

Лабораторна робота № 11

Тема: Основи роботи з файлами.

Мета роботи: Навчитися створювати, читати, записувати та редагувати файли у мові програмування C#.

Методичні рекомендації

У C# для роботи з файлами використовуються простори імен:

- System.IO.File — статичні методи для створення, копіювання, видалення, переміщення файлів.
- System.IO.StreamReader та System.IO.StreamWriter — для читання та запису текстових файлів.
- System.IO.FileStream — для роботи з потоками байтів.

Мова C# має функції для роботи з файлами. Як і інші команди вводу-виводу вони розміщені в системному модулі System.IO, який треба підключити на початку програми командою using System.IO;.

Актуалізація опорних знань.

1. Які класи простору імен System.IO ви знаєте?
2. Охарактеризуйте Класи File та FileInfo
3. Які є методи для роботи з файлами?
4. Чим відрізняється StreamReader від File.ReadAllText()?
5. Яким чином можна оголосити і відкрити файл?
6. Як відкрити файл для дозапису?
7. Наведіть приклади запису і читання по символах, по словах, по рядках та блоками довільного розміру.

Зміст роботи.

Завдання 1. Створіть новий файл numbers.txt у текстовому редакторі і запишіть у нього 10 чисел, кожне з нового рядка. Напишіть програму, яка зчитує ці числа з файла і обчислює їх суму, виводить цю суму на екран і, водночас, записує цю суму у інший файл з назвою sum_numbers.txt.

Завдання 2. Реалізуйте програму, яка зчитує довільну кількість цілих чисел, що вводяться з командного рядка, і записує у текстовий файл інформацію, щодо парності або непарності чисел.

Завдання 3. Дано текстовий файл, який містить у єдиному рядку деяку кількість

дійсних чисел розділених пробілами. Знайти найбільше та найменше з чисел та вивести їх у двох рядках іншого текстового файлу.

Завдання 4. Прочитати рядок з текстового файлу. З перших літер кожного слова рядка скласти нове слово, яке записати до іншого файлу.

Завдання 5. Є текстовий файл. Переписати до іншого файлу всі рядки, обернуті справа наліво.

Завдання 6. З текстового файлу причитати речення, першу літеру кожного слова зробити великою. Отримане речення записати до іншого файлу.

Завдання 7. З текстового файлу причитати речення, після кожного слова поставити знак оклику. Крапку в кінці речення (якщо вона є) видалити. Отримане речення вивести на екран.

Завдання 8. Отримати інформацію про всі створені файли.

Приклад:

```
using System;
```

```
using System.IO;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        string path = "example.txt";
```

```
        // 1. Створення та запис у файл
```

```
        using (StreamWriter writer = new StreamWriter(path))
```

```
        {
```

```
            writer.WriteLine("Привіт, це перший рядок!");
```

```
            writer.WriteLine("А це другий рядок.");
```

```
        }
```

```
        Console.WriteLine("Файл створено та записано.");
```

```
        // 2. Читання з файлу
```

```
        using (StreamReader reader = new StreamReader(path))
```

```
        {
```

```
            string content = reader.ReadToEnd();
```

```
            Console.WriteLine("Вміст файлу:");
```

```
            Console.WriteLine(content);
```

```
        }
```

```
// 3. Додавання тексту у файл
using (StreamWriter writer = new StreamWriter(path, true))
{
    writer.WriteLine("Цей рядок додано пізніше.");
}

Console.WriteLine("Дані успішно додані.");
}
}
```