

Міністерство освіти і науки України

Харківський радіотехнічний коледж

ПРАКТИЧНА РОБОТА

**Windows - програма роботи з керуючими елементами типу Check Box та
Edit control панелі елементів головного діалогового вікна
з використанням класів CButton, CDialog 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 Check Box та Edit Control панелі елементів головного діалогового вікна

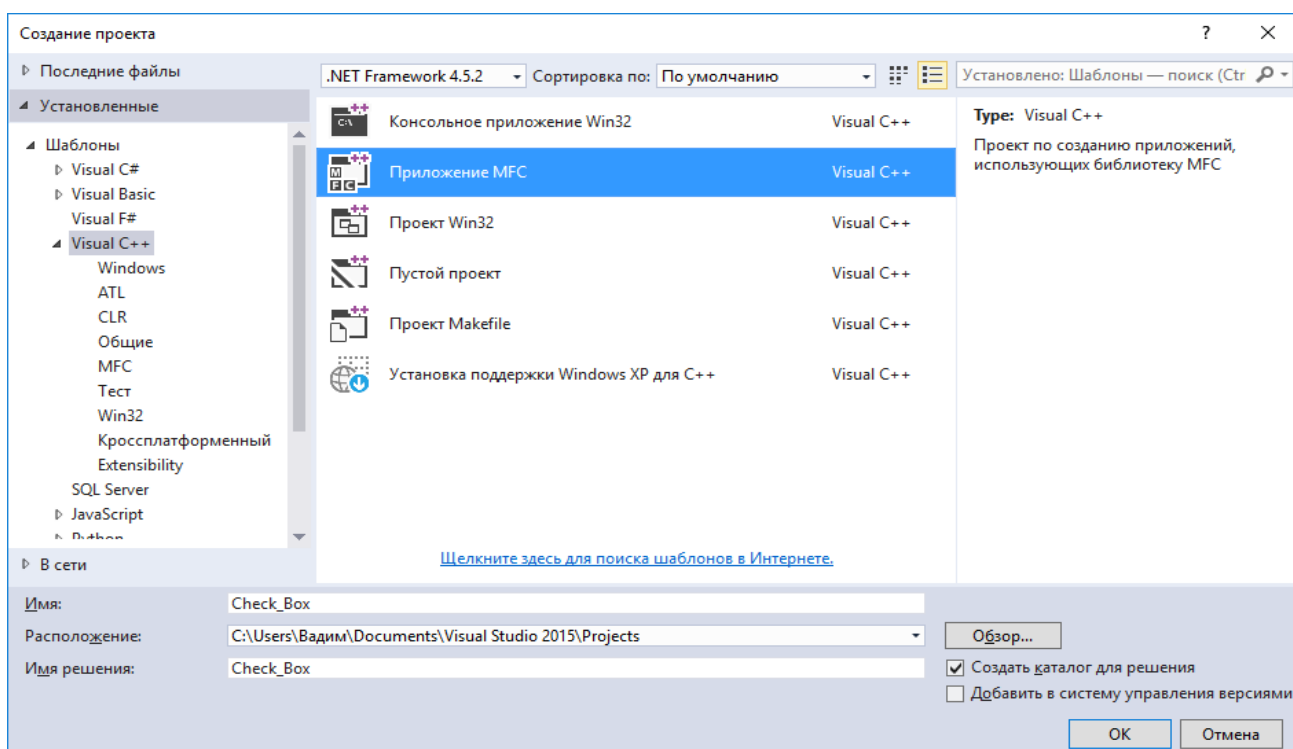
1 Мета роботи

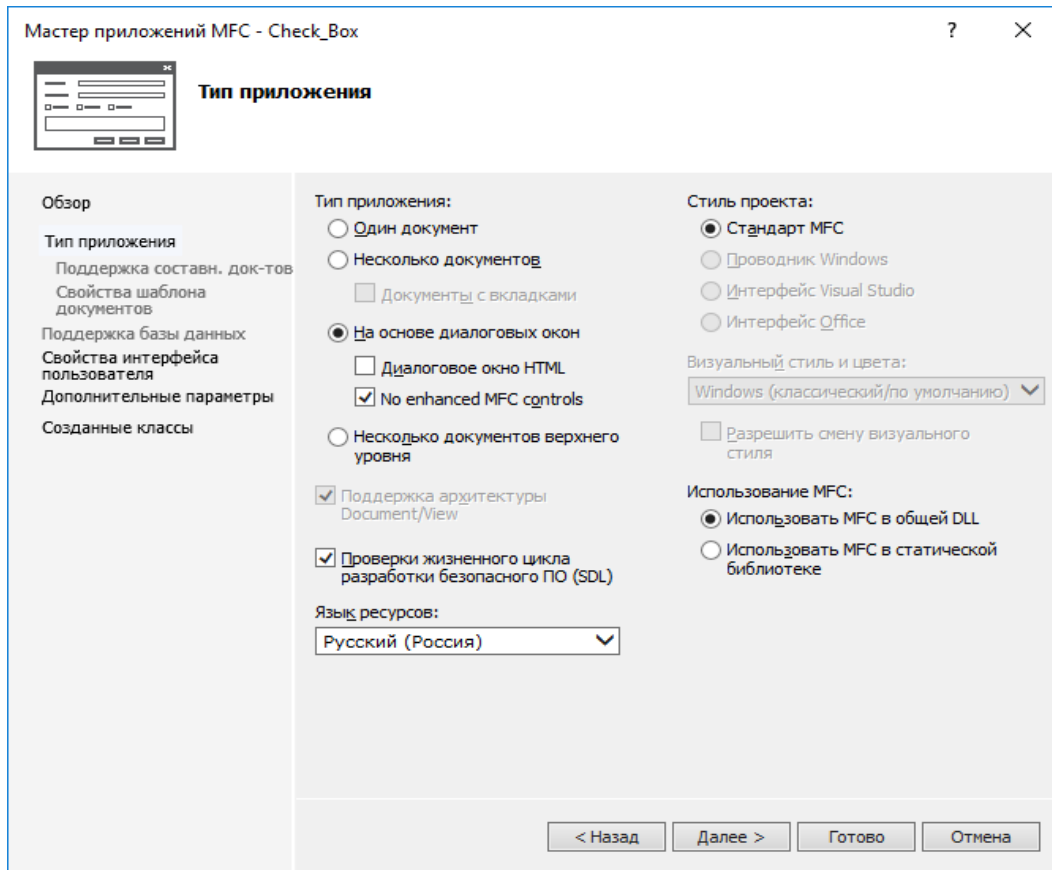
Удосконалення теоретичних навичок роботи з класами CButton, CDialog та CEdit бібліотеки MFC з застосуванням керуючих елементів типу Check Box та Edit control.

2 Послідовність виконання практичної роботи

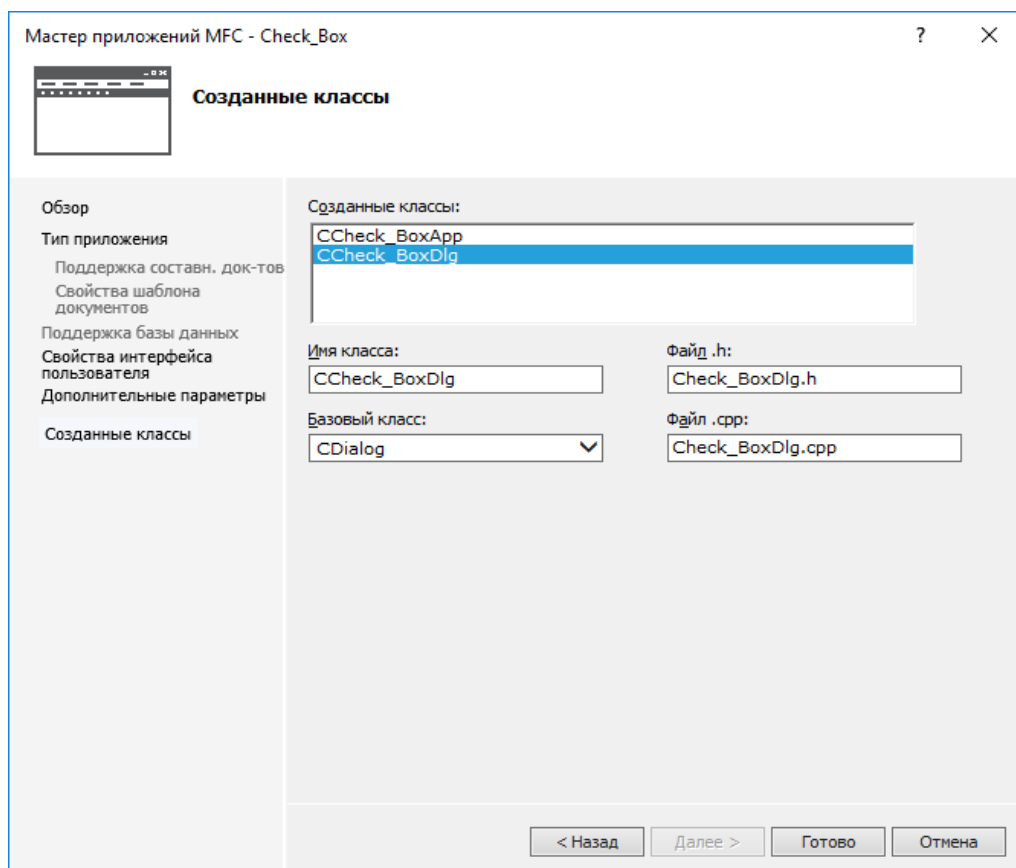
2.1 Робота з майстром додатків по створенню Windows - проєкту на Visual C++ з MFC на базі діалогового вікна

Використовуючи майстер додатків створимо новий Windows - проєкт на Visual C++ з MFC з ім'ям Check_Box, де встановимо перемикачі «No enhanced MFC control» та «Стандарт MFC»:



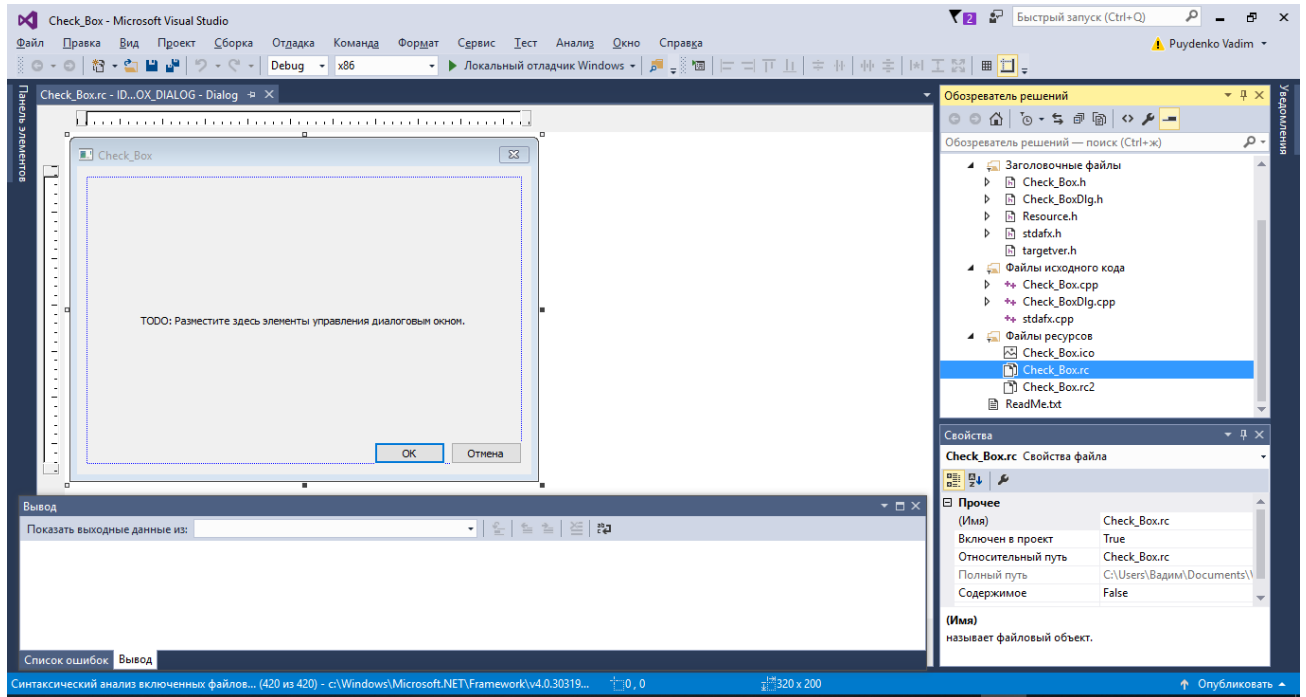


Продовжуючи використовувати майстер додатків зробимо активним представлений похідний клас «CCheck_BoxDlg» від базового класу CDialog:

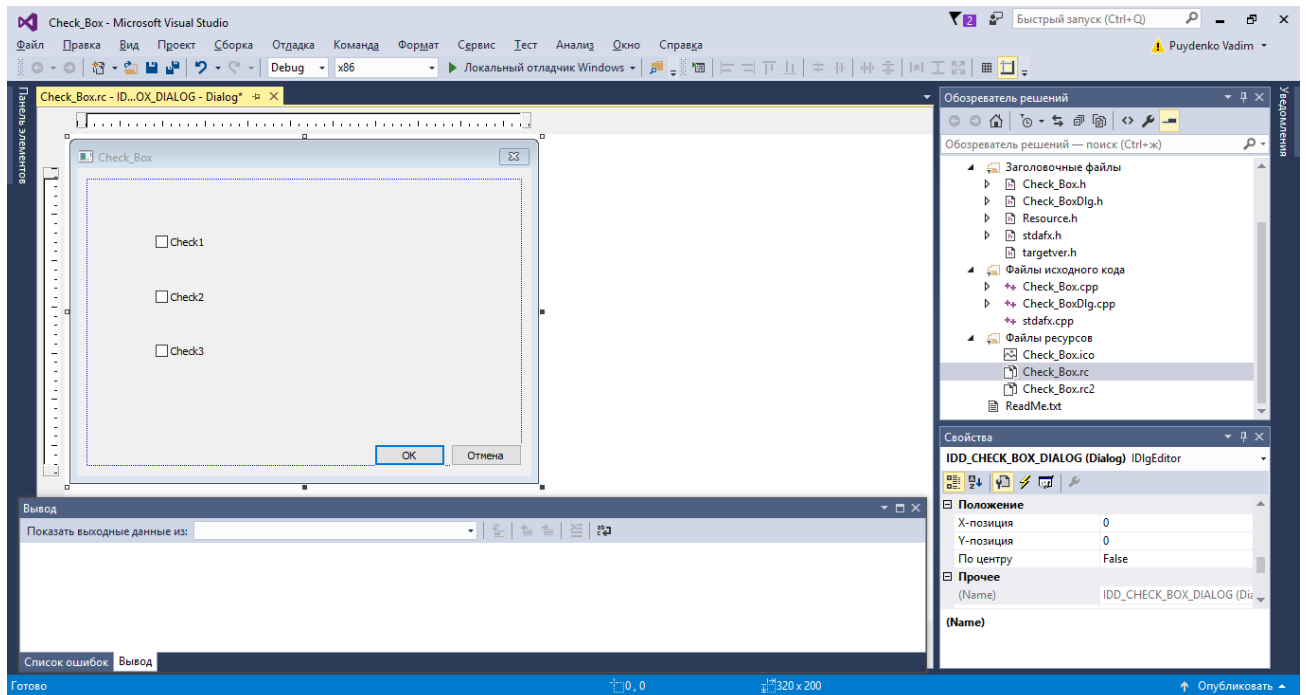


2.2 Работа с оглядачем рѣшень, файлом ресурсов, панелью элементов та вѣнком властивостей

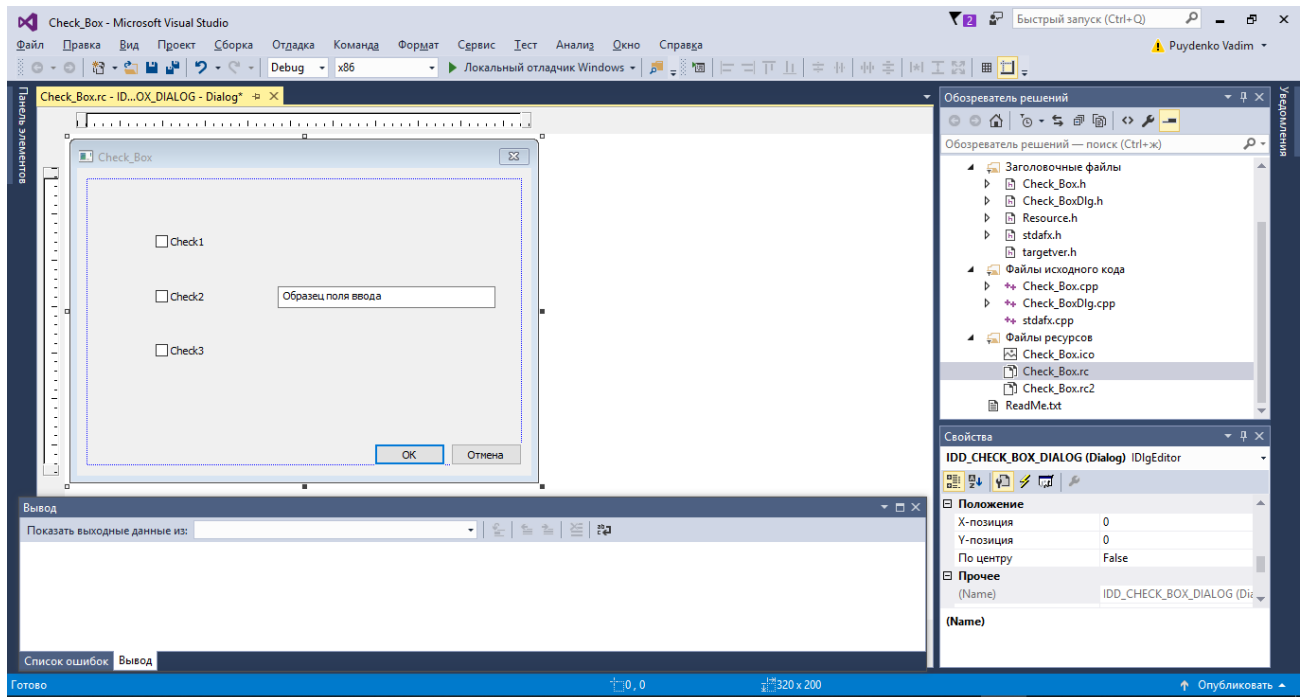
Оглядач рѣшень мѣстит файл ресурсов Check_Box.rc з дѣалоговим вѣнком:



Додамо до дѣалогового вѣнка три керуючих элемент «Check» - натискаємо «Панель элементов» та послѣдовно перетягуємо элемент Check Box:

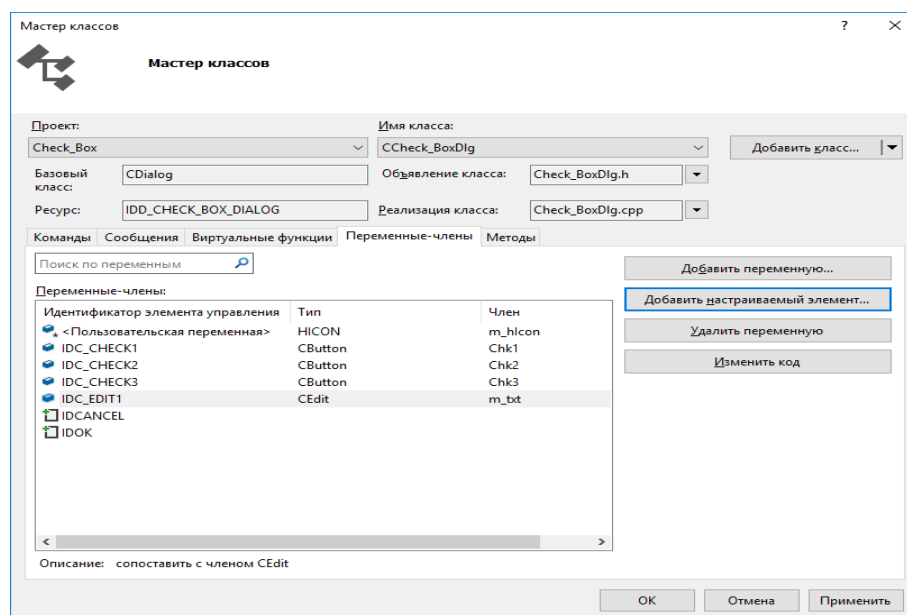


Додамо до діалогового вікна керуючий елемент «Edit Control» - натискаємо «Панель элементов» та розташовуємо елемент у поле діалогового вікна:

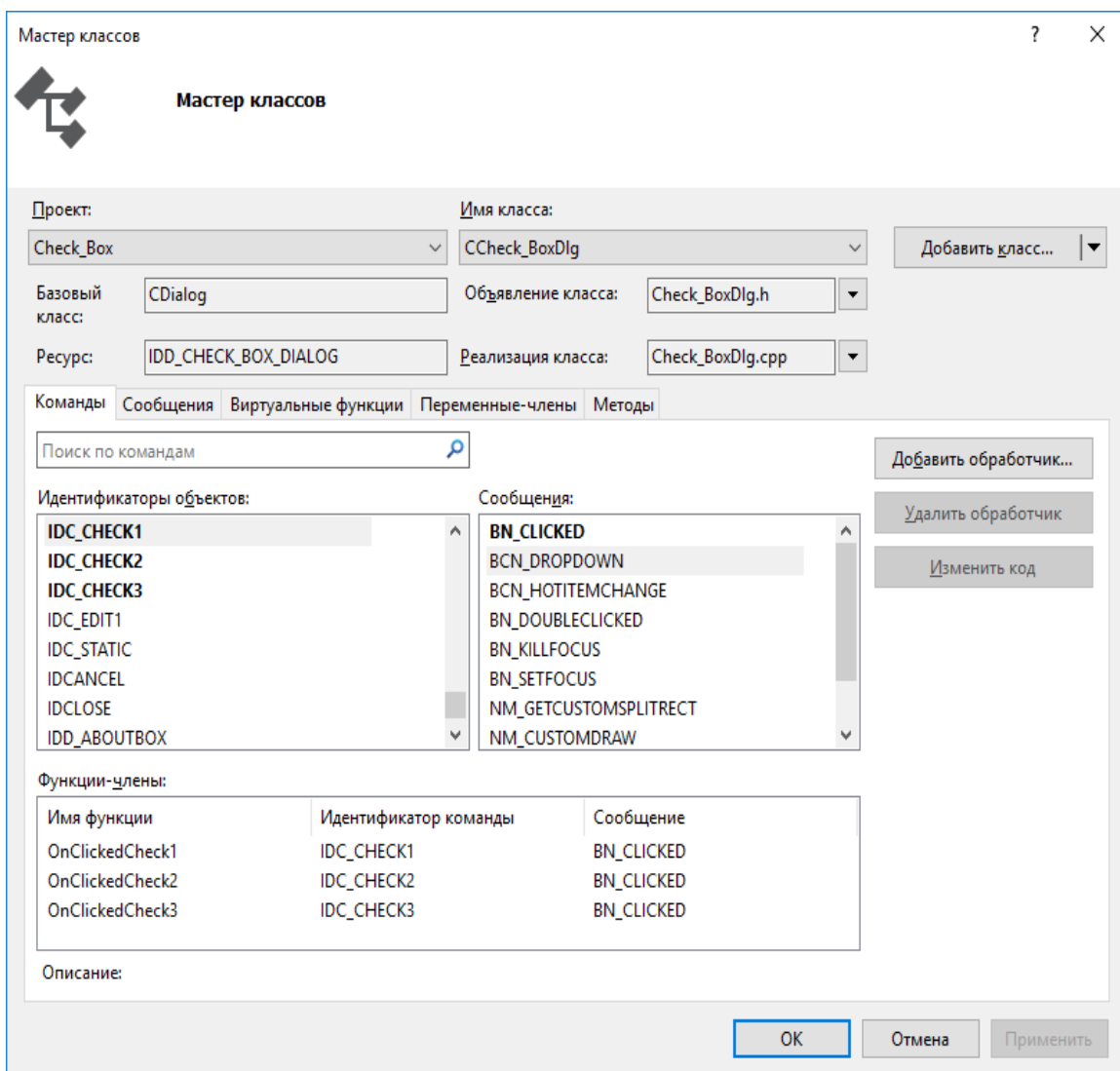
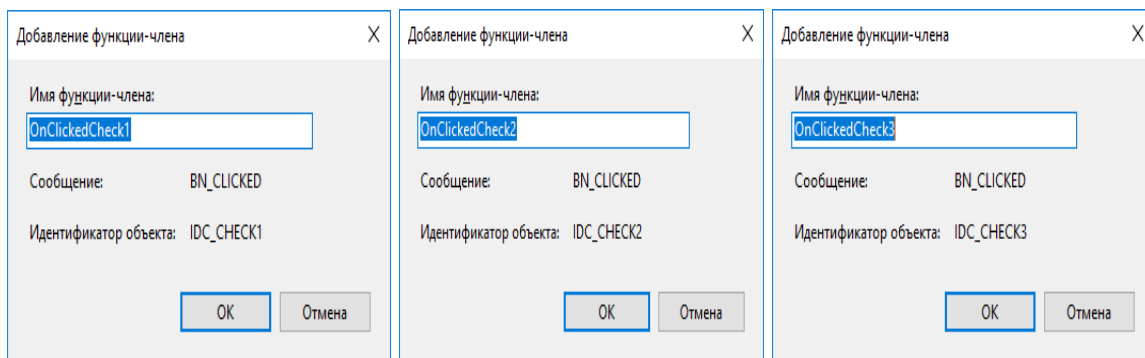


2.3 Работа з майстром класів

В майстрі класів створюємо об'єкти Chk1, Chk2 и Chk3 на клас CButton та об'єкт m_txt на клас CEdit.



У майстрі класів знаходимо команди IDC_CHECK1, IDC_CHECK2 і IDC_CHECK3 та створимо відповідні обробники OnClickedCheck1(), OnClickedCheck2() та OnClickedCheck3():



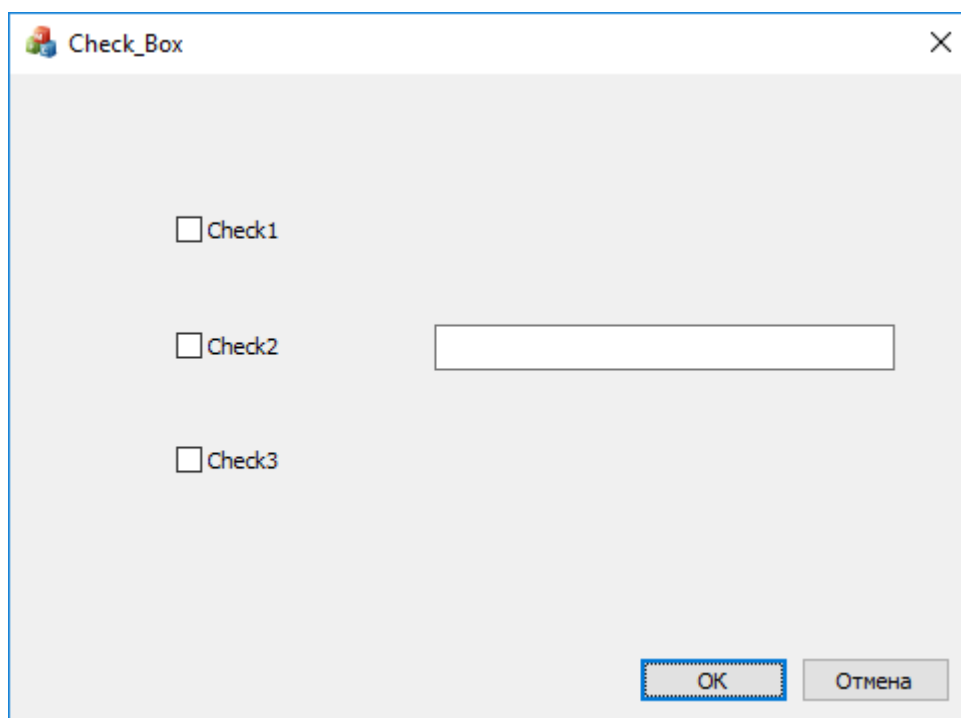
Пропишемо у відповідних обробниках наступний програмний код:

```
if (Chk1.GetCheck()==bool(1)) m_txt.SetWindowTextW(CString("Активен флажок 1"));  
    else m_text.SetWindowTextW(CString(" "));  
    UpdateData(false);
```

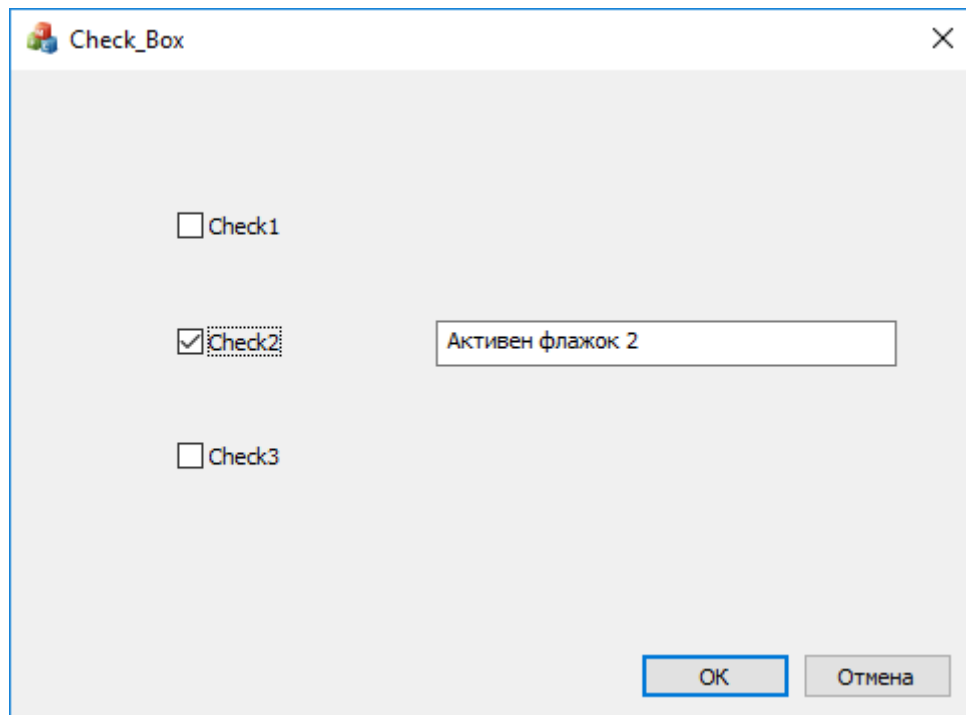
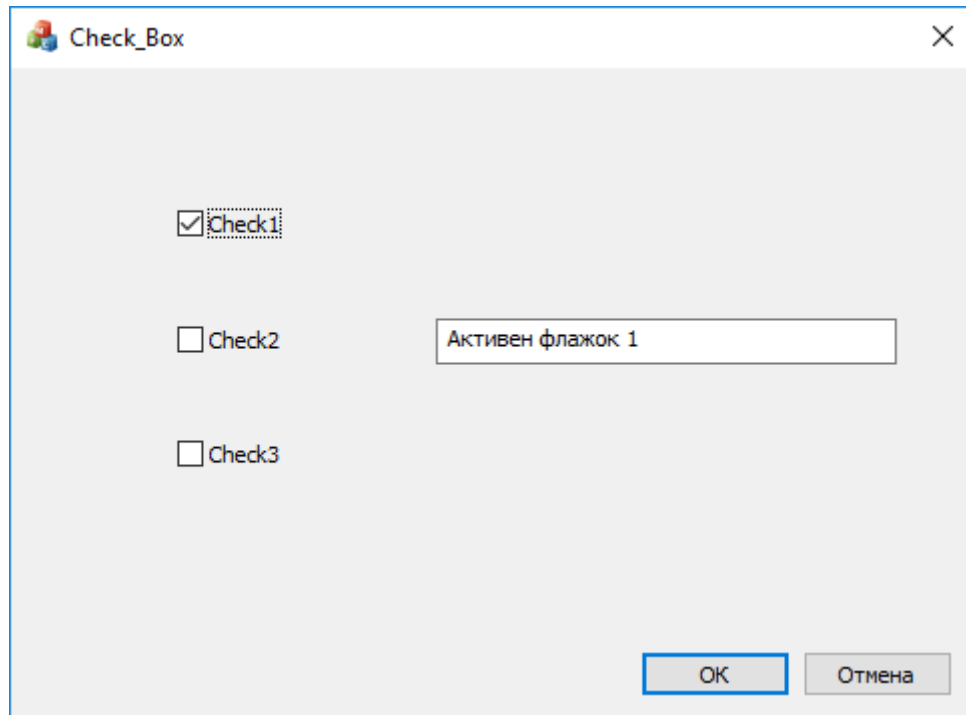
```
-----  
if (Chk2.GetCheck() == bool(1)) m_txt.SetWindowTextW(CString("Активен флажок 2"));  
    else m_text.SetWindowTextW(CString(" "));  
    UpdateData(false);
```

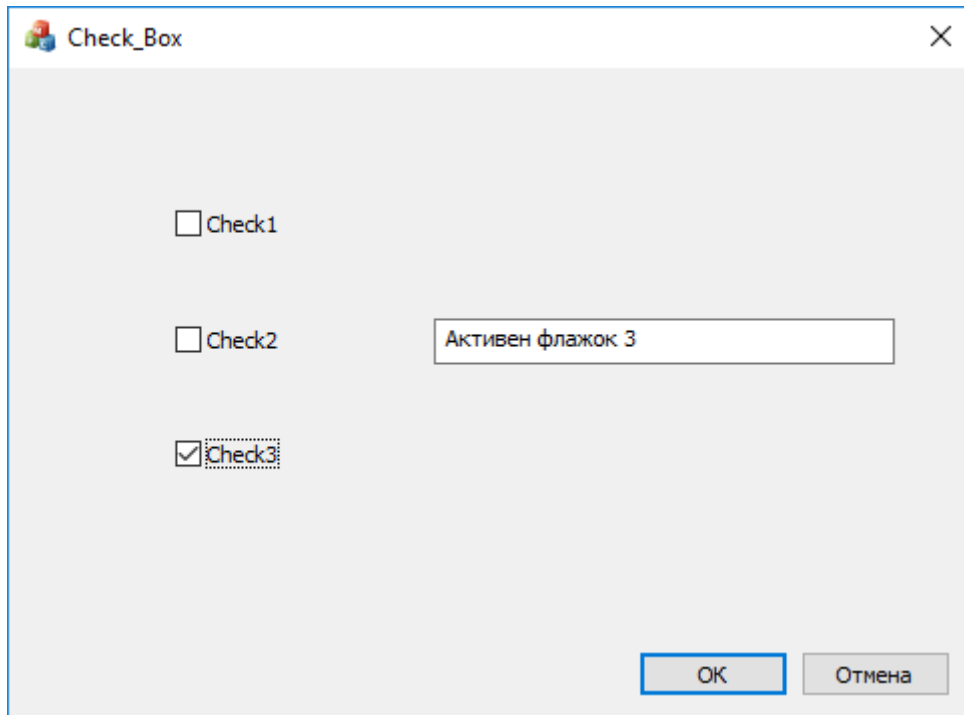
```
-----  
if (Chk3.GetCheck() == bool(1)) m_txt.SetWindowTextW(CString("Активен флажок 3"));  
    else m_text.SetWindowTextW(CString(" "));  
  
    UpdateData(false);
```

Windows-програма Check_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 за допомогою системної мишки відповідного прапорця CheckX, у тексті поле діалогового вікна буде виводитись інформація про активацію певного прапорця CheckX:





3 Порядок виконання роботи

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

3.2 Зібрати рішення Windows – проєкту.

3.3 Запустити зібраний Windows – проєкту Ctrl-F5.

3.4 Отриманням 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.