

Лабораторна робота № 7

Файловий ввід-вивід

План

1. Класи, що реалізують файловий ввід-вивід.
- 2 Завдання.

1. Для різних маніпуляцій з каталогами та файлами у платформі .NET Framework реалізована бібліотека класів, описаних у просторі імен **System.IO**.

Для різних маніпуляцій з каталогами, наприклад, для **створення** (Create(), CreateSubdirectory()), **видалення** (Delete()), **переміщення** (MoveTo()) використовується клас DirectoryInfo. Крім того, він містить методи, що повертають **список підкаталогів** поточного каталогу (GetDirectories()) та **список файлів**, що містяться у вихідному каталозі (GetFiles()).

Наприклад, вивести на екран вміст заданого каталогу в стилі команди dir можна таким чином:

```
using System;
using System.IO;

namespace FileTest
{
    class Program
    {
        static void Main(string[] args)
        {
            // Підключення до заданого каталогу
            DirectoryInfo dir = new DirectoryInfo("D:/Work/Lazarus/test1");

            // Виведення списку вкладених підкаталогів та часу/дати їх створення
            Console.WriteLine("{0}\t<DIR>\t", dir.CreationTime);
            foreach (var it in dir.GetDirectories())
                Console.WriteLine("{0}\t<DIR>\t{1}", it.CreationTime, it.Name);

            // Виведення списку наявних файлів
            foreach (var it in dir.GetFiles())
                Console.WriteLine("{0}\t\t{1}", it.CreationTime, it.Name);
            Console.ReadKey();
        }
    }
}
```

Результат роботи цієї програми наведено на рис. 1.

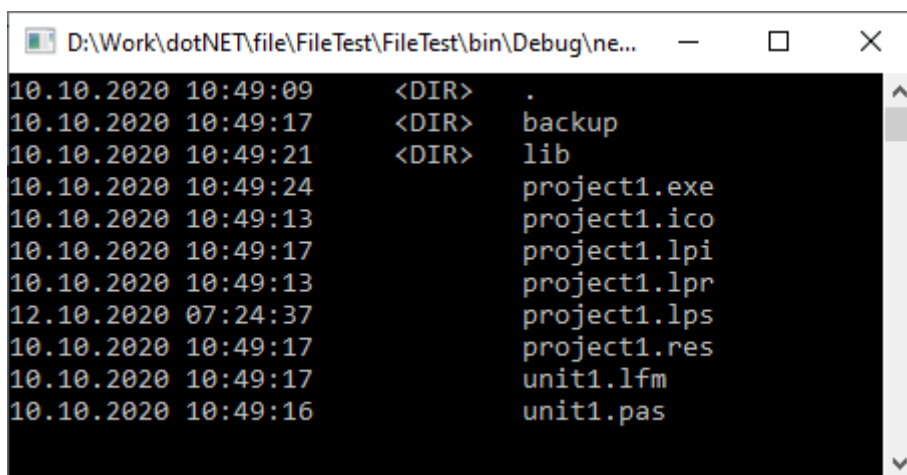


Рис. 1. Виведення вмісту каталогу

Для читання символічних файлів використовується клас **StreamReader**. Детальний його опис наведено в Лекції № 7. Нижче описано приклад програми, що здійснює виведення на екран назв файлів у заданій папці, які містять деякий текст:

```
using System;
using System.IO;

namespace FileTest
{
    class Program
    {
        static void Main(string[] args)
        {
            StreamReader sr; // Потік для читання файлу
            DirectoryInfo dir = new DirectoryInfo("D:/Work/Lazarus/test1");
            string str;
            bool is_find;

            // Обробка всіх файлів, що містяться в зазначеному каталозі
            foreach (var it in dir.GetFiles())
            {
                is_find = false;
                sr = new StreamReader(it.FullName); // Відкриття чергового файлу
                while (sr.Peek() != -1)
                {
                    str = sr.ReadLine(); // Зчитування наступного рядка
                    if (str.Contains("Form1")) // Пошук в ній контексту "Form1"
                    {
                        is_find = true;
                        break;
                    }
                }
            }
        }
    }
}
```

```

    }
}
// Виведення на екран імені файлу, що містить контекст.
if (is_find)
    Console.WriteLine(it.FullName);
sr.Close();
}
Console.ReadKey();
}
}
}

```

Результат роботи цієї програми наведено на рис. 2.

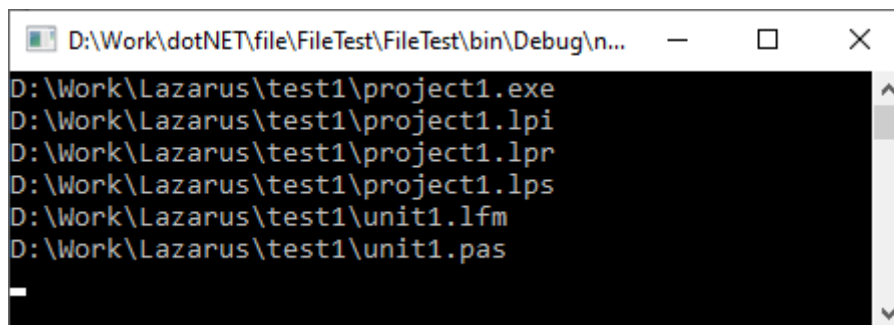


Рис. 2. Приклад роботи програми, що виводить на екран список файлів, які містять заданий текст

2. Використовуючи наведені в Лекції № 7 та даному документі приклади, написати програму, яка здійснює пошук у заданому каталозі всіх файлів, що містять вказаний користувачем текст. Передбачити передачу імені каталогу та тексту в програму у вигляді параметрів командного рядка.

В якості шуканого тексту задати ПІБ студента, що здає лабораторну роботу, а як шуканий каталог – папку, що містить всі попередні і поточну лабораторні роботи.

Передбачити виведення на екран ПІБ та номери групи студента, що здає роботу.