**Хмельницький політехнічний коледж**

Відділення “Програмування”



***Людинно-машинна взаємодія***

*Практична робота №3*

*“”*

***Теоретичні відомості***

**Робота з файлами.**

Всі необхідні класи знаходяться в просторі імен System.IO.

*Створення файлу.*

Для цього скористаємося класом FileInfo з простору імен System.IO. У конструктор передаємо ім'я майбутнього файлу. Створення файлу здійснюється викликом методу Create ().

            FileInfo fi = new FileInfo ("new.txt");

            fi.Create ();

*Видалення файлу.*

Все аналогічно попередньому прикладу. Щоб видалити файл використовуємо метод Delete ().

            FileInfo fi = new FileInfo ("new.txt");

            fi.Delete ();

*Запис у файл.*

Для запису (а в подальшому і для читання) у файл ми будемо використовувати символьні потоки дозволяють оперувати безпосередньо з символами Unicode.

            StreamWriter sw = new StreamWriter ("name.txt");

            sw.WriteLine ("блаблабла");

            sw.Close ();

Після всіх дій над файлом не забуваємо закривати його використовуючи метод Close (), в іншому випадку зміни у файлі не тільки не будуть збережені. Наведений вище приклад запису файлу повністю перезаписує його, якщо ж треба дописати в кінець файлу:

            StreamWriter sw;

            FileInfo fi = new FileInfo ("name.txt");

            sw = fi.AppendText ();

            sw.WriteLine ("blablabla");

            sw.Close ();

*Читання з файлу.*

Для того щоб файл був коректно лічений (без всяких незрозумілих символів), він повинен бути збережений у юникоде.

            StreamReader streamReader = new StreamReader ("name.txt");

            string str = "";

            while (! streamReader.EndOfStream)

            {

                str + = streamReader.ReadLine ();

            }

Ще один варіант, для різноманітності

            StreamReader streamReader = new StreamReader ("name.txt");

            string str = "", str\_temp = "";

            while ((str\_temp = streamReader.ReadLine ())! = null)

            {

                str + = str\_temp;

            }

*Примітка.*

Для коректної роботи з потоками введення/виведення рекомендується використовувати конструкцію using:

            using (StreamWriter streamWriter = new StreamWriter ("name.txt"))

            {

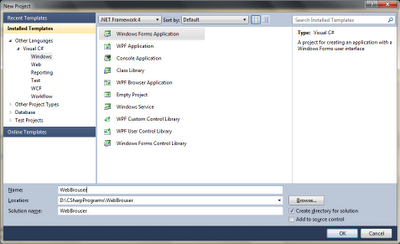
                / / ...

            }

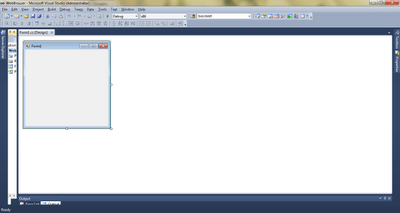
По виходу з блоку using, потік, з яким ви працювали, автоматично закриється

***Хід роботи***

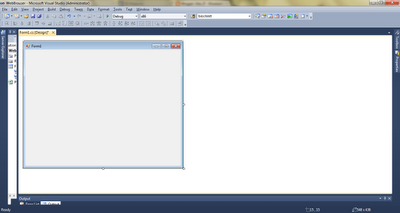
Заходимо в VS C# та вибираємо «**Создать- Новый -Проект Windows Forms**»

[](http://4.bp.blogspot.com/-gmGiquq74RE/TmH-qIZo0cI/AAAAAAAAACs/JmyZEpVHbHk/s1600/1.png)

Називаємо проект.

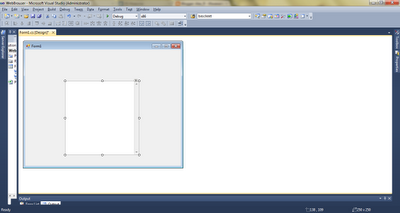
[](http://3.bp.blogspot.com/-6a2fA4lHoQU/TmILTsyN3hI/AAAAAAAAACw/UMZe0XbmmMg/s1600/2.png)

З'явилися форму можна розтягувати як хочемо, що ми і зробимо щоб збільшити наше вікно.

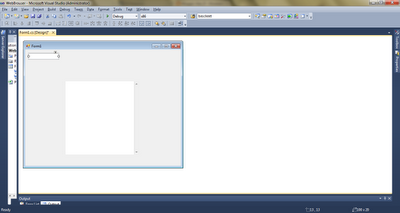
[](http://2.bp.blogspot.com/-aNHVIyhqD9c/TmILdzpSf6I/AAAAAAAAAC0/E1BuTSLgLSg/s1600/3.png)

1.Дальше з ToolBox-а вибираємо і перетягуємо на форму елемент управління WebBrouser.

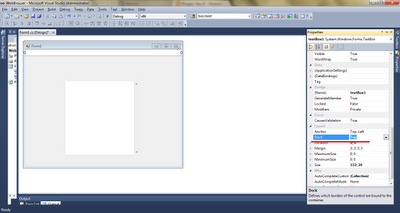
Перед вами веб браузер постав у такому вигляді як на збражені нижче, то нічого не робіть

[](http://1.bp.blogspot.com/--yNqjV4gqQM/TmIMxVPIPvI/AAAAAAAAAC4/2hFwndxBIFA/s1600/4.png)

2.Далі потрібно перетягнути з тулбокса (ToolBox) на нашу форму текстбокс (textbox).

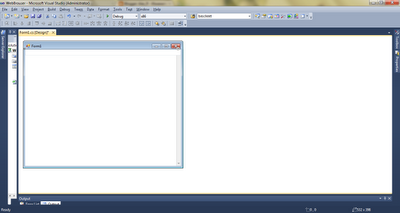
[](http://1.bp.blogspot.com/-vnHHxSC8q1s/TmINmYkbP2I/AAAAAAAAAC8/SiGjWd34bK4/s1600/5.png)

3.Поместіть його у вверхню частину екрана. Дальше натискаємо правою кнопкою миші на текстбоксі і викликаємо властивості (properties) знаходимо властивість Dock і встановлюємо його значення в Top.

[](http://4.bp.blogspot.com/-hiPF_aXiEGU/TmIPjRNlfmI/AAAAAAAAADA/wObCGp36HRc/s1600/6.png)

4.Дальше натискаємо на трикутничку на елементі управління веб-браузер (WebBrouser) і тиснемо Dock in Parent Container.

Тепер наша форма прийняла вигляд:

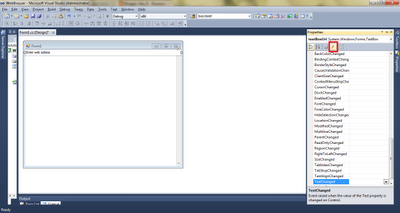
[](http://1.bp.blogspot.com/-w_Xtuy6zB1Q/TmIQ3ivkRDI/AAAAAAAAADE/8PDuv_0ulzA/s1600/7.png)

Типово Visual Studio дало текстбоксу і веб-браузер імена textBox1 webBrouser1, гарним стилем програмування вважається міняти такі імена на більш зрозумілі, тобто ім'я textBox1 не несе в собі ніякого сенсу, от якщо я зміню це ім'я на textBoxUrl, то будь-яка людина здогадається що цей текстбокс призначено введення веб адреса. Також властивість Text елемента управління textBoxUrl можна змінити з значення порожнього рядка на фразу Enter web adress, для того щоб навіть самий нетямущий користувач зрозумів що в це поле потрібно ввести назву веб адреси.

Ім'я webBrouser1 поміняти на webBrouser, бо у нас буде використовуватися тільки один елемент управління то індекс 1 можна і прибрати.

Поки що наш браузер буде виконувати тільки одну функцію: після того як користувач ввів веб адресу і натиснув на клавішу "Enter" браузер завантажить вибрану користувачем сторінку.

5.Заходимо в властивості текстбокса і переключаємось на Events (події)

[](http://3.bp.blogspot.com/-6dVrERVN8Dk/TmIa5432vJI/AAAAAAAAADU/9DHjCq43x3o/s1600/11events.png)

6. Знаходимо в списку подій KeyDown та натискаємо в поле навпроти нього два рази.

Отримали згенерований віжуал студією код.

В C# базовим для всіх класів виступає клас object, і власне наш параметр **sender** представляє об'єкт представляє подія, а другий параметр **e** який має тип **KeyEventArgs** описує цю подію, тобто за допомогою параметра **е** ми будемо дізнаватися яка клавіша натиснута коли виділений елемент текстбокс.

Прописуємо такий код:

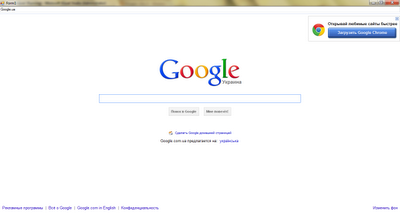
if (e.KeyCode.ToString () == "Return") / / якщо натиснута клавіша enter

webBrowser.Navigate (@ "http://www." + textBoxUrl.Text) ;/ \* / / то передаємо методу Navigate нашого елемента управління webBrowser

/ / метод Navigate сприймає рядок - адреса веб сайту. також для того що б постійно не писати http://www. ми додаємо

/ / Його до кожного адресою введеному в текст бокс символ @ перед рядком означає що в цьому рядку не будуть шукатись escape

/ / послідовності, а це означає що "/ /", сприйматися як подвійний слеш, а не як одинарний \* /

[](http://2.bp.blogspot.com/-6BxfuBfHAwQ/TmIpie61xGI/AAAAAAAAADk/tq9zslufLNQ/s1600/15s.png)

Основні методи використання:

webBrowser1.Navigate("http://www.google.com.ua"); - перехід за вказаною адресою.

webBrowser1.Refresh(); - оновити сторінку.

webBrowser1.Stop(); - зупинити завантаження, перехід сторінки.

webBrowser1.GoSearch(); - здійснити пошук.

webBrowser1.GoForward(); - перейти вперед.

webBrowser1.GoBack(); - перейти назад.

***Завдання***

Взяти за основу якийсь браузер і створити власний за функціональною подобою

вибраного та обовязково передбачити наступний функціональний набір:

- Можливість переходу на сайт за вказаним адресом.

- Можливість повернутись на попередню сторінку.

- Перехід на домашню сторінку. Встановлення домашньої сторінки.

- Зберігання історії.

- Зберігання адрес сайтів, як улюблені.

- Розгортання браузера на все вікно.

- Вікно налаштування(для зміни домашньої адреси, закладок і т.п.).

- Статусний рядок, який відображає поточний стан та конкретну інформацію про

сторінку.

**Додатково:**

- Забезпечення роботи із закладками(створення, видалення).

- Рядок заголовку повинен містити назву сайта(активної вкладки).

- Забезпечити можливість пережоду до сайту навіть якщо ввести «google.ru», а не «www.google.ru».

- Добавити інші можливості браузеру(типу перегляд коду сторінки, збереження файлу сторінки).