

# C# شرح التعامل مع الملفات في

---

## (StreamWriter) إنشاء ملف وكتابة محتوى فيه 1.

المكتبات المستخدمة:

- **System**: وظائف أساسية مثل الطباعة.
- **System.IO**: أدوات التعامل مع الملفات.

**StreamWriter** الشرح: ينشئ الملف تلقائيًا إذا لم يكن موجودًا، ويكتب داخله باستخدام

```
using System;           // وظائف أساسية (مثل Console)
```

```
using System.IO;        // للتعامل مع الملفات
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        string filePath = "myfile.txt";
```

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

```
        {
```

```
            writer.WriteLine("Hello, my name is Mohammed.");
```

```
            writer.WriteLine("This is my first file.");
```

```
        }
```

```
        Console.WriteLine("File has been created and written successfully.");  
    }  
}
```

---

## 2. إضافة محتوى إلى نهاية الملف (Append)

المكتبات المستخدمة:

- `System`
- `System.IO`

الشرح: يضيف سطرًا جديدًا إلى نهاية الملف بدون حذف الأسطر السابقة.

```
using System;
```

```
using System.IO;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        string filePath = "myfile.txt";
```

```
        using (StreamWriter writer = new StreamWriter(filePath, true)) // true = append
```

```
        {
```

```
            writer.WriteLine("This line was added later.");
```

```
        }
```

```
        Console.WriteLine("Line appended to file.");
```

```
}  
  
}
```

---

### 3. إدراج سطر داخل الملف (في السطر الثاني مثلاً)

المكتبات المستخدمة:

- `System`
- `System.IO`
- `System.Collections.Generic`: للتحكم بالقوائم (`List`)

ثم نضيف سطر جديد في الموضع الذي نريد، ثم نكتب القائمة كاملة من `List` الشرح: نقرأ كل الأسطر من الملف إلى قائمة جديد إلى الملف.

```
using System;
```

```
using System.IO;
```

```
using System.Collections.Generic;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        string filePath = "myfile.txt";
```

```
        List<string> lines = new List<string>(File.ReadAllLines(filePath));
```

```
        lines.Insert(1, "This line was inserted by Mohammed."); // السطر الثاني (المؤشر 1)
```

```
File.WriteAllLines(filePath, lines);

Console.WriteLine("Line inserted successfully.");
}
}
```

---

## 4. استخراج معلومات عن الملف (FileInfo)

المكتبات المستخدمة:

- `System`
- `System.IO`

الشرح: يعرض معلومات مثل الاسم، المسار، الحجم، تاريخ الإنشاء، آخر تعديل، وهل الملف للقراءة فقط أم لا

```
using System;
```

```
using System.IO;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        string filePath = "myfile.txt";
```

```
        FileInfo fileInfo = new FileInfo(filePath);
```

```
        Console.WriteLine("File Name: " + fileInfo.Name);
```

```
        Console.WriteLine("Full Path: " + fileInfo.FullName);
```

```
        Console.WriteLine("Extension: " + fileInfo.Extension);
```

```
Console.WriteLine("Size (bytes): " + fileInfo.Length);

Console.WriteLine("Created On: " + fileInfo.CreationTime);

Console.WriteLine("Last Modified: " + fileInfo.LastWriteTime);

Console.WriteLine("Is Read Only? " + fileInfo.IsReadOnly);

}

}
```

---

### شرح تفصيلي لبعض الأوامر:

**StreamWriter writer = new StreamWriter(filePath);**

- **StreamWriter**: كلاس مخصص لكتابة النصوص في ملفات.
- **writer**: (هو اسم المتغير الذي نستخدمه للكتابة داخل الملف (يمكن تغييره).
- **= new**: **StreamWriter** تعني إنشاء كائن جديد من الكلاس.
- **(filePath)**: هو مسار الملف الذي سيتم الكتابة فيه.

**FileInfo fileInfo = new FileInfo(filePath);**

- **FileInfo**: كلاس يُستخدم للحصول على معلومات عن ملف مثل الحجم، الاسم، الوقت، إلخ.
  - **fileInfo**: اسم المتغير الذي يمثل الكائن.
  - **= new**: **FileInfo** إنشاء كائن جديد من نوع.
  - **(filePath)**: المسار الذي يشير إلى الملف الذي نريد معرفة معلوماته.
-