**Хмельницький політехнічний коледж**

Відділення “Програмування”



***Людинно-машинна взаємодія***

*Практична робота №1*

*“”*

***Теоретичні відомості***

**Типи проектів**

Новий проект можна створити з допомогою меню **File | New | Project**. У діалоговому вікні потрібно вибрати тип проекту та мову програмування. Для C# означені такі типи проектів:

|  |  |
| --- | --- |
| **Тип проекту** | **Стартовий код** |
| *Windows* |  |
| **Windows Application** | Порожня форма. |
| **Class Library** | Бібліотека класів, яку можна використовувати в довільному коді, підтримуючий .NET. |
| **Windows Control Library** | Клас .NET, який можна використати іншим кодом .NET, що має інтерфейс користувача. |
| **Web Control Library** | Елемент керування, активізується сторінками ASP.NET для генерування HTML-коду представлення елемента керування при перегляді у браузері. |
| **Console Application** | Аплікація, яка виконується з командного рядка, або у вікні консолі. |
| **Windows Service** | Служба, яка працює у фоновому режимі Windows. |
| **Empty Project** | Проект Windows Application без початкового коду. |
| **Crystal Report Windows Application** | Порожня форма проекту для генератора звітів Crystal Report. |
| *Office* | Проекти для Microsoft Word та Excel. |
| *Smart Device* | Проекти для Pocket PC, Windows Рhone. |
| *Data Base* | Проекти для SQL Server. |
| *Web Site* |  |
| **ASP.NET Web Site** | Веб-сайт на основі ASP.NET. |
| **ASP.NET Web Service** | Клас Веб-служби. |
| **Empty Web Project** | Проект ASP.NET без стартового коду. |
| **Project From Existing Code** | Нові файли для порожнього проекту, використовують для конвертування існуючого коду C#. |

**Файли проекту**

При створенні нового проекту утворює папку проекту наступної структури:



Каталоги *bin* та *obj* призначені для розташування компільованих та тимчасових файлів.

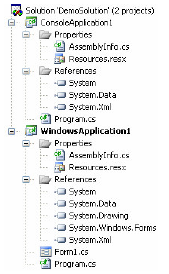
Основний каталог *WindowsApplication1*(відповідає назві проекту) містить інформацію про проект. У процесі розробки до проекту можна додавати нові каталоги.

**Рішення та проекти**

Рішення(*Solution*) – це набір усіх проектів, який утворює програмне забезпечення для вирішення проблеми.

Проект(*Project*) – це набір усіх файлів вихідного коду та ресурсів, які компілюються в один складений модуль(*Assembly*). У простих проектах тільки один модуль.

Структура рішення відображається у вікні провідника по рішенню (*Solution Explorer*). Якщо утворити два проекти то буде наступний вигляд:



Поточний проект виділяється жирним шрифтом. Рішення може містити проекти на різних алгоритмічних мовах адаптованих до платформи .NЕТ.

**Розробка проекту**

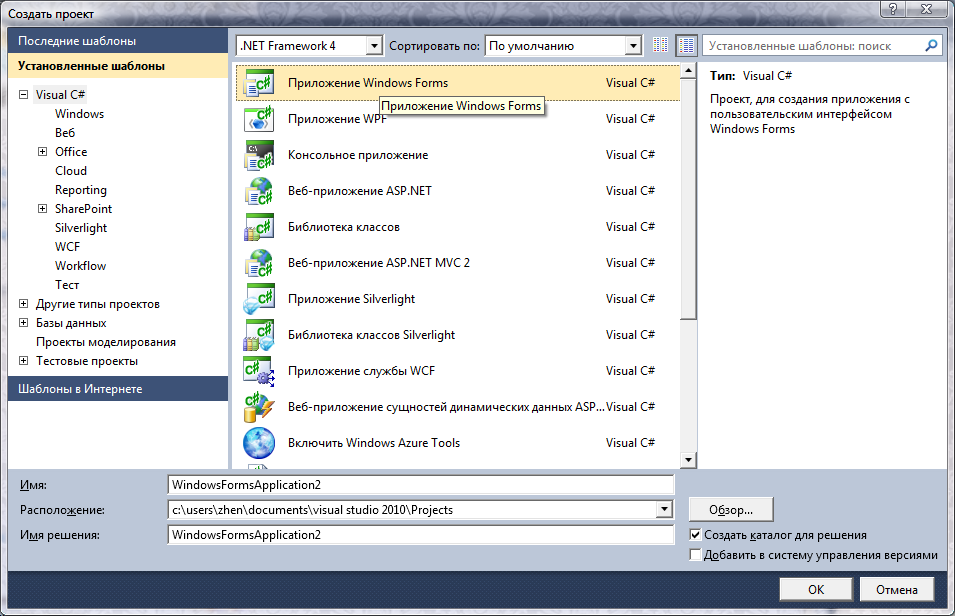
При створенні проекту середовище автоматично генерує дві конфігурації: *Debug* та *Release*. Головною відмінністю конфігурацій є те, що оптимізація коду не проводиться, а у виконавчі файли додається інформація відлагодження. Очевидно, що відлагоджений продукт повинен розповсюджуватись у *Release.*

Процес розробки проекту загалом містить етапи проектування, кодування та відлагодження.

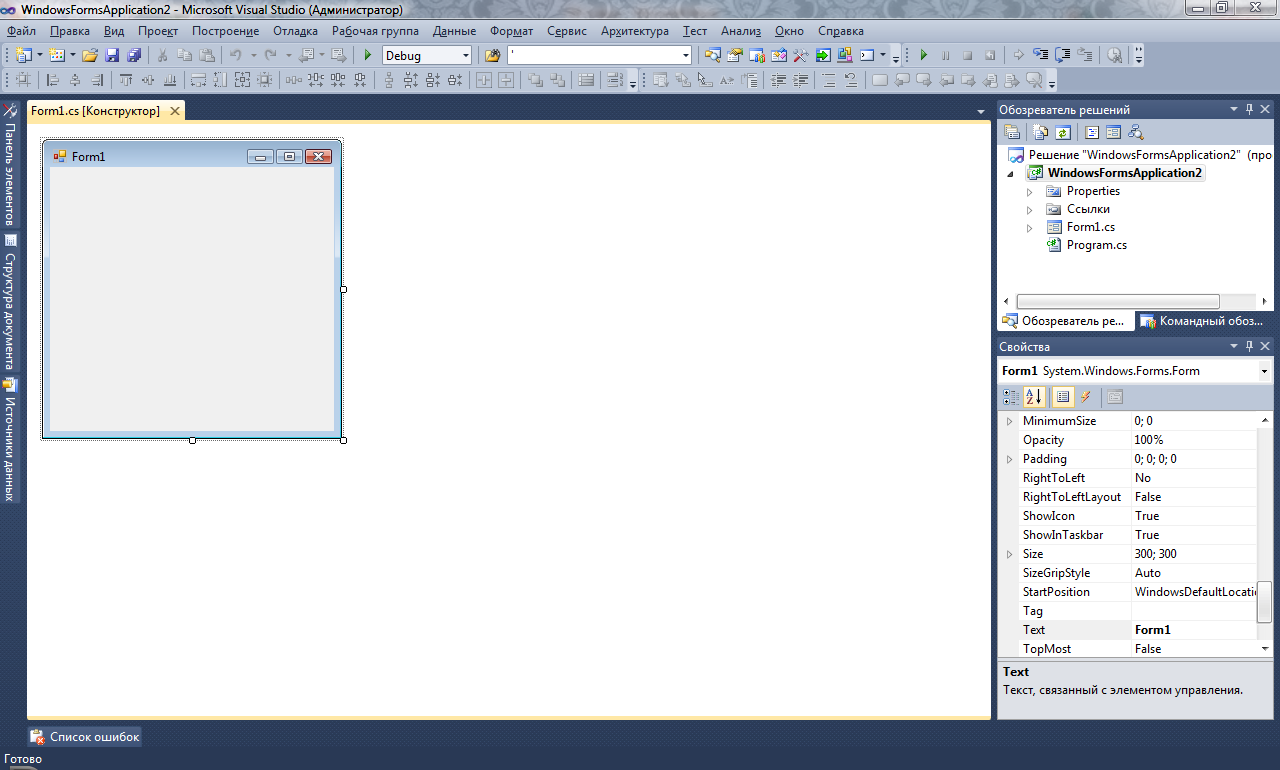
Одним з головних інструментів відлагодження є точки переривання(*Break Point*) процесу виконання програми.

***Хід роботи***

Запустимо середовище розробки ПЗ **Microsoft Visual Studio 2010**, далі в головному меню – **File|New|Project,** для створення нового проекту, у вікні що появилось(див. рис.), потрібно вибрати мову програмування Visual C# та віконий тип проекту. У полі Имя (*Name*), є автоматично згенерована середовищем назва проекту (*WindowsFormApplication1*), яку потрібно змінити, а саме потрібно латиницею написати власне прізвище, добавивши номер практичної роботи (наприклад *Ivanov02, Konoplyanka01, Kychma12*), адже Ви мусите для заліку здати викладачу усі проекти із звітами до проекту.



По завершенню натискаємо на кнопку «ОК». Перед нами появляється вікно проекту із вже створеною формою. Компілюємо проект натиснувши на кнопку F5.



Повертаємось до проекту, натискаємо клавішу F7, і відкривається вікно редактора коду.

Так як у цій практичній роботі потрібно буде реалізувати власний клас, тому і розглянемо як це здійснити. Найпростіше у просторі імен проекту написати наш клас.



Але найправильніше, додати в проект новий файл класу **File|New|File,** де потім вибрати значення файл класу, якому потрібно надати назву.

Після того як реалізовано клас, потрібно на формі розташувати елементи керування для керування класом, вибираючи необхідні компоненти із панелі елементів, яку можна вмикнути комбінацією клавіш Ctrl+W, або ж скориставшись меню **View**.

Для використання створеного класу потрібно на подію, вибраного елемента керування прописати код опису змінної класу:



А для того щоб створити об’єкт, потрібно викликати конструктор для екземпляру:



Приклад реалізації класу:



Приклад використання класу:





***Завдання***

Ознайомтесь із меню середовища. Ознайомтесь із компонентами та їх властивостями.

Створіть клас згідно варіанту. Реалізуйте його всіма необхідними методами. Розробіть програму для використання класу. Забезпечте створення масиву об’єктів та роботу із кожним елементом масиву.

Використовуючи інформацію про сприймання та інтерфейс користувача, розташуйте елементи керування, та змініть їх властивості, що не будуть перечити правилам КІ.

**ВАРІАНТИ ЗАВДАНЬ**

1. Бібліотека (книга, автор, читач, прокат)
2. Відділ кадрів (людина, звання, посада, штатний розклад)
3. Склад (тип товара, товар, розхід/прихід, організація)
4. Облік успішності (студент, оцінка, дисципліна, вид оцінки)
5. Транспорт (тип транспорта, транспорт, маршрут, зупинка)
6. Лабіринт (рівень, клітка, двері/стінка, артефакт)
7. Фотоапарат (знімок, галерея, режим знімання, відео, режим відео)
8. Плеєр (трек, виконавець, альбом, плейліст)
9. Транспорт2 (водій, перевезення, маршрут, тип транспорта)
10. Туризм (маршрут, ділянка, категорія ділянки, похід)
11. Альбом (альбом, фотокартка, тема, часовий період)
12. Телеіндустрія (телепрограма, телепередача, показ, відповідальні)
13. Морський бій (поле, корабль, палуба, гравець)
14. Тетріс (поле, фігура, кубик, гра(рівень))
15. Шахи/шашки (фігура, гра, гравець, турнір)
16. Проект (проект, задача, виконавець, стан задачі)
17. Банк (кредит, людина, депозит, менеджер)
18. Мережа магазинів (людина, покупка, продавець, магазин)
19. Телефонні розмови (клієнт, поповнення, розмова, блокування)
20. Кухня (продукт, блюдо, кухня, тип блюда)
21. Електронний документ (документ, частина, пункт, абзац)
22. Географ. атлас (країна, область, район, національність, місто, пам’ятка)
23. Обсерваторія (сузір’я, зірка, планета, супутник, відкривач)

До кожної теми потрібно написати звіт, який повинен містити наступне:

* Тему практичної роботи;
* Короткі теоретичні відомості(при необхідності);
* Завдання;
* Хід роботи;
* Текст програми;
* Результати;
* Діаграму класів(при реалізації програм з використанням ООП)
* Висновок.