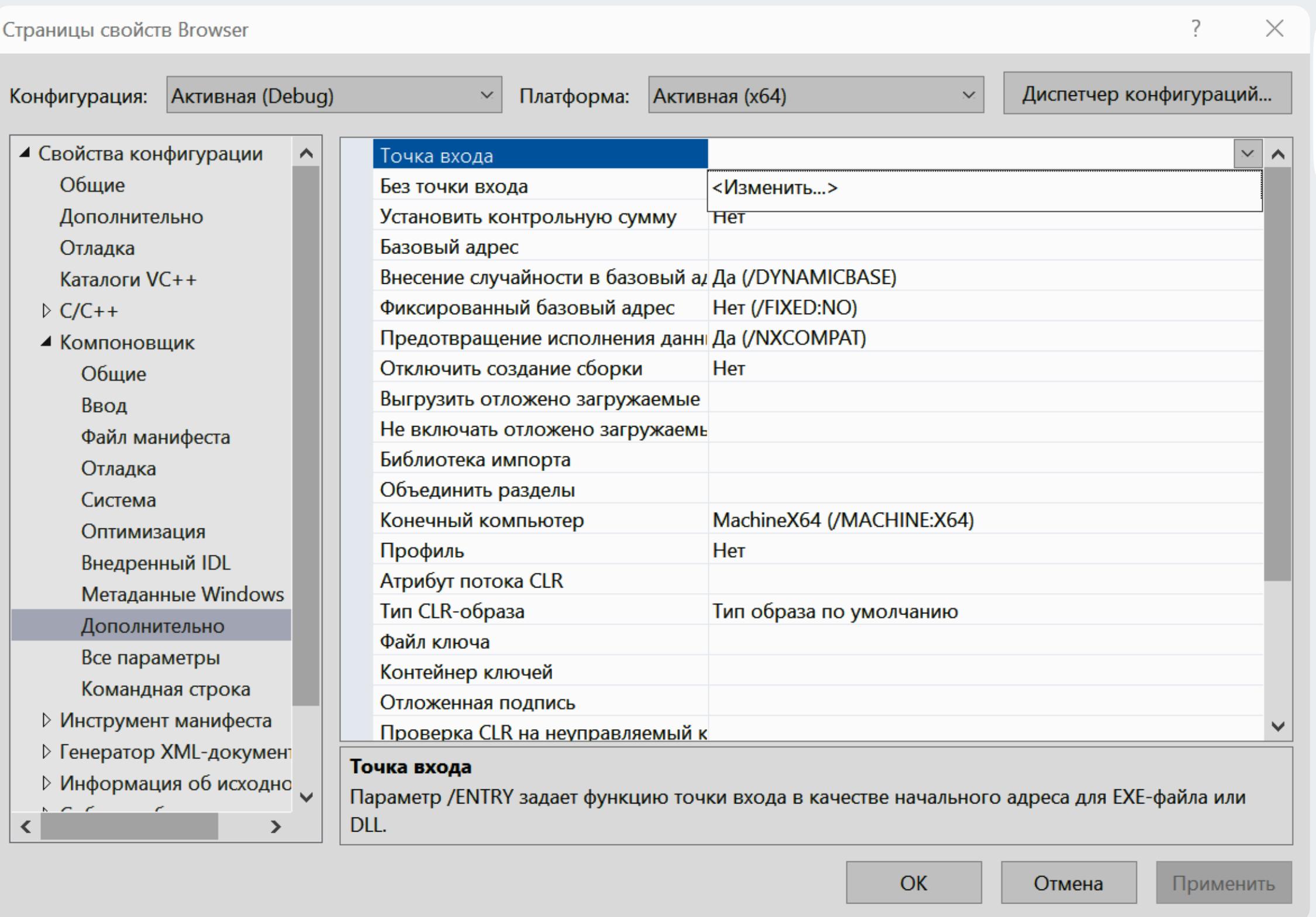


# Настроим проект



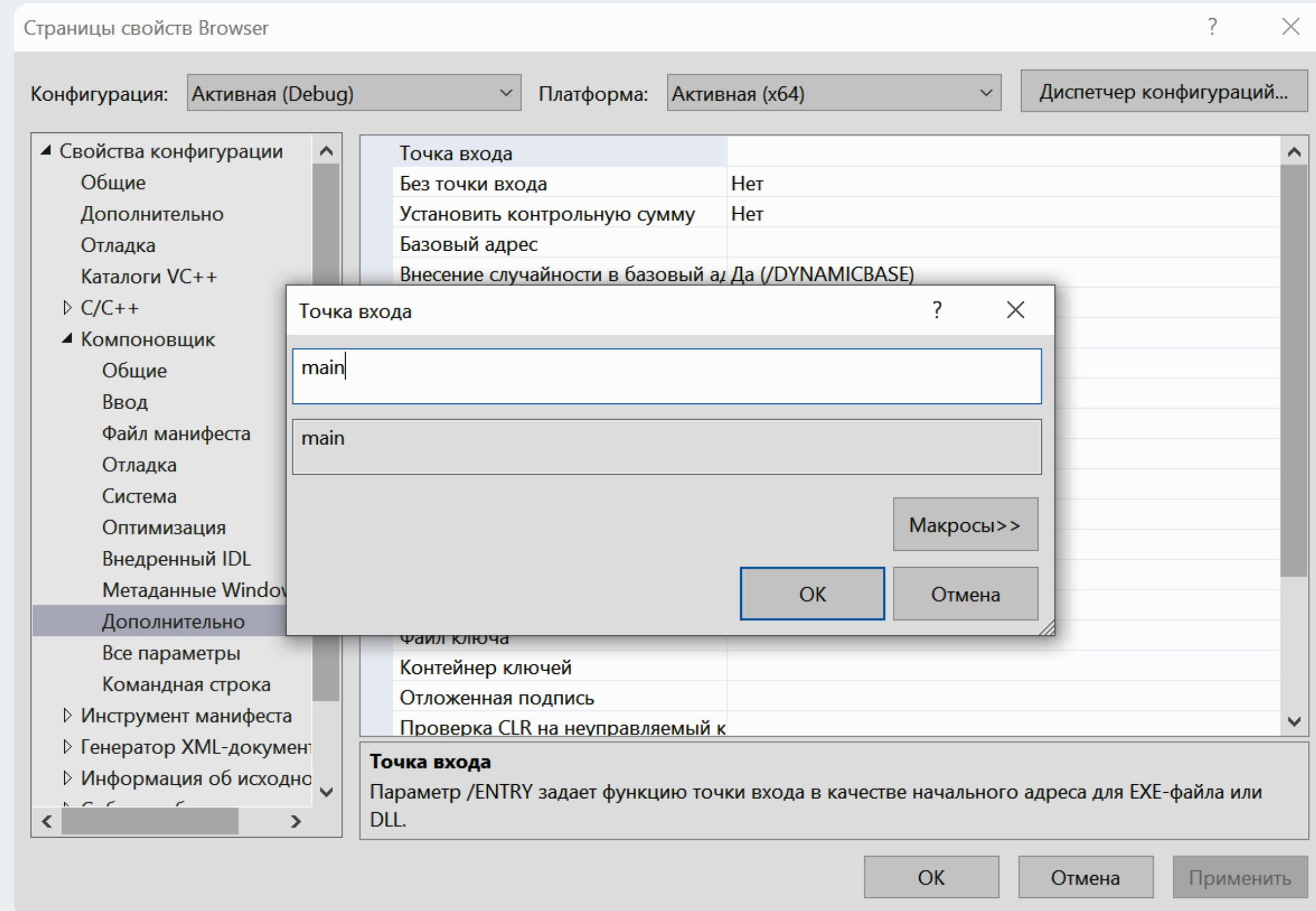
# Настройка проекта

Страницы свойств Browser



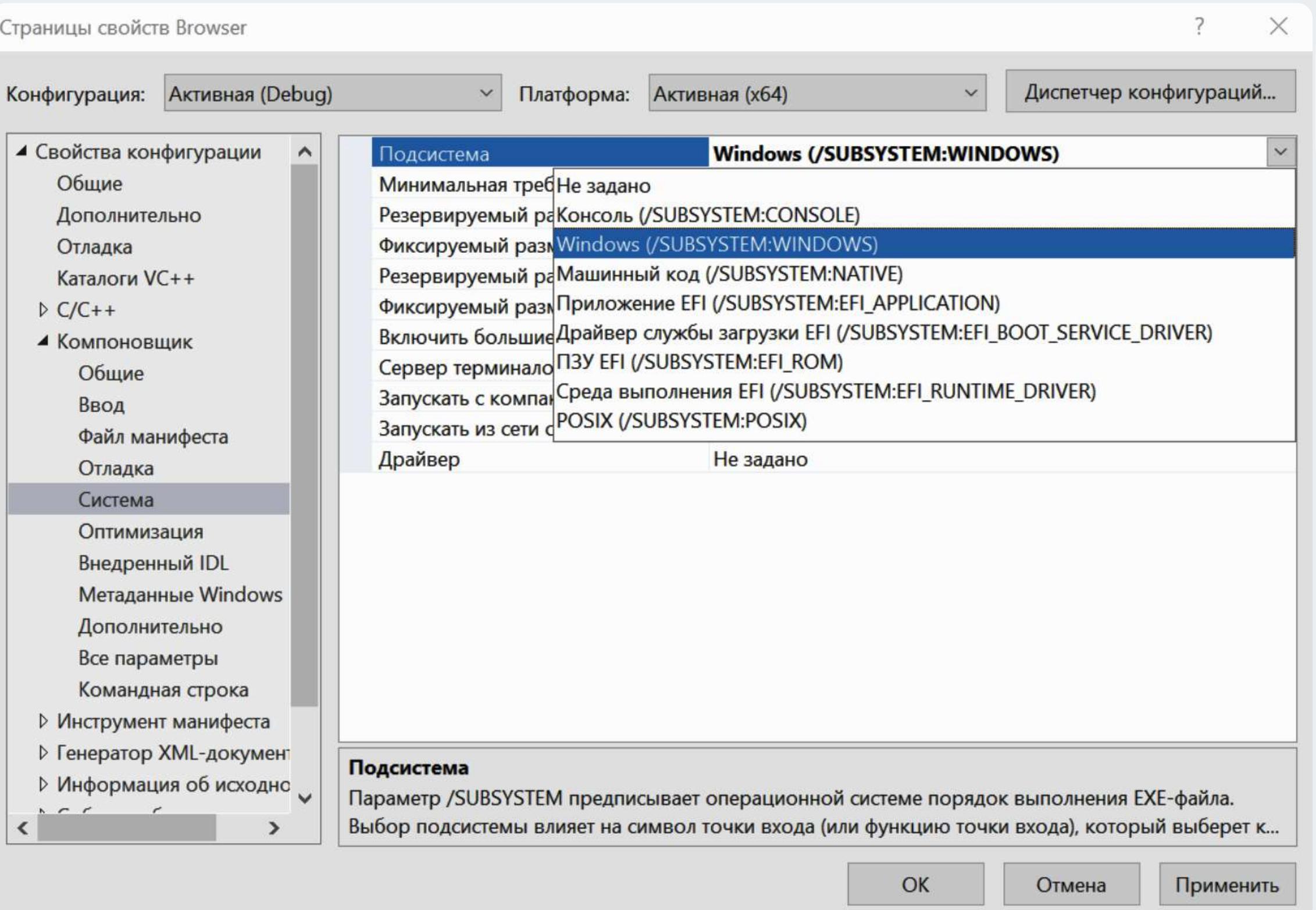
Свойства → Компоновщик →  
Дополнительно → Точка входа →  
<Изменить...>

# Настройка проекта



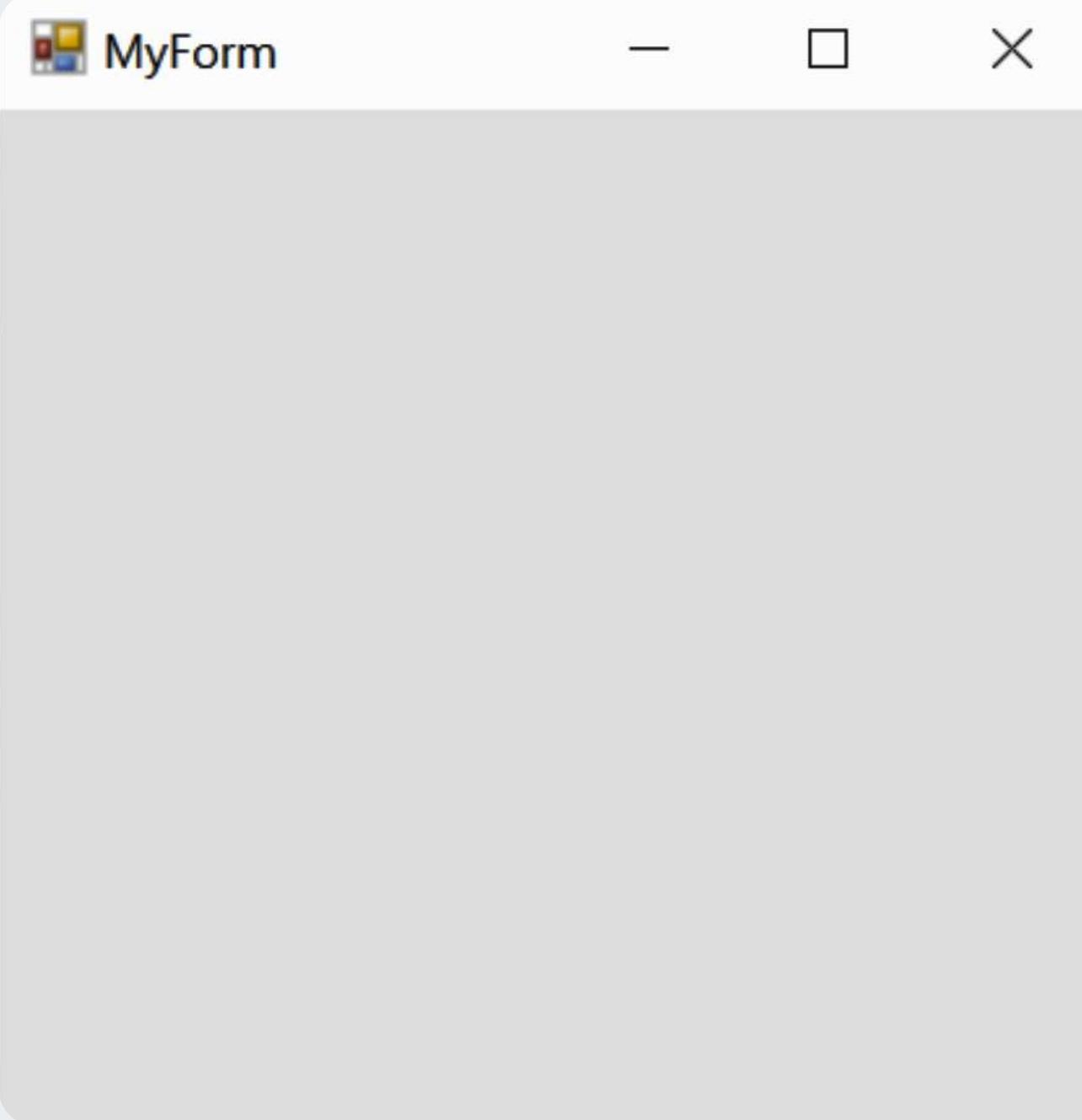
# Настройка проекта

Страницы свойств Browser



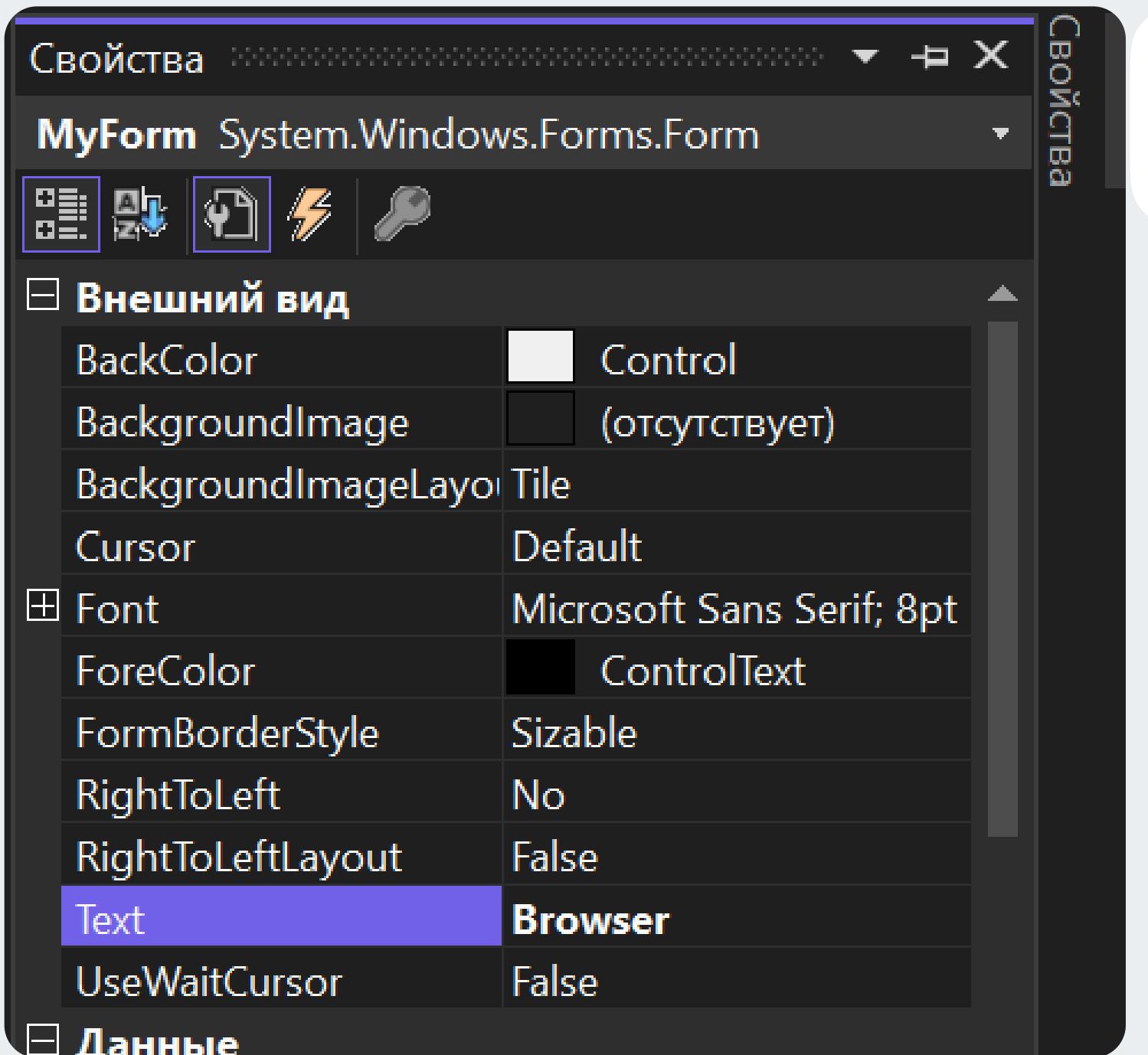
Свойства → Компоновщик →  
Система → Подсистема →  
Windows

# Запустите проект



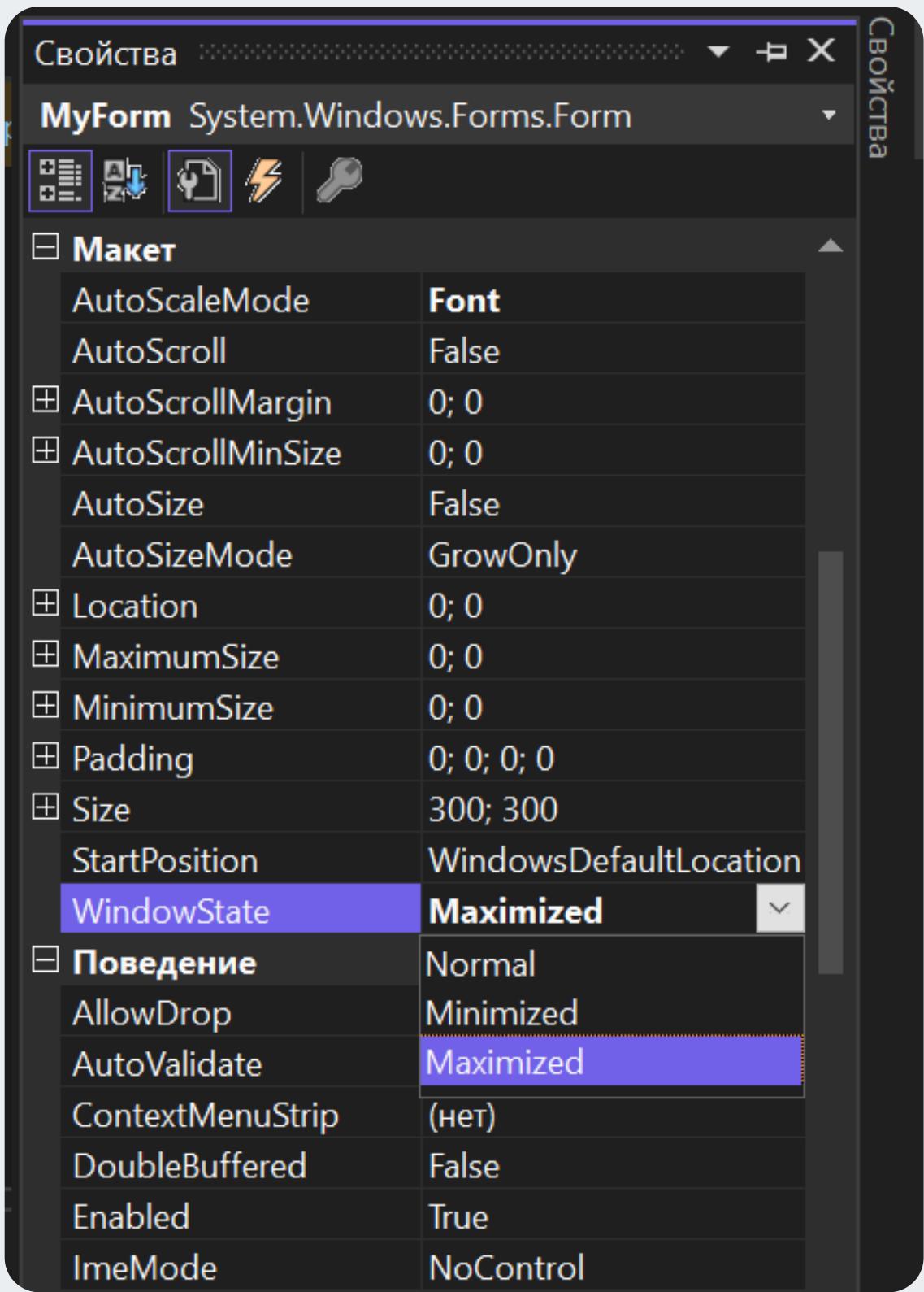
Теперь открывается только  
графический интерфейс без консоли

# Настройка формы



Поменяйте заголовок окна  
(свойство Text)

# Настройка формы

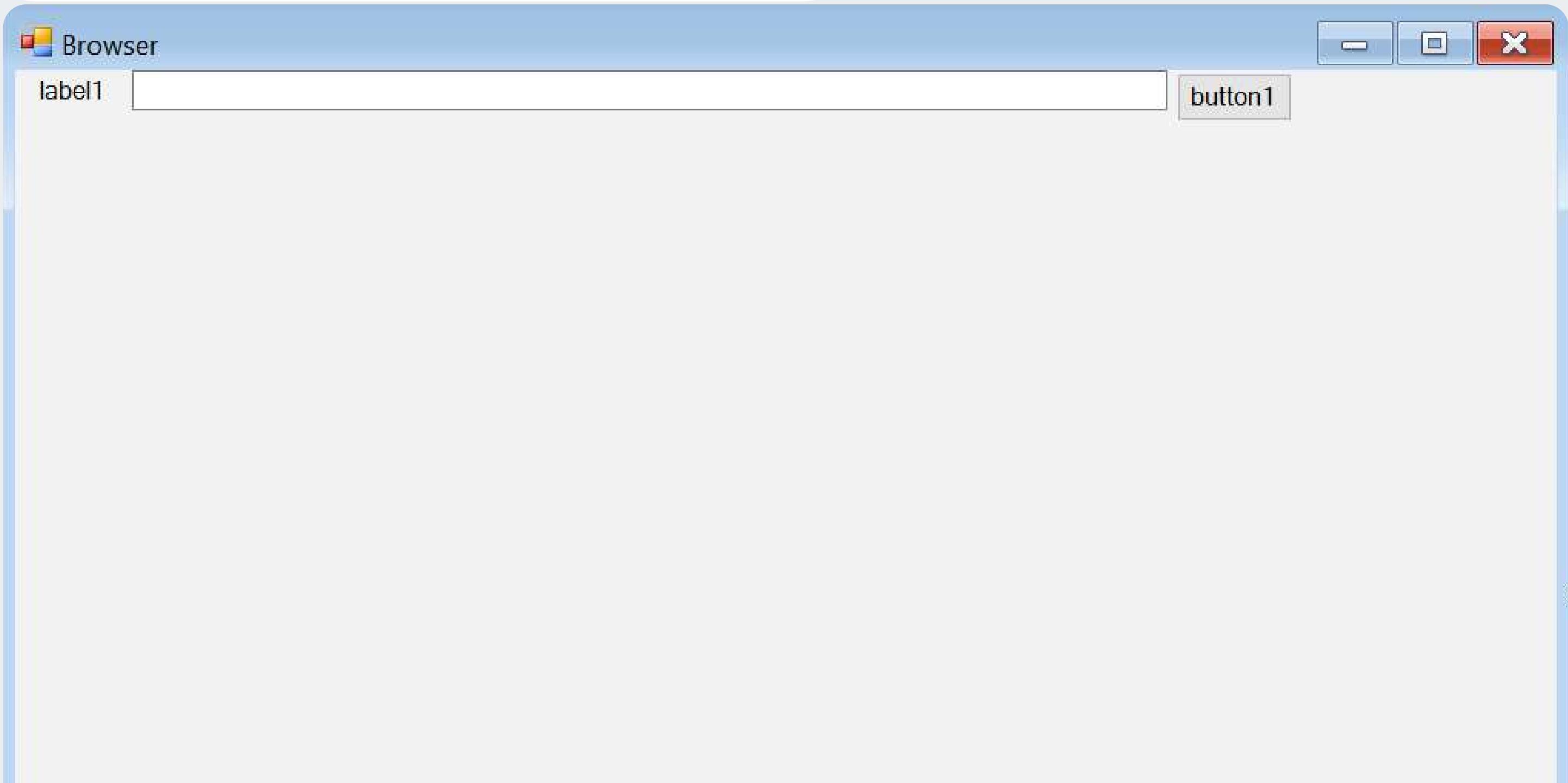


При старте окно должно разворачиваться на весь экран.

Установите свойству `WindowState` значение «`Maximized`».

# Добавим компоненты

Добавьте TextBox, Label и Button

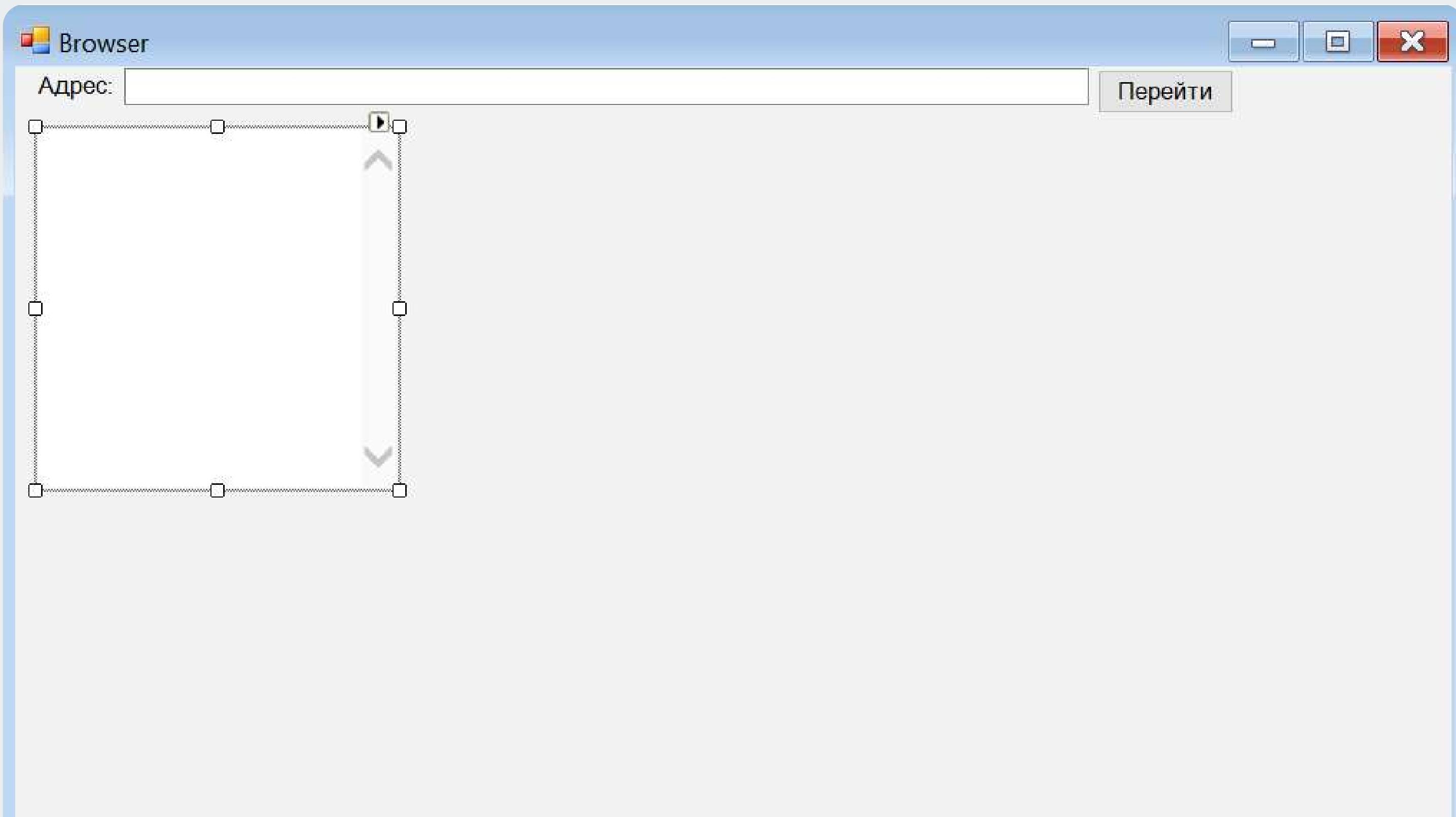


# Настройте компоненты

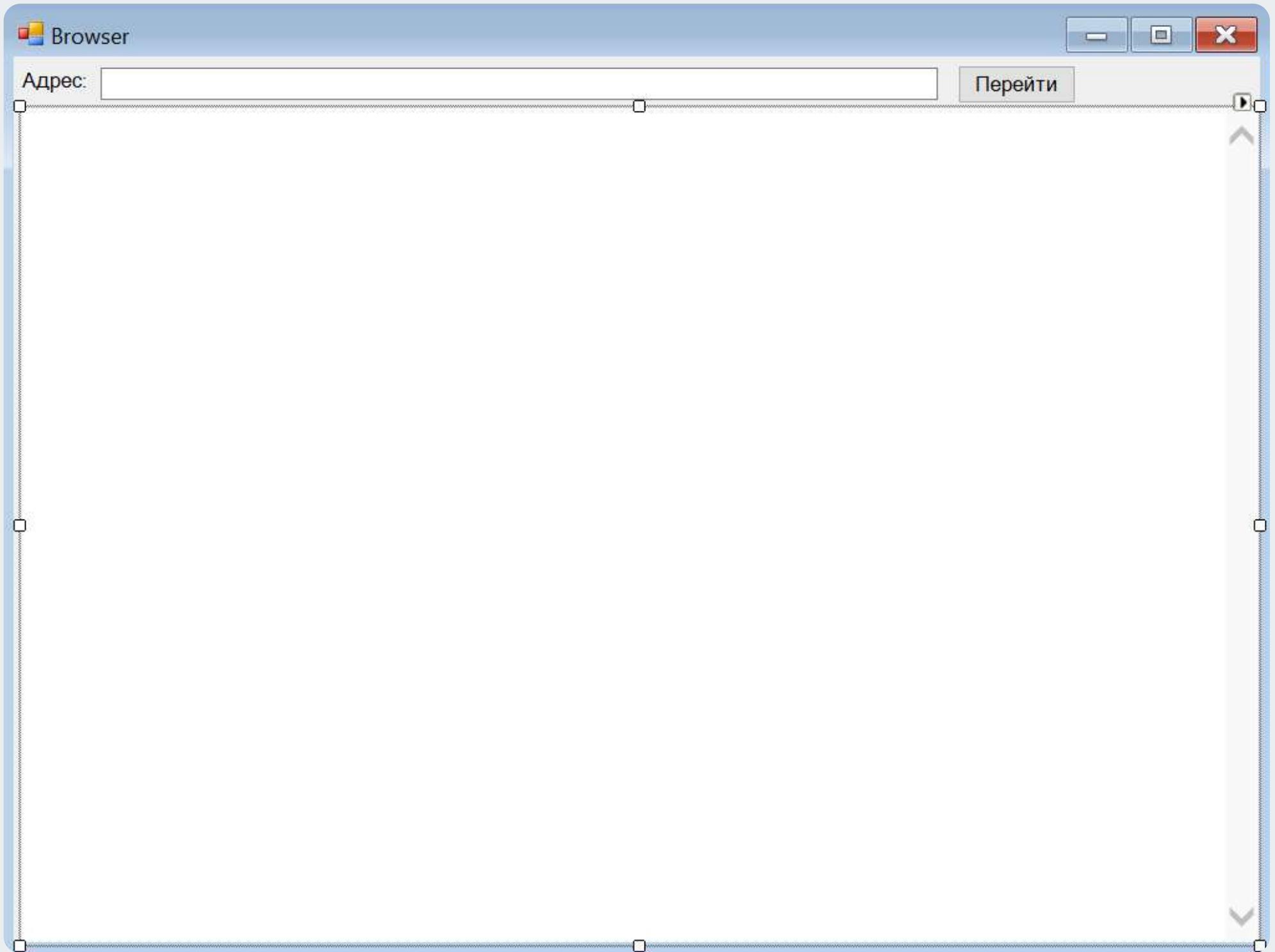
Поменяйте свойство text у кнопки и метки



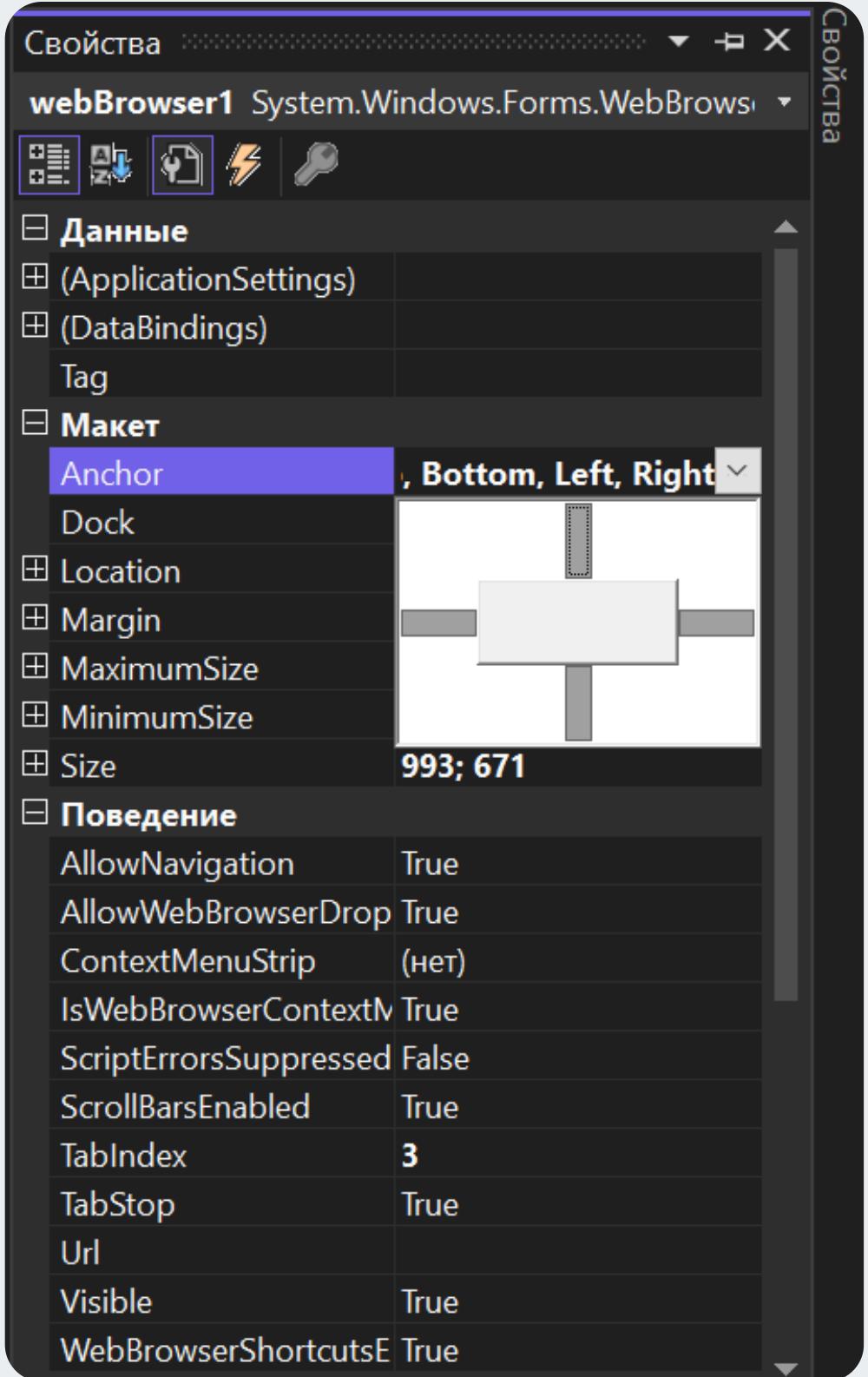
# Добавьте компонент WebBrowser



# Растяните компонент WebBrowser на свободную часть формы



# Установите привязку (Anchor)



# Создайте функцию-обработчик события click для кнопки button1

При нажатии на кнопку «Перейти» в компоненте WebBrowser должен открыться сайт с адресом, введённым в TextBox.

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->webBrowser1->Navigate(this->textBox1->Text);
}
```

# Проверьте работу приложения

Browser

Адрес: [yandex.ru](http://yandex.ru) Перейти

↗ Пермь

Настройка Завести почту Почта

Логин  
Пароль ?  
 Чужой компьютер

В ... Войти

Видéo Картинки Карты Маркет Переводчик Авто.ру Недвижимость ещё

**Яндекс**

Найдётся всё. Например, фильм, где в морозилке живёт целый город

Установите Яндекс Браузер

Погода +14 °C  
Вечером +13, ночью +2  
Сегодня: -2...+14° · ясно, без осадков · слабый ветер 1–2 м/с

Пробки 3 балла  
Прогноз 3 3 4 6 6  
15 ч 16 17 18 19

Скачайте браузер  
с защитой от сайтов с вирусами 0+

Реклама

Директ — разместить рекламу Метрика Реклама

?

Вакансии Блог Компания About Пользовательское соглашение © Яндекс

# Проверьте работу приложения

Browser

Адрес: doma.uchi.ru

Перейти

UCHi DOMA

Начальная школа  
Средняя школа  
ОГЭ / ЕГЭ  
О репетиторах  
Стоимость  
FAQ

Записаться на урок

Войти

Онлайн-занятия  
с репетиторами  
от Учи.ру

Записывайтесь на бесплатный урок

Ваше имя

Класс



# Добавим вывод адреса сайта в заголовок окна

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->webBrowser1->Navigate(this->textBox1->Text);
    this->Text = "Browser-" + this->textBox1->Text;
}
```

Browser-yandex.ru

Адрес: [yandex.ru](http://yandex.ru)

Перейти

↗ Пермь

Настройка Завести почту Почта

Логин

Пароль ?

Чужой компьютер

[В](#) [f](#) [t](#) ... Войти

Видео Картинки Карты Маркет Переводчик Авто.ру Недвижимость ещё

**Яндекс**

Найдётся всё. Например, пасьянс косынка

Найти

Погода +14 °C  
Вечером +13, ночью +2  
Сегодня: -2...+14° · ясно, без осадков · слабый ветер 1–2 м/с

Пробки 3 балла  
Прогноз 3 3 4 6 6  
15 ч 16 17 18 19

Установите Яндекс Браузер

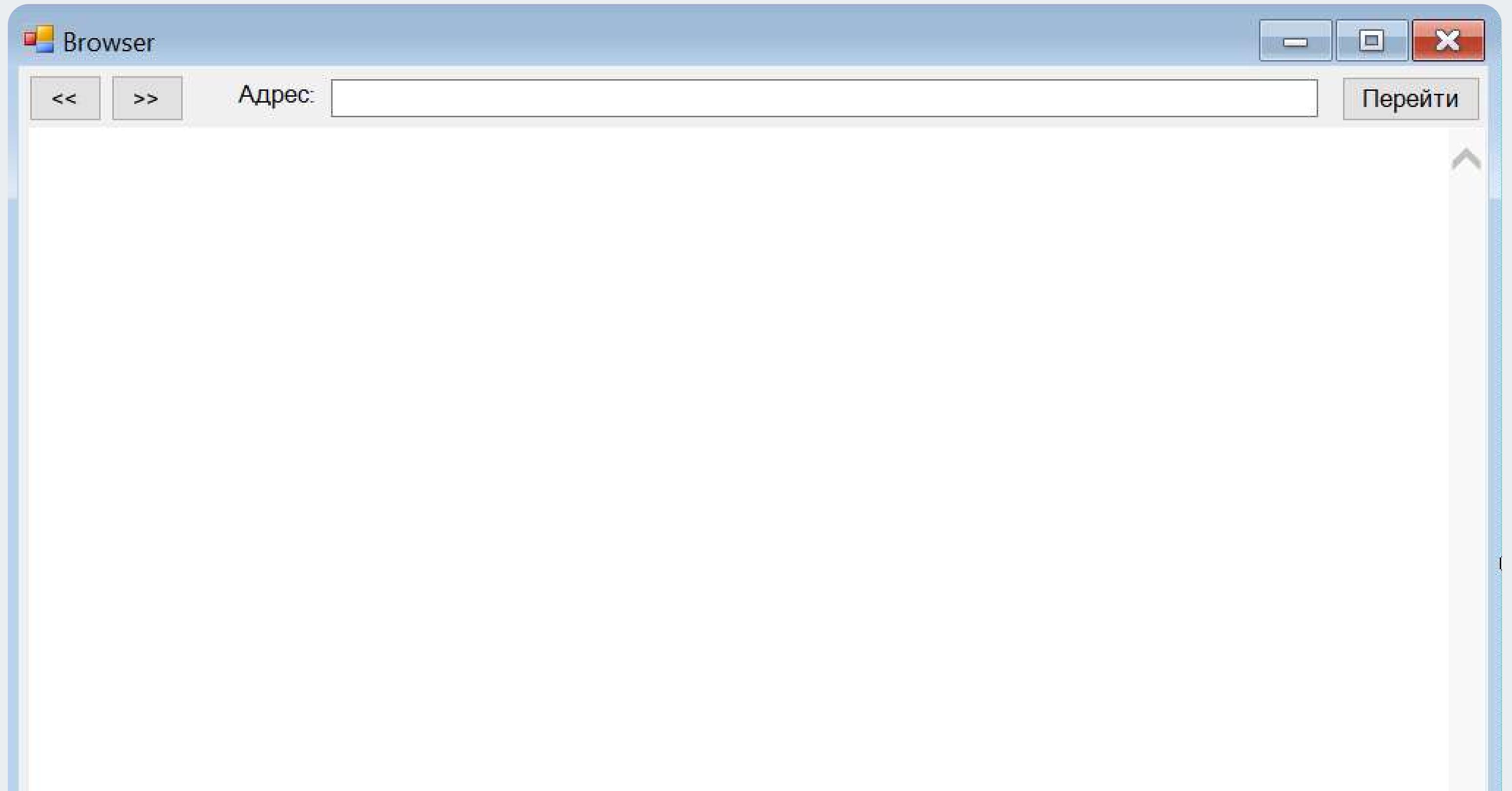
Скачайте браузер  
с защитой от сайтов с вирусами 0+

Реклама

Директ — разместить рекламу Метрика Реклама

FAQ Вакансии Блог Компания About Пользовательское соглашение © Яндекс

# Добавим кнопки «Вперёд» и «Назад»



Создадим две глобальные переменные,  
для хранения адресов сайтов  
для перехода назад и вперёд.

```
public:  
    String^ url_back;  
    String^ url_fd;
```

# Дорабатаем функцию-обработчик нажатия кнопки «Перейти»

Перед переходом на новый адрес, введённый в TextBox, будем сохранять в переменной url\_back адрес текущей страницы.

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->url_back = Convert::ToString(this->webBrowser1->Url);
    this->webBrowser1->Navigate(this->textBox1->Text);
    this->Text = "Browser-" + this->textBox1->Text;
}
```

# Создадим функцию-обработчик события click для кнопки «Назад»

Будем запоминать адрес текущей страницы в переменную url\_fd и открывать страницу, адрес которой сохранен в переменной url\_back.

```
private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->url_fd = Convert::ToString(this->webBrowser1->Url);
    this->webBrowser1->Navigate(this->url_back);

}
```

# Создадим функцию-обработчик события click для кнопки «Вперёд»

Будем запоминать адрес текущей страницы в переменную url\_back и открывать страницу, адрес которой сохранен в переменной url\_fd.

```
private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e)
{
    this->url_back = Convert::ToString(this->webBrowser1->Url);
    this->webBrowser1->Navigate(this->url_fd);
}
```

Web-браузер  
Готов!

