Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій

Кафедра мережевих та інтернет технологій

Лабораторна робота №6

З дисципліни “Системне програмування”

Студента Фуркала Олександра

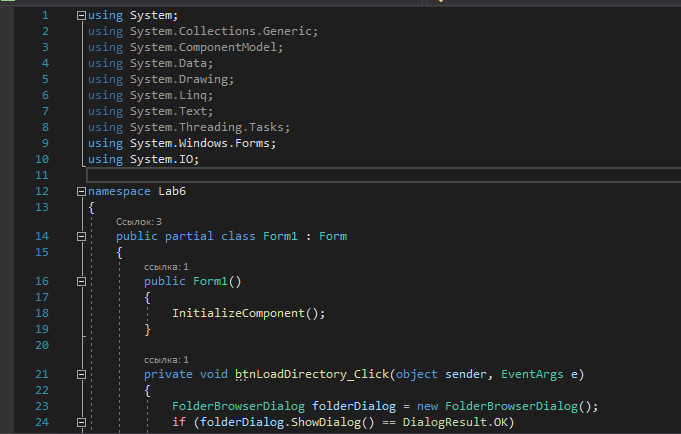
За темою:

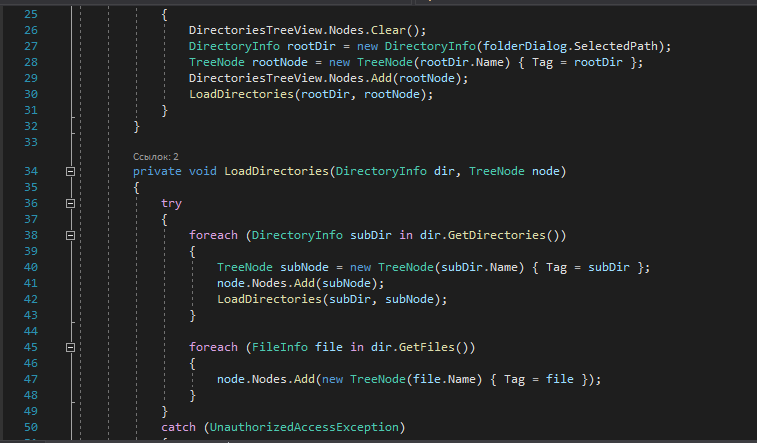
“Файлове введення-виведення”

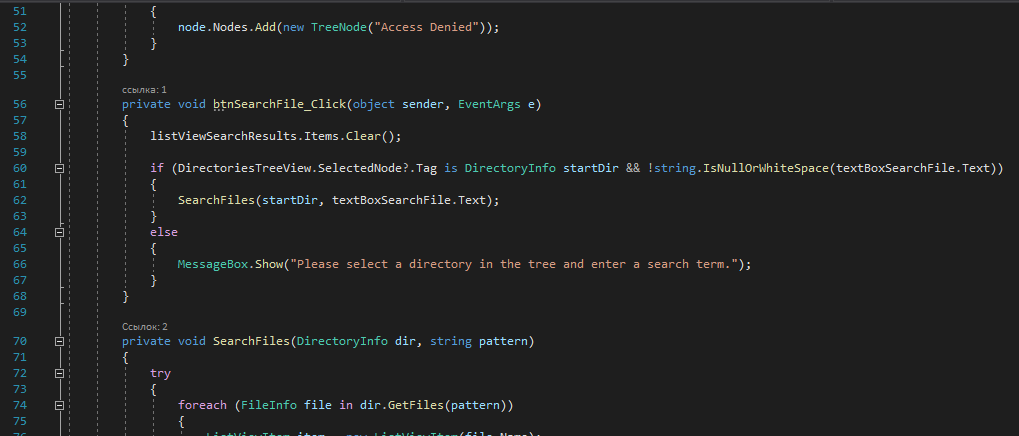
Варіант-3

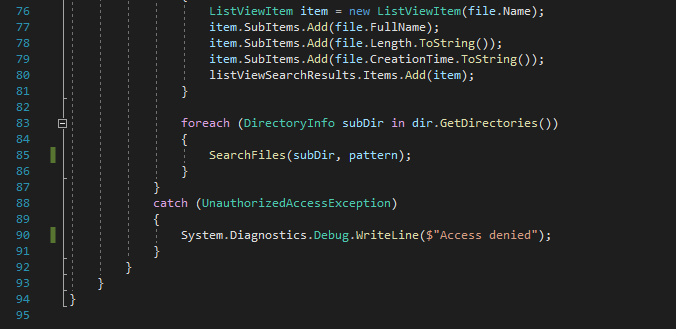
Хід роботи

1. Повторити задачі
2. Створити програму, яка виводить на екран усю ієрархічну структуру певної директорії (файли та папки).
3. Створити програму, що виконує пошук заданого файлу та виводить повну інформацію про місце розташування знайдених екземплярів.
4. Створити програму з графічним інтерфейсом, яка б відтворювала основні можливості файлового менеджера.

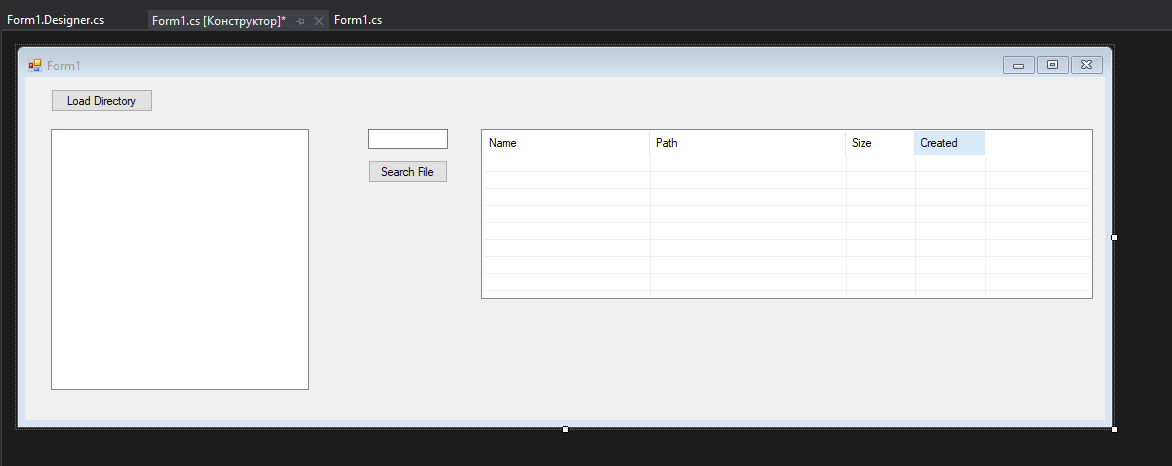


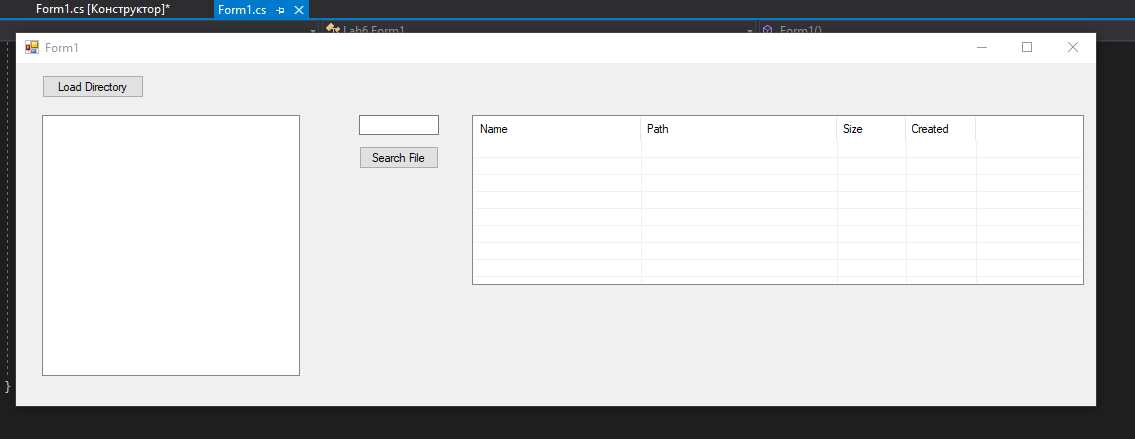


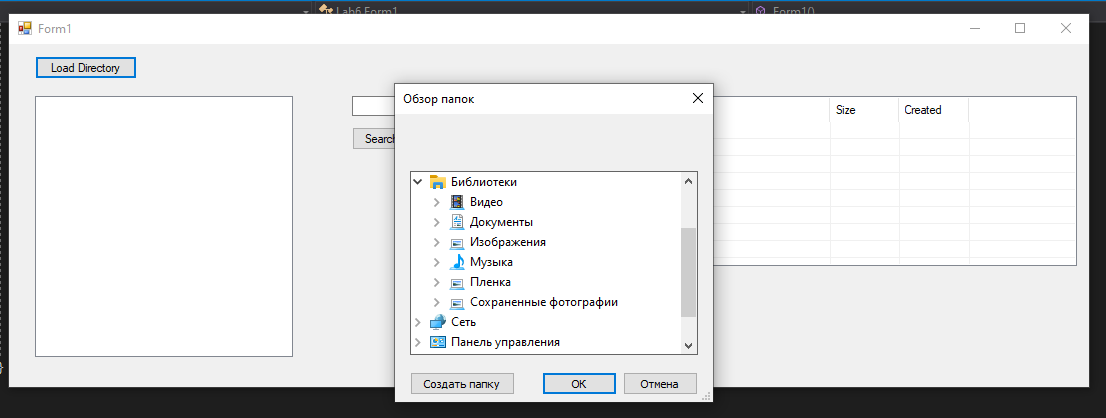


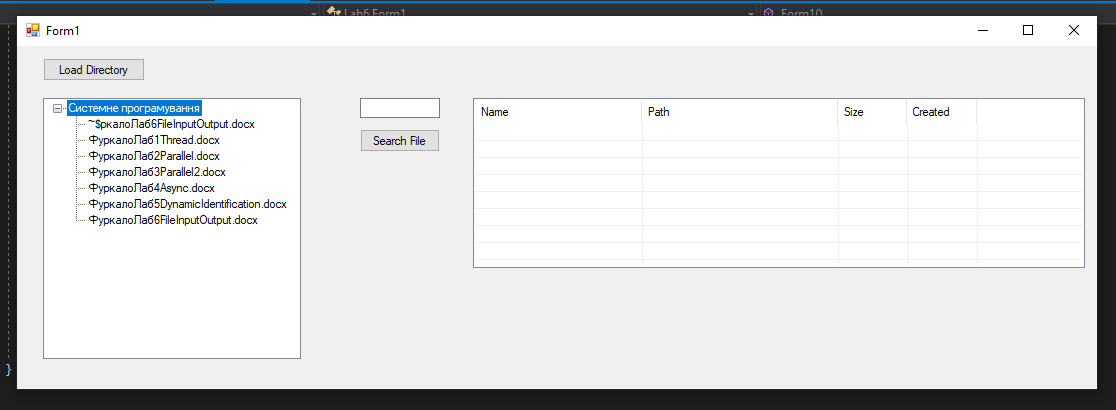


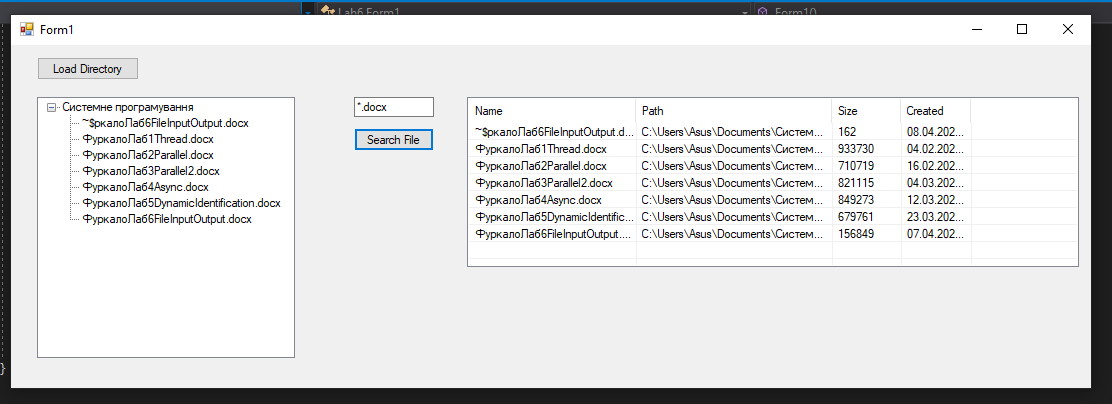
Форма

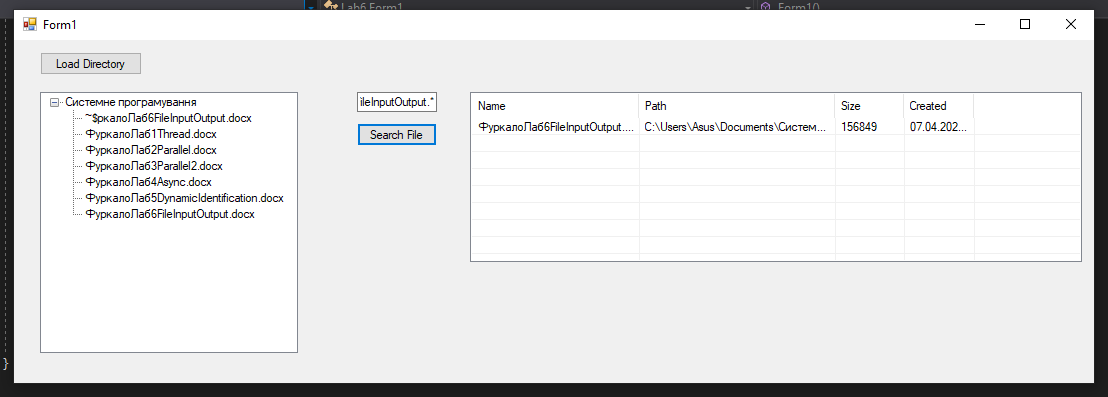


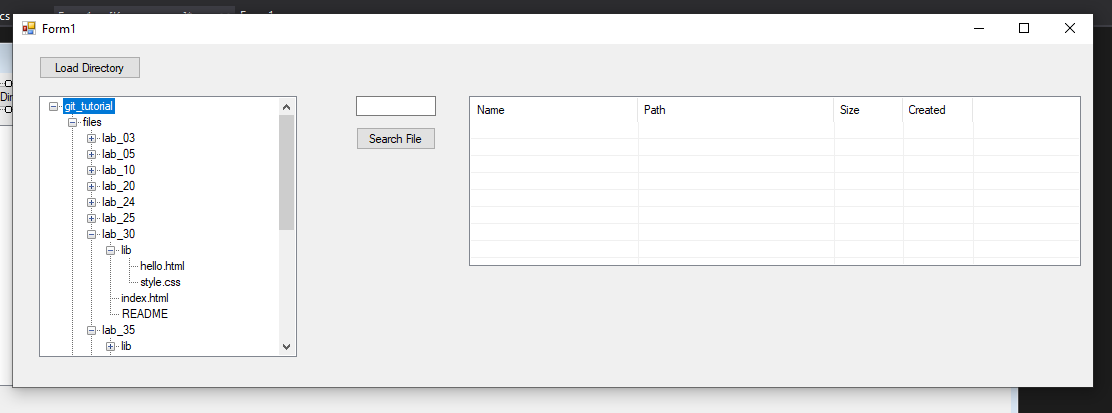


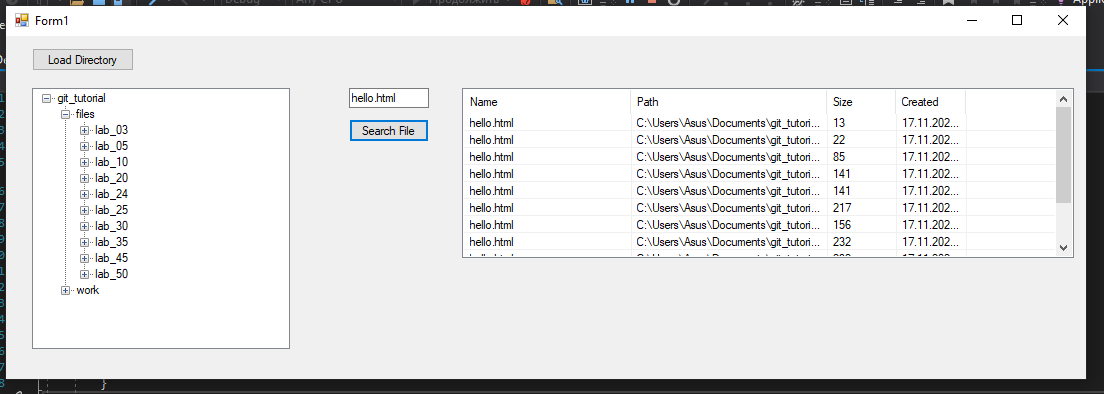












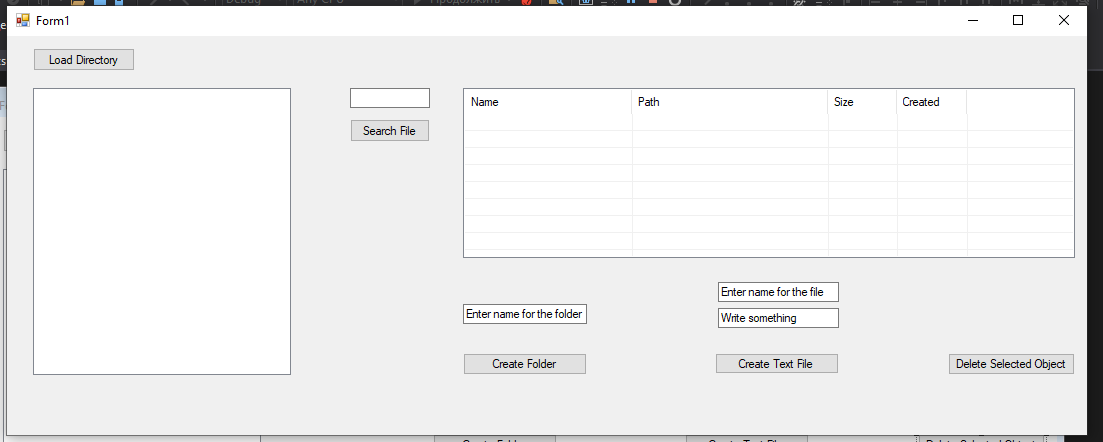
FolderBrowserDialog відкриває стандартне діалогове вікно вибору папки Windows, що дозволяє користувачеві вибрати кореневий каталог. Потім, через DialogResult ми перевіряємо, чи натиснув в цьому вікні користувач “ОК” – якщо так, то переходимо до завантаження структури каталогу.

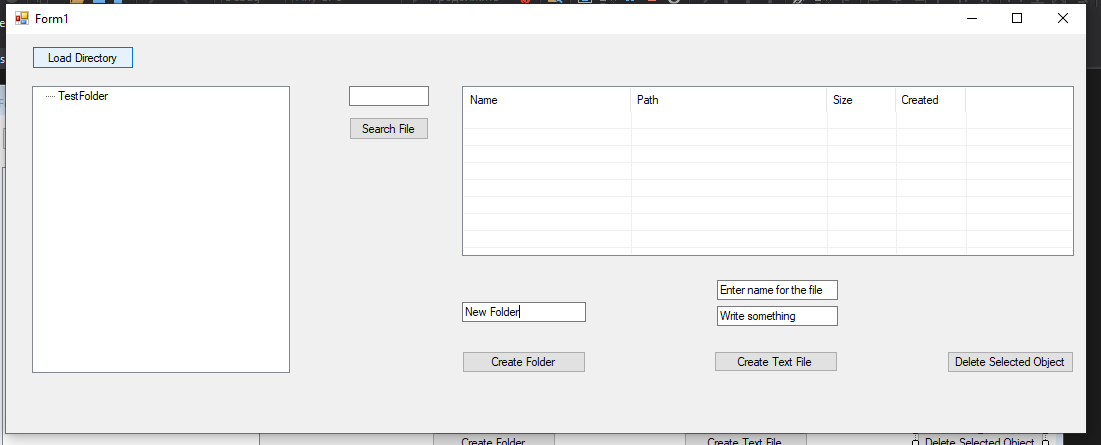
DirectoryInfo rootDir обгортає обраний шлях в об’єкт DirectoryInfo – це дозволяє легко отримати доступ до підкаталогів і файлів програмним шляхом (та ми можемо його використати для отримання корисних даних про каталог). Через TreeNode ми створюємо кореневий вузол дерева з назвою вибраної папки, та також зберігаємо повний об’єкт DirectoryInfo у тезі. Потім ми додаємо кореневий вузол до TreeView, та викликаємо наступний метод, LoadDirectories: нам буде потрібен dir DirectoryInfo, який є поточним каталогом, та node, TreeNode, до якого будуть додані дочірні папки та файли.

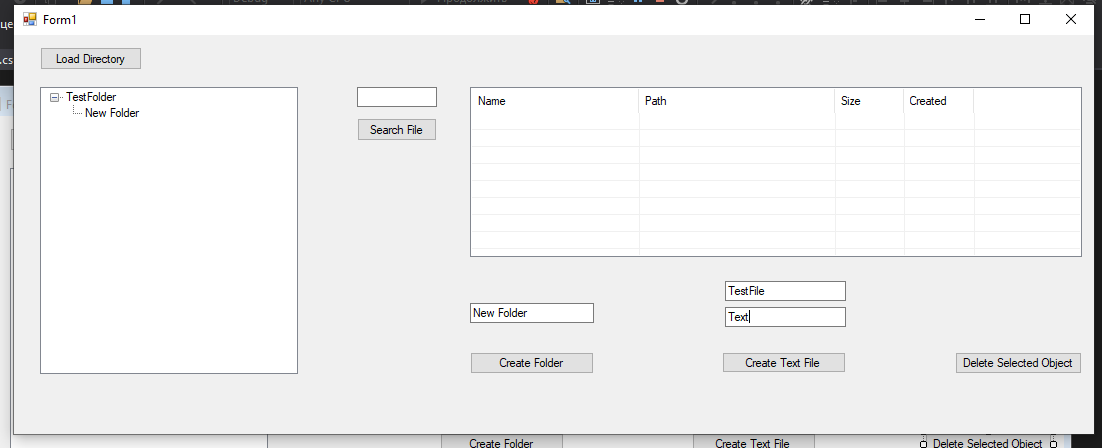
LoadDirectories try розподілен на два варіанти foreach, DirectoryInfo та FileInfo. Спочатку, ми перебираємо кожний підкаталог dir, створюючи для кожного новий TreeNode із назвою папки, потім додаючи цей новий вузол до батьківського, рекурсивно викликаючи себе. Потім, після додавання, проходимо іншим циклом по кожному файлу у поточному каталозі та додаємо TreeNode.

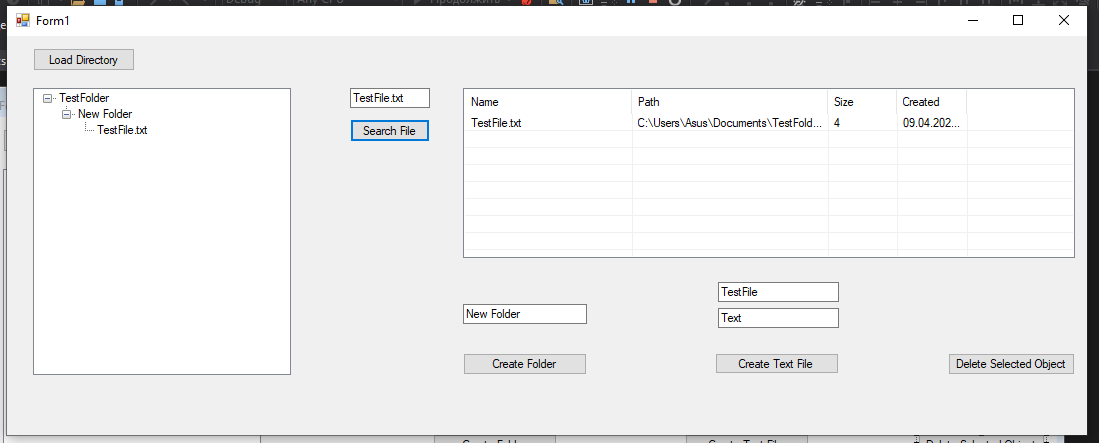
DirectoriesTreeView.SelectedNode?.Tag is DirectoryInfo перевіряє, чи вибрано вузол у TreeView, та чи є тег на цьому вузлі об’єктом DirectoryInfo (каталог) - ? використовується, щоб уникнути збою. Якщо так, призначаємо нову змінну startDir. !string.IsNullOrWhiteSpace перевіряє, що поле пошуку не пусте (null) та не лише пробіл (white space). Після успішної перевірки код починає пошук, використовуючи пошуковий термін.

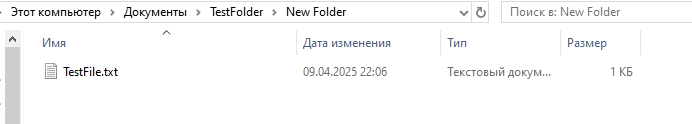
В SearchFiles, foreach FileInfo отримує всі файли в поточному каталозі, які відповідають шаблону пошуку, та для кожного з них створює ListViewItem - назва, шлях, розмір, дата створення. Цей елемент буде додано до ViewSearchResults для відображення. Для DirectoryInfo, метод буде викликано рекурсивно.

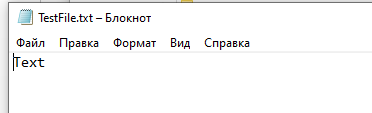


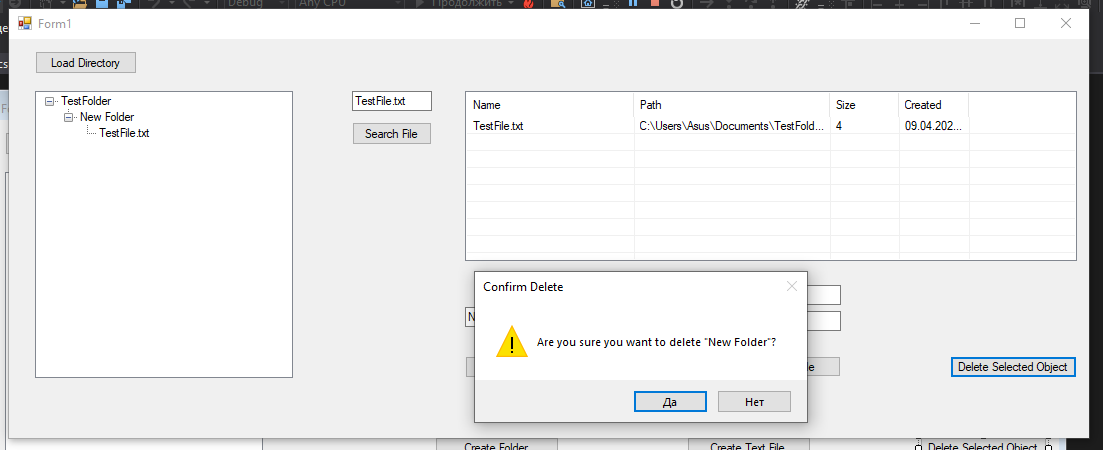


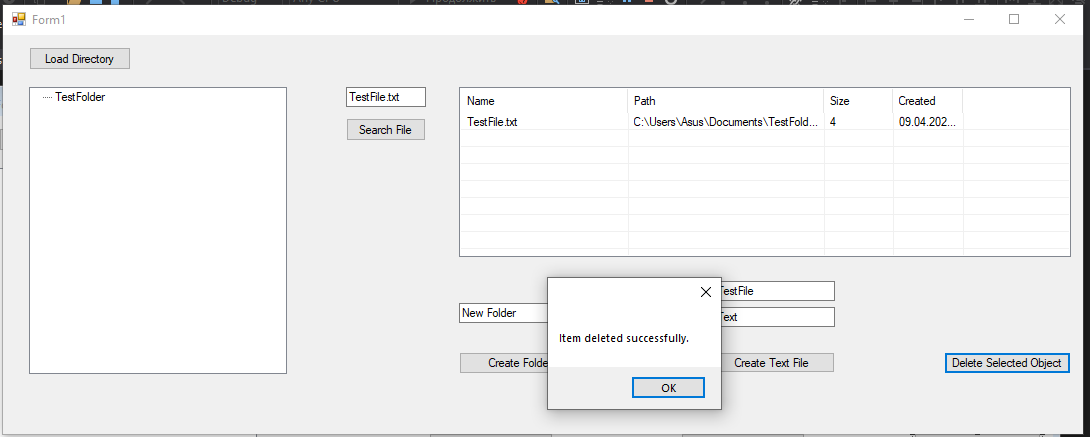


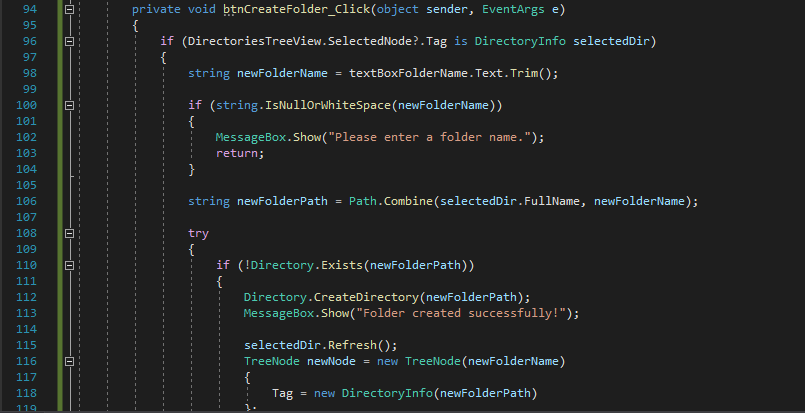


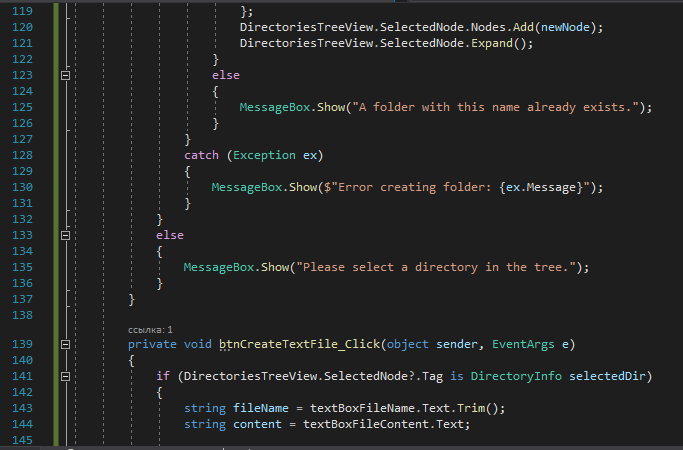




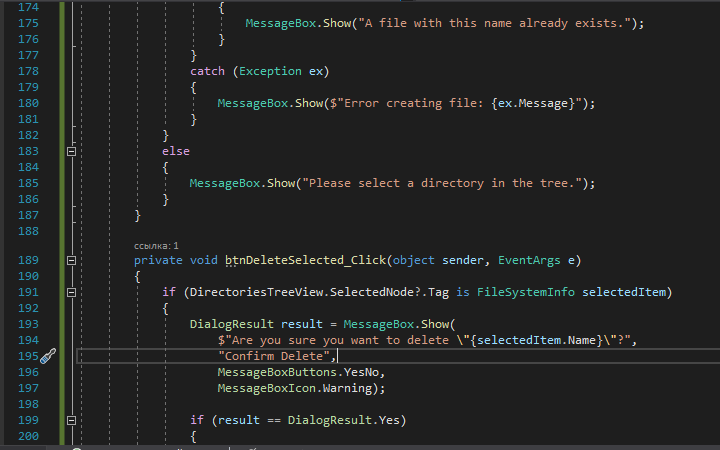


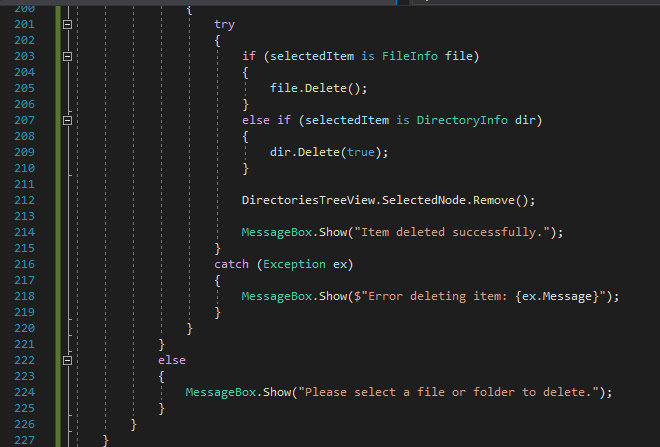












Text.trim видаляє всі початкові та кінцеві пробіли.

Path.Combine створює повний шлях до файлової системи шляхом об’єднання шляху до вибраної папки із назвою, яку користувач ввів у TextBox (назва звідти).

Створюємо новий каталог завдяки Directory.CreateDirectory, та потім оновлюємо представлення попереднього завантаженого каталогу: TreeNode створює новий вузол в інтерфейс користувача, який представляє щойно створену папку, та Nodes.Add додає нову папку як дочірній вузол у поточну обрану папку в TreeView. Створення нового файлу працює схожим чином.

В DeleteSelected, видаляємо через file.Delete – для папок потрібен окремий Delete(true), щоб також видалити підкаталоги. NodeRemove видаляє візуальний вузол із інтерфейсу користувача після видалення файлу чи папки.

1. Оформити звіт