

**Міністерство освіти і науки України**

**Харківський радіотехнічний коледж**

**ПРАКТИЧНА РОБОТА**

**Windows - програма роботи з керуючими елементами типу List Box та Edit Control панелі елементів головного діалогового вікна з використанням класів CListBox, CDialog, CString та CEdit бібліотеки MFC**

навчальна дисципліна

**„Системне програмування”**

спеціальність 123

**„ Комп’ютерна інженерія ”**

**Укладач:**

Заступник директора з НВР,  
викладач комп’ютерних дисциплін вищої кваліфікаційної категорії  
Пуйденко В.О.

**ХАРКІВ 2020**

# Практична робота

## Windows - програма роботи з керуючими елементами типу List Box та Edit Control панелі елементів головного діалогового вікна

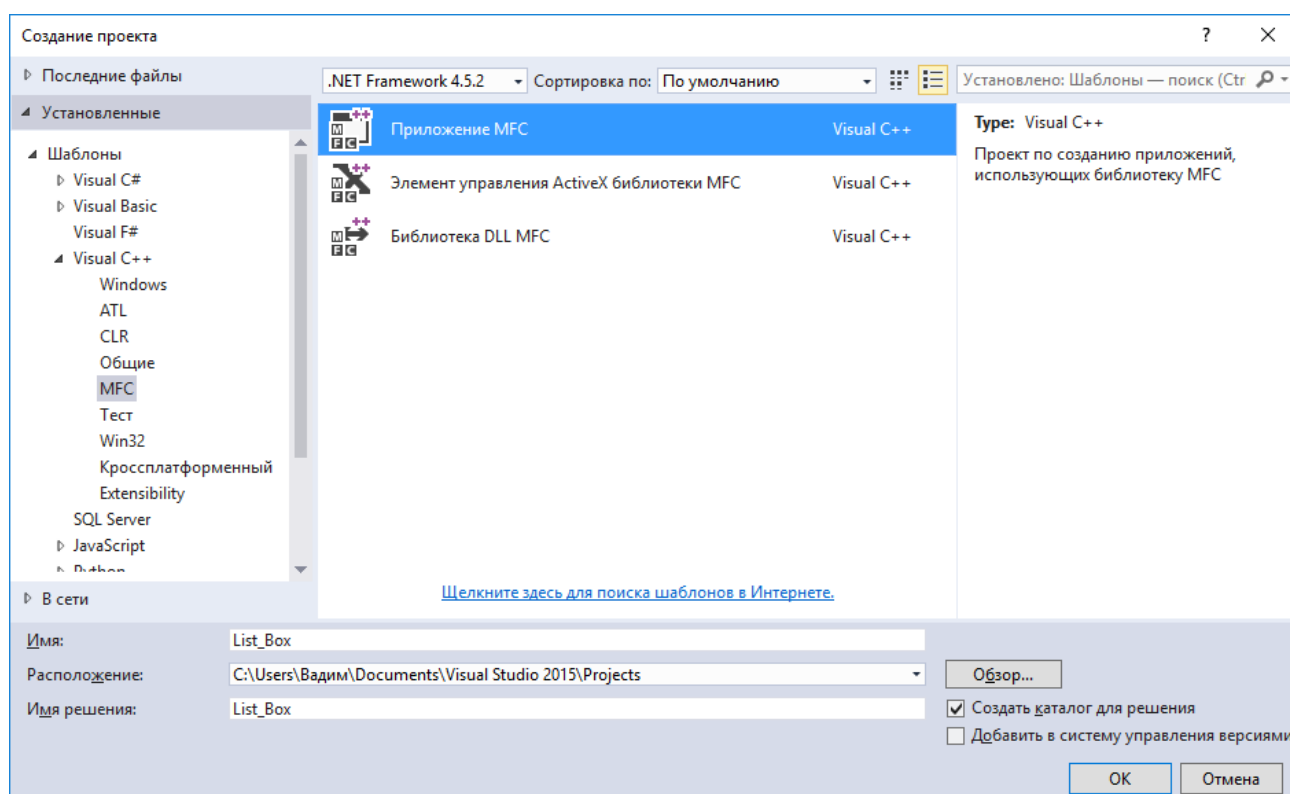
### 1 Мета роботи

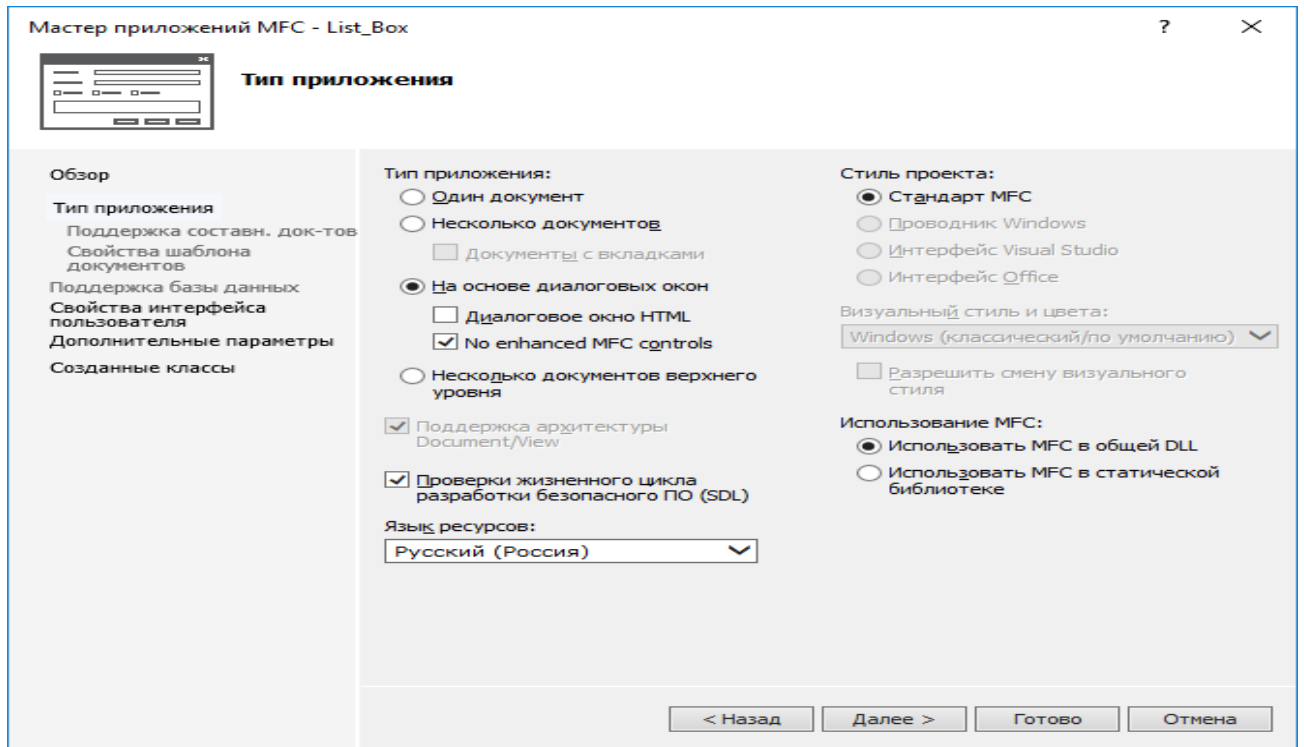
Удосконалення теоретичних навичок роботи з класами CListBox, CDialog, CString та CEdit бібліотеки MFC з застосуванням керуючих елементів типу List Box та Edit control.

### 2 Послідовність виконання практичної роботи

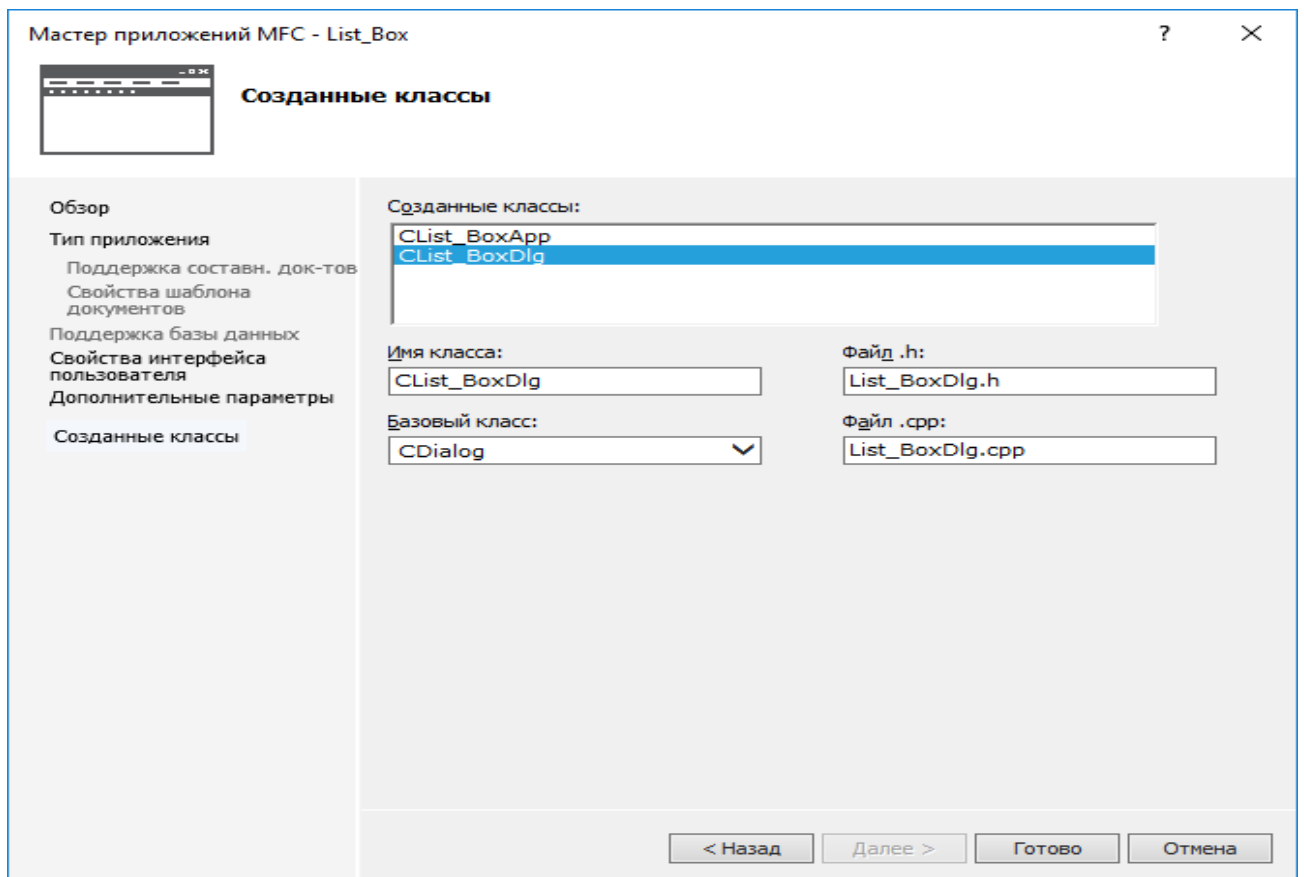
#### 2.1 Робота з майстром додатків по створенню Windows - проєкту на Visual C++ з MFC на базі діалогового вікна

Використовуючи майстер додатків створимо новий Windows - проєкт на Visual C++ з MFC з ім'ям List\_Box:



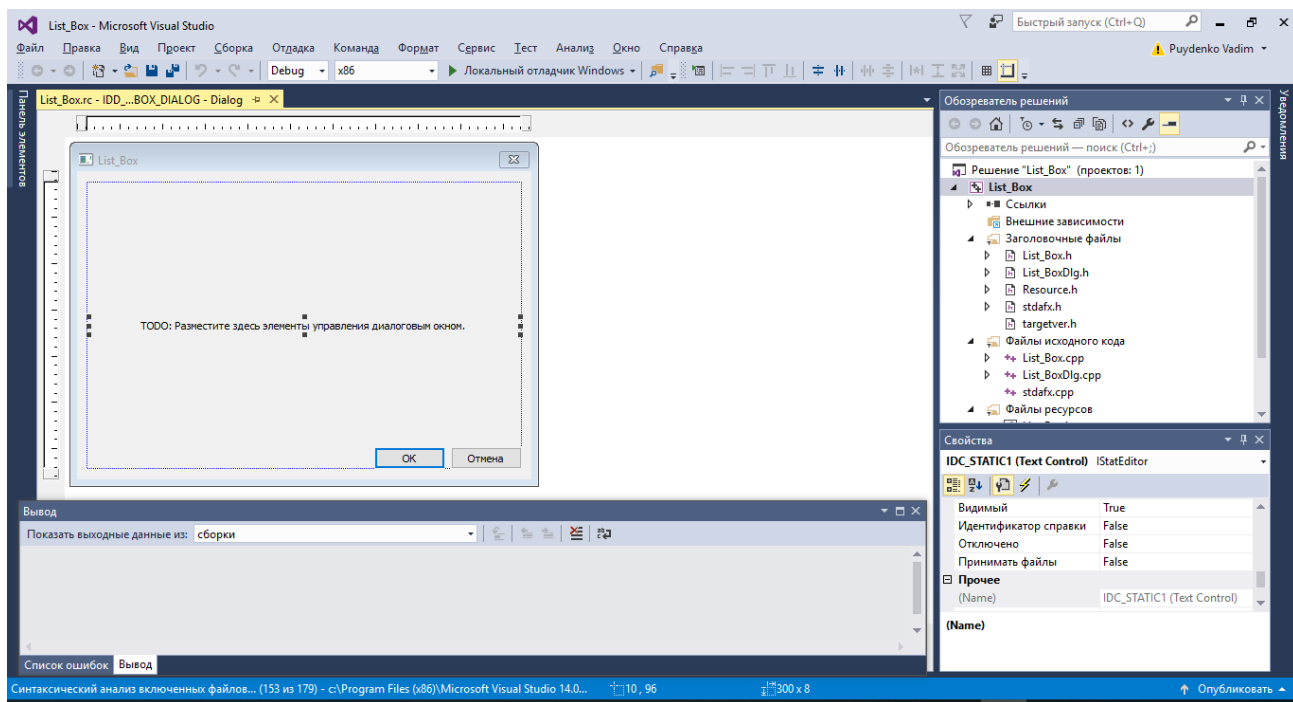


Продовжуючи використовувати майстер додатків зробимо активним представлений похідний клас «CList\_BoxDlg» від базового класу CDialog:

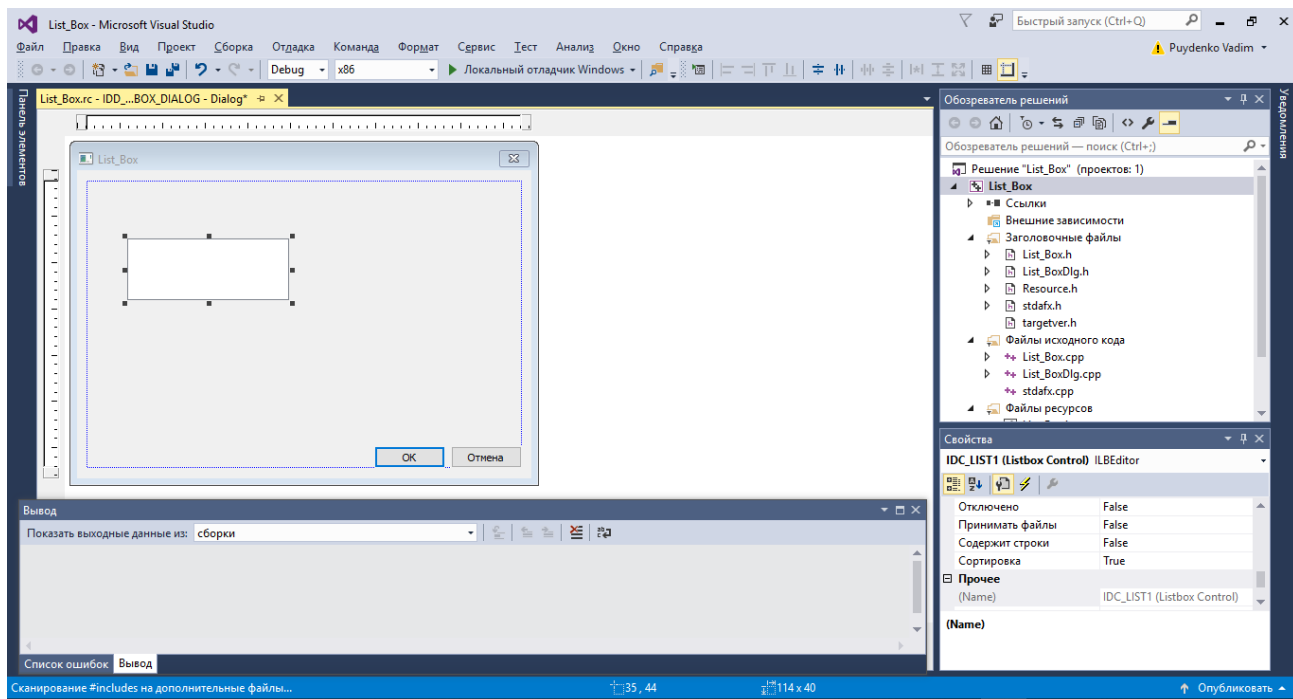


## 2.2 Работа с оглядачем рѣшень, файлом ресурсов, панелью элементов та вѣнком властивостей

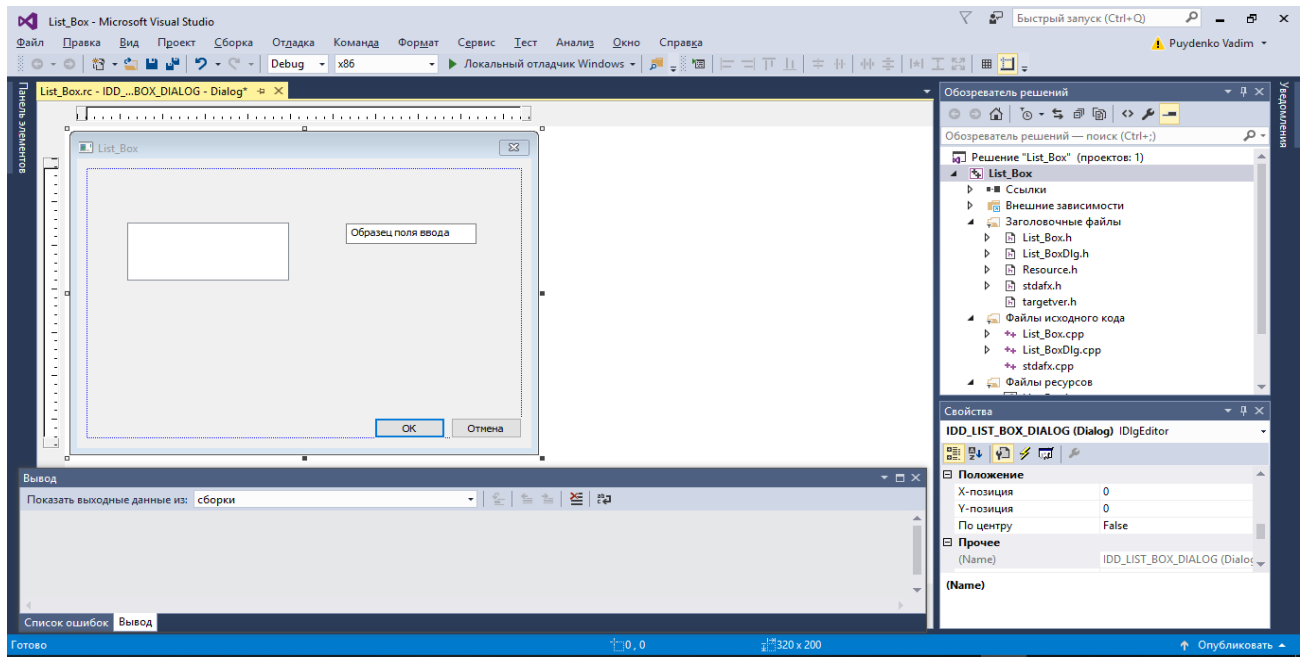
Оглядач рѣшень мѣстит файл ресурсов List\_Box.rc з дѣалоговим вѣнком:



Додамо до дѣалогового вѣнка керуючий элемент «List Box» - натискаємо «Панель элементов» та розташовуємо элемент List Box:

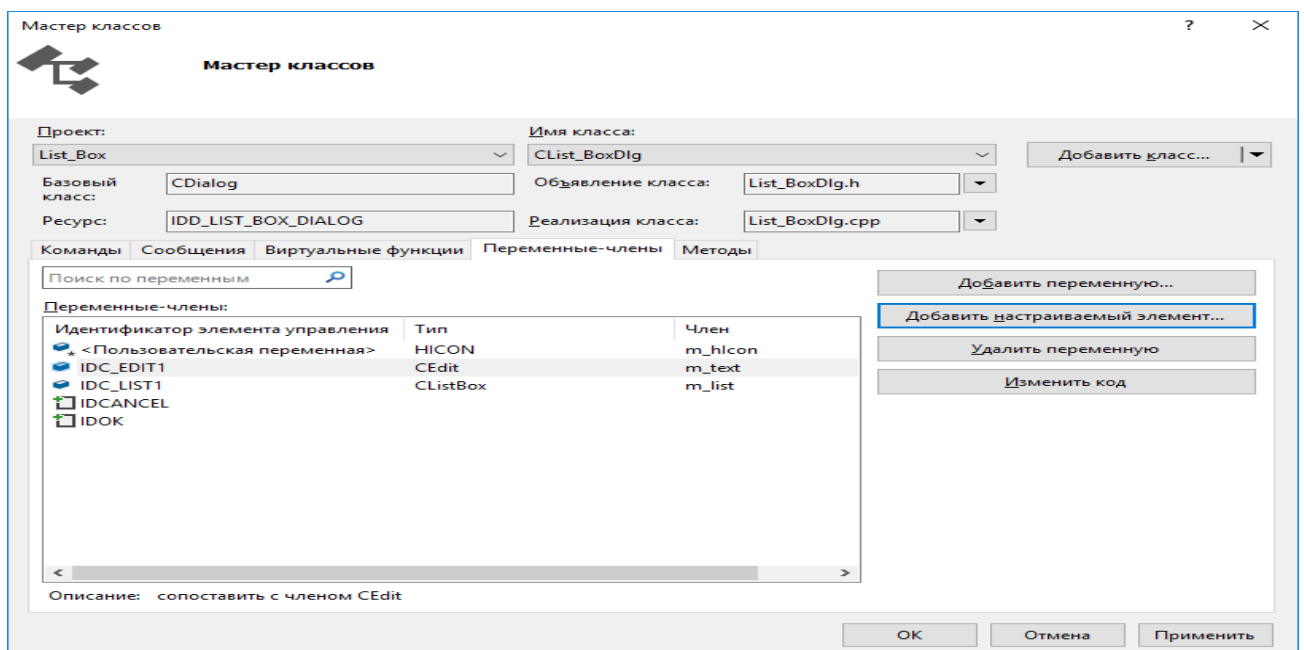


Додамо до діалогового вікна керуючий елемент «Edit Control» - натискаємо «Панель элементов» та розташовуємо елемент у поле діалогового вікна:

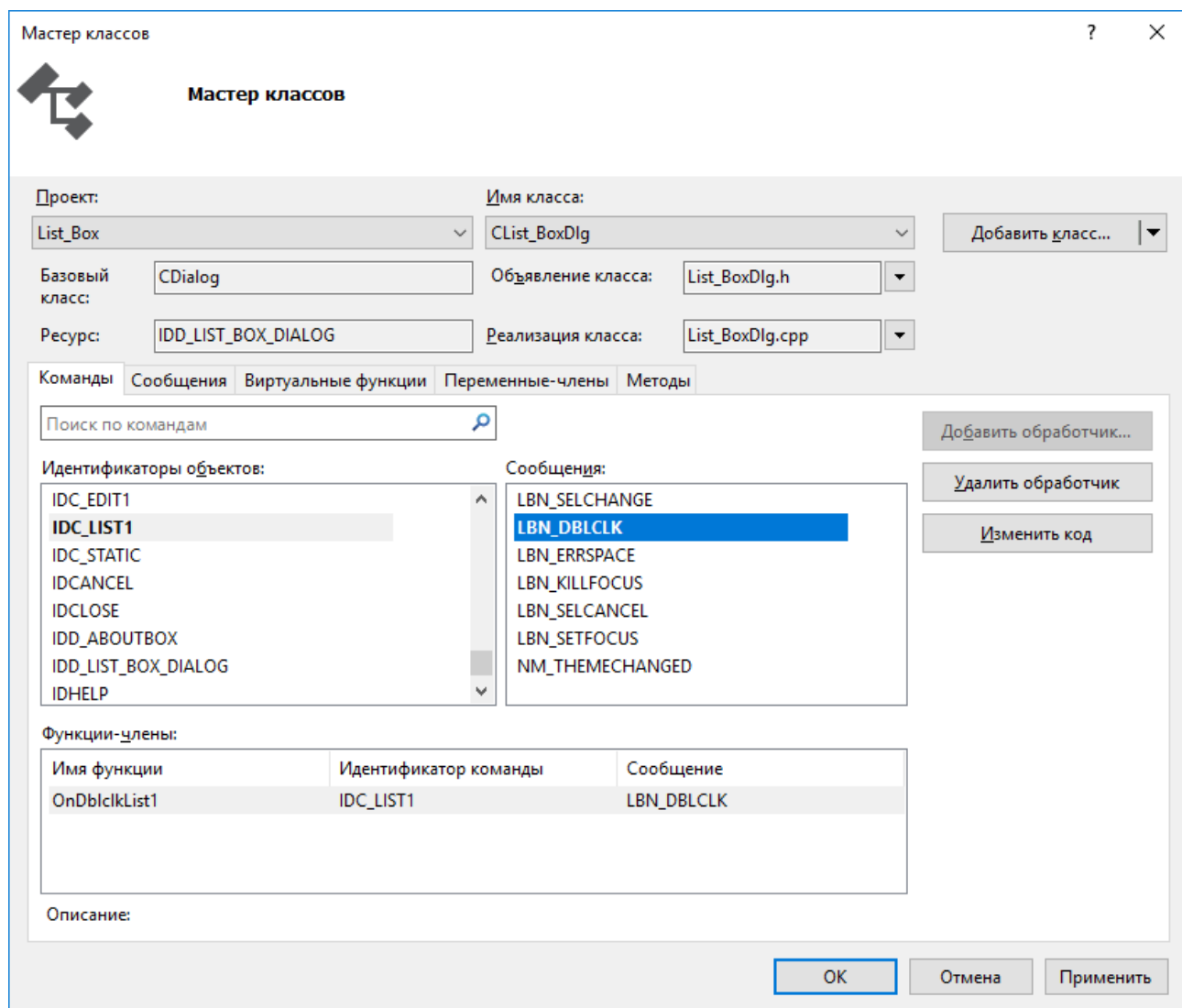


## 2.3 Работа з майстром класів

В майстрі класів створити об'єкти: `m_list` на клас `CListBox` та `m_text` на клас `CEdit`.



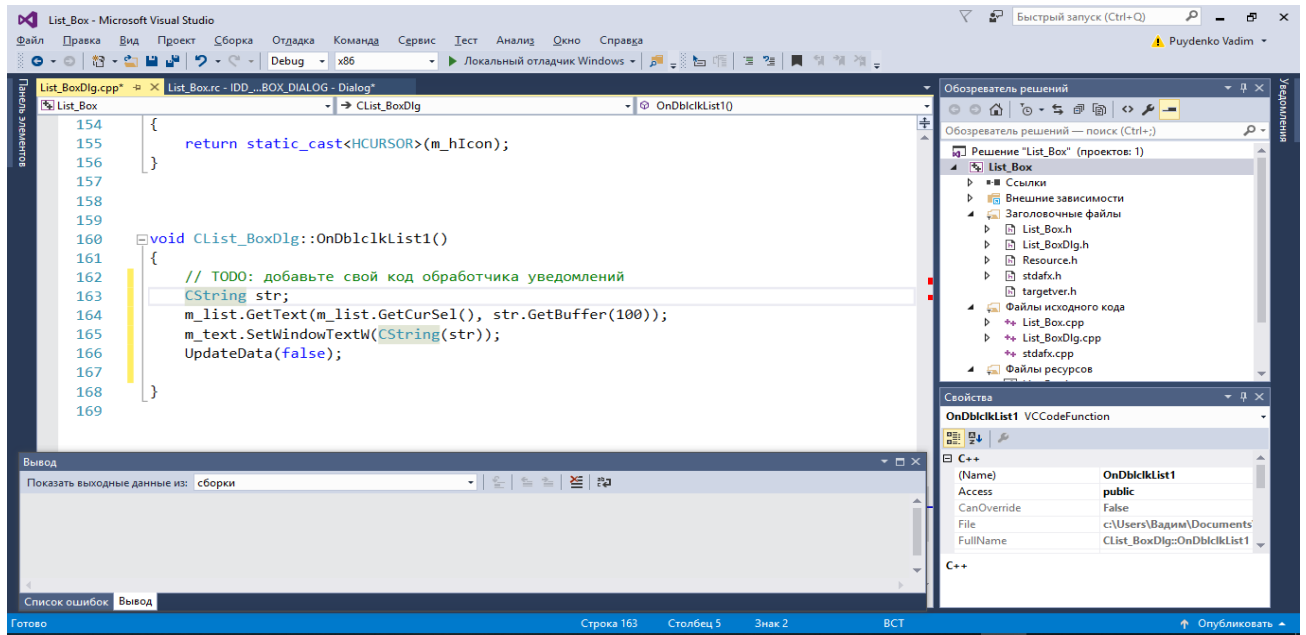
Далі, через команду IDC\_LIST1 пов'яжемо повідомлення операційної системи LBN\_DBLCLK з відповідним обробником OnDblclkList1():



## 2.4 Работа з файлом вихідного коду

Пропишемо у нашому обробнику наступний програмний код, де разом з копіюванням відповідного текстового елемента переліку у буфер забезпечимо отримання індексу поточного обраного рядка переліку та організуємо вивід інформації про подію обраного елемента у текстове поле:

```
CString str;  
m_list.GetText(m_list.GetCurSel(), str.GetBuffer(100));  
m_text.SetWindowTextW(CString(str));  
UpdateData(false);
```

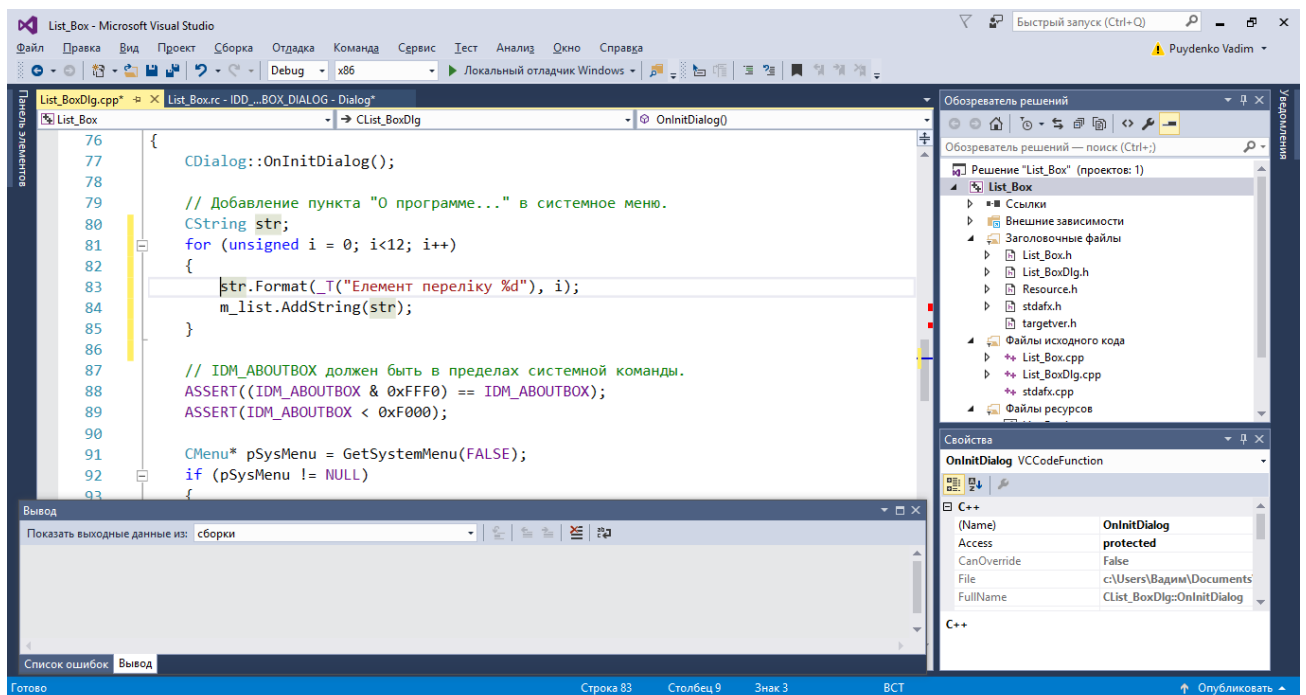


Також, знайдемо метод OnInitDialog(), де організуємо заповнення переліку для керуючого елементу List Box:

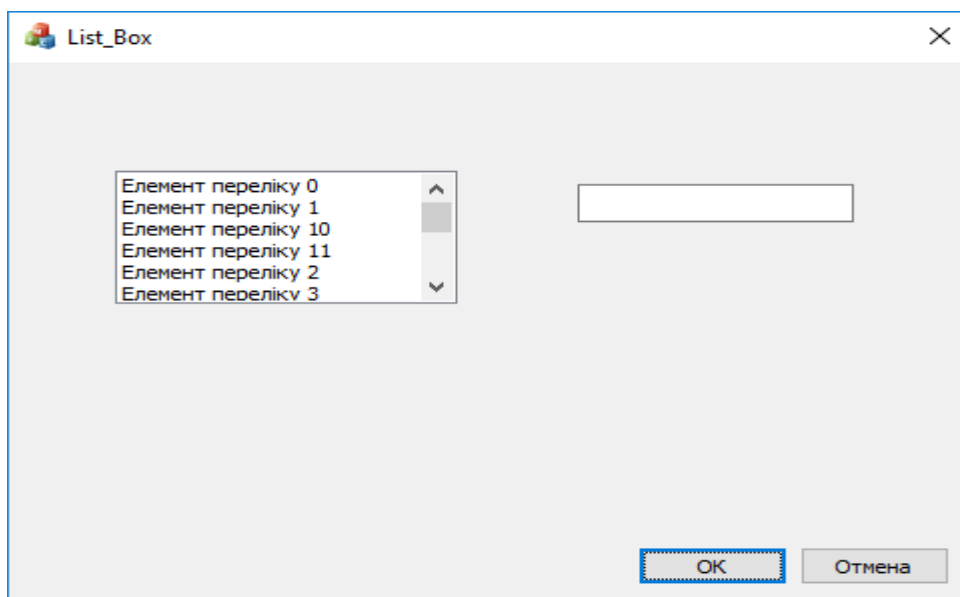
```

CString str;
for (unsigned i=0; i<12; i++)
{
str.Format(_T("Елемент переліку %d"), i);
m_list.AddString(str);
}

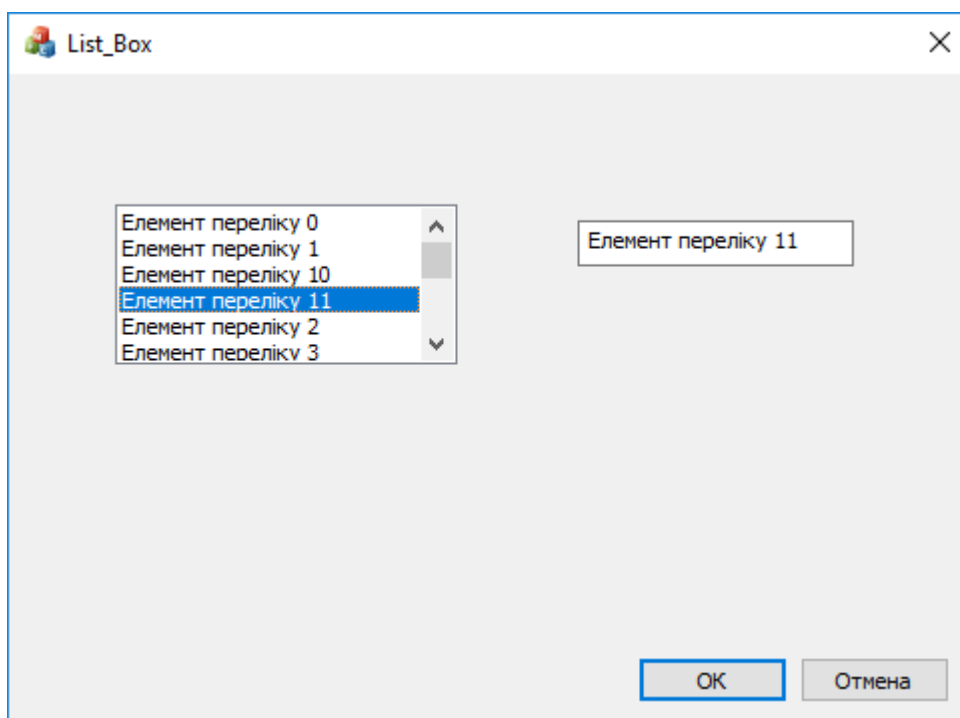
```



Windows-програма List\_Box готова. Зберемо рішення Ctrl+Shift+B та запустимо проект Ctrl+F5. На моніторі з'явиться головне діалогове вікно з відповідними керуючими елементами:



При активації, через подвійне натискання лівої кнопки системної мишки, відповідного рядка переліку елемента List Box у текстове поле діалогового вікна буде виводитись інформація про його активацію:





### **3 Порядок виконання роботи**

3.1 В інтегрованому середовищі Visual Studio створити відповідний Windows – проєкт на Visual C++ з використанням бібліотеки MFC на базі діалогового вікна з рекомендованими керуючими елементами, дотримуючись послідовності виконання практичної роботи за п. 2.

3.2 Зібрати рішення Windows – проєкту: Ctrl+Shift+B.

3.3 Запустити зібраний Windows – проєкт: Ctrl+F5.

3.4 Отримати результати дії Windows – проєкту на моніторі ПК.

### **4 Зміст звіту**

4.1 Звіт оформлюється із урахуванням вимог до навчальної документації і містить всі розкриті пункти виконання роботи.

4.2 Результати роботи програми копіюються з монітору у текстову частину звіту.

4.3 Висновки з роботи.