

## Немного интернета в проекте на C# и платформе .NET.

Связать вашу форму с интернетом позволяют два компонента **LinkLabel** и **WebBrowser**.

Особый тип меток представляют элементы **LinkLabel**, которые предназначены для вывода ссылок, которые аналогичны ссылкам, размещенным на стандартных веб-страниц. Также, как и с обычными ссылками на веб-страницах, мы можем по отношению к данному элементу определить три цвета:

- Свойство **ActiveLinkColor** задает цвет ссылки при нажатии
- Свойство **LinkColor** задает цвет ссылки до нажатия, по которой еще не было переходов
- Свойство **VisitedLinkColor** задает цвет ссылки, по которой уже были переходы

Кроме цвета ссылки для данного элемента мы можем задать свойство **LinkBehavior**, которое управляет поведением ссылки. Это свойство принимает четыре возможных значения:

- **SystemDefault**: для ссылки устанавливаются системные настройки
- **AlwaysUnderline**: ссылка всегда подчеркивается
- **HoverUnderline**: ссылка подчеркивается только при наведении на нее курсора мыши
- **NeverUnderline**: ссылка никогда не подчеркивается

По умолчанию весь текст на данном элементе считается ссылкой. Однако с помощью свойства **LinkArea** мы можем изменить область ссылки. Ссылка открывается в браузере установленном в системе по умолчанию.

**WebBrowser** предоставляет функции интернет-браузера, позволяя загружать и отображать контент из сети интернет. В то же время важно понимать, что данный элемент не является полноценным веб-браузером, и возможности по его настройке и изменению довольно ограничены.

Рассмотрим основные его свойства:

- **AllowWebBrowserDrop**: при установке для данного свойства значения `true` можно будет с помощью мыши переносить документы в веб-браузер и открывать их.
- **CanGoBack**: определяет, может ли веб-браузер переходить назад по истории просмотров
- **CanGoForward**: определяет, может ли веб-браузер переходить вперед
- **Document**: возвращает открытый в веб-браузере документ
- **DocumentText**: возвращает текстовое содержание документа
- **DocumentTitle**: возвращает заголовок документа
- **DocumentType**: возвращает тип документа
- **IsOffline**: возвращает `true`, если отсутствует подключение к интернету
- **ScriptErrorsSuppressed**: указывает, будут ли отображаться ошибки javascript в диалоговом окне
- **ScrollBarsEnabled**: определяет, будет ли использоваться прокрутка
- **URL**: возвращает или устанавливает URL документа в веб-браузере

Кроме того, **WebBrowser** содержит ряд методов, которые позволяют осуществлять навигацию между документами:

- **GoBack()**: осуществляет переход к предыдущей странице в истории навигации (если таковая имеется)
- **GoForward()**: осуществляет переход к следующей странице в истории навигации
- **GoHome()**: осуществляет переход к домашней странице веб-браузера
- **GoSearch()**: осуществляет переход к странице поиска
- **Navigate**: осуществляет переход к определенному адресу в сети интернет

Таким образом, чтобы перейти к определенному документу, надо использовать метод `Navigate`.

Создадим небольшой проект. `TextBox` будет играть роль командной строки браузера, в внизу формы ссылка на короткий Яндекс.



Для кнопки «Открыть» опишем событие:

```
private void button1_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate(textBox1.Text);
}
```

Для того чтобы сделать гиперссылку рабочей, привяжем к ней событие. Двойным щелчком мыши по тексту гиперссылки. В ней пропишем следующую строку.

```
private void linkLabel1_LinkClicked(object sender,
LinkLabelLinkClickedEventArgs e)
{
    System.Diagnostics.Process.Start("https://ya.ru/");
}
```

Должно получиться нечто подобное.

