

Programming in C#

Mục tiêu:

Sau nội dung thực hành này bạn có khả năng:

Làm việc với các Streams và Files

Phần I: Thực hành từng bước – 45 phút

Bài tập 1: Use BinaryReader to read file in binary format

```
using System;
using System.IO;

class MainClass{
    public static void Main(){
        FileStream outStream = File.Create("c:\\BinaryTest.dat");

        BinaryWriter bw = new BinaryWriter(outStream);

        bw.Write( (int) 3);
        bw.Write( (decimal)
4.5); string s = "Test
String"; bw.Write(s);

        bw.Flush();
        bw.Close();

        FileStream inStream = File.OpenRead("c:\\BinaryTest.dat");

        BinaryReader br = new BinaryReader(inStream);

        int i = br.ReadInt32();
        decimal d = br.ReadDecimal();
        string s2 = br.ReadString();

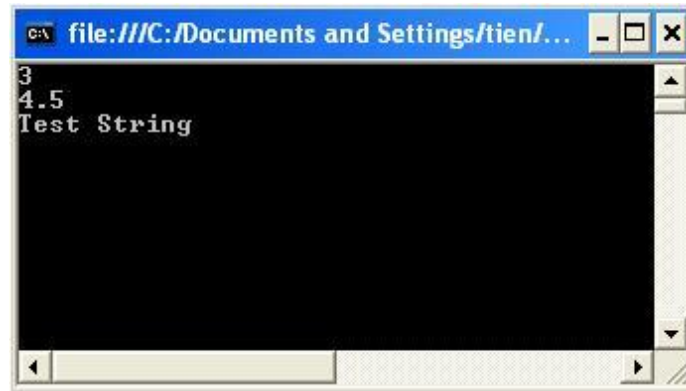
        Console.WriteLine(i);
        Console.WriteLine(d);
        Console.WriteLine(s2);
    }
}
```

```

        br.Close();
    }
}

```

Result:



Bài tập 2: The FileStream class is used to create a file output.txt in C: drive

```

using System;

using System.IO;

class MainClass{
    public static void Main(){
        FileStream f = new FileStream("c:\\output.txt",
        FileMode.Create); StreamWriter s = new StreamWriter(f);

        s.WriteLine("{0} {1}", "test", 55);
        s.Close();
        f.Close();
    }
}

```

Bài tập 3: Open a file using StreamWriter

```

using System;
using System.IO;

class MainClass{
    public static void Main(){

        StreamWriter fstr_out;
        try{
            fstr_out = new StreamWriter(@"c:\test.txt");
        }
    }
}

```

Streams and Files

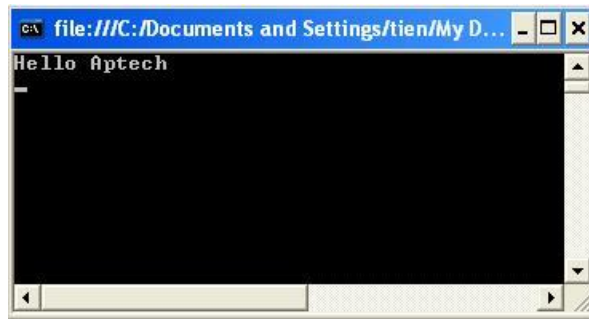
```
        catch (IOException exc) {
            Console.WriteLine(exc.Message + "Cannot open file.");
            return;
        }
        try {
            fstr_out.Write("Hello Aptech");
        }
        catch (IOException exc) {
            Console.WriteLine(exc.Message + "File Error");
            return;
        }
        fstr_out.Close();
    }
}
```

Bài tập 4: A FileReader

```
using System;
using System.IO;

class MainClass{
    public static void Main() {
        FileStream fin;
        string s;
        try {
            fin = new FileStream(@"C:\test.txt", FileMode.Open);
        }
        catch (FileNotFoundException exc) {
            Console.WriteLine(exc.Message + "Cannot open file.");
            return;
        }
        StreamReader fstr_in = new StreamReader(fin);
        // Read the file line-by-line.
        while ((s = fstr_in.ReadLine()) != null) {
            Console.WriteLine(s);
        }
        fstr_in.Close();
        Console.ReadLine();
    }
}
```

The result:



Bài tập 5: Get DirectoryInfo from FileInfo: name, directory exists

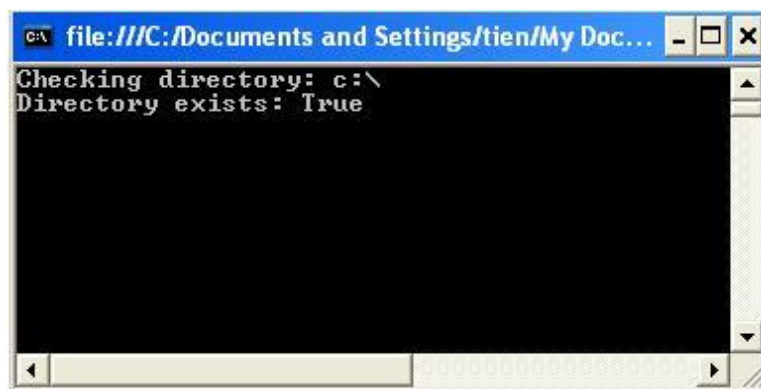
```
using System;
using System.IO;

class MainClass{
    public static void Main(string[] args){
        FileInfo file = new FileInfo("c:\\test.txt");

        // Display directory information.
        DirectoryInfo dir = file.Directory;

        Console.WriteLine("Checking directory: " + dir.Name);
        Console.Write("Directory exists: ");
        Console.WriteLine(dir.Exists.ToString());
    }
}
```

The result:



Bài tập 6: DirectoryInfo: Directory attribute list and Directory contains

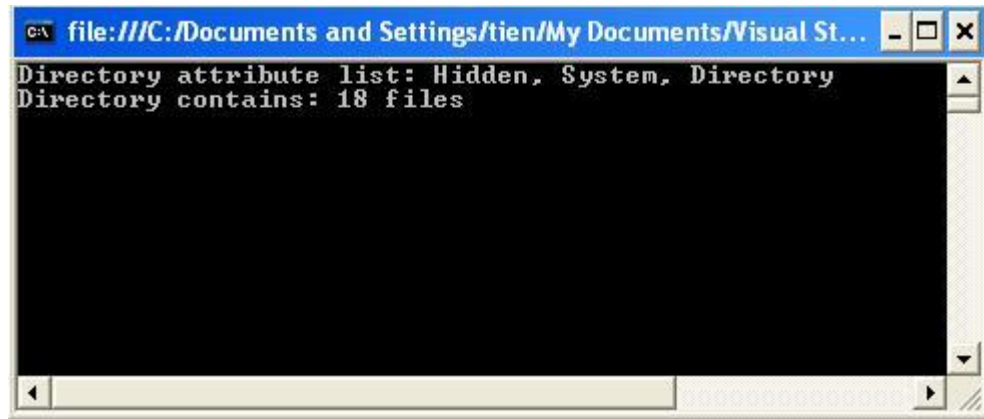
```
using System;
using System.IO;

class MainClass{
    public static void Main(string[] args)
    {
        FileInfo file = new FileInfo("c:\\test.txt");

        // Display directory information.
        DirectoryInfo dir = file.Directory;

        if (dir.Exists){
            Console.Write("Directory attribute list: ");
            Console.WriteLine(dir.Attributes.ToString());
            Console.WriteLine("Directory contains: "
                + dir.GetFiles().Length.ToString() + " files");
        }
    }
}
```

The result:



Bài tập 7. FileStream class

```
using System;
using System.IO;
using System.Text;

class Test{

    public static void Main(){
        string path = @"c:\temp\MyTest.txt";

        // Delete the file if it exists.
        if (File.Exists(path)){
            File.Delete(path);
        }

        //Create the file.
        using (FileStream fs = File.Create(path)){
            AddText(fs, "This is some text");
            AddText(fs, "This is some more text,");
            AddText(fs, "\r\nand this is on a new line");
            AddText(fs, "\r\n\r\nThe following is a subset of
                        characters:\r\n");

            for (int i = 1; i < 120; i++){
                AddText(fs, Convert.ToChar(i).ToString());

                //Split the output at every 10th character.
                if (Math.IEEERemainder(Convert.ToDouble(i), 10) == 0){
                    AddText(fs, "\r\n");
                }
            }
        }

        //Open the stream and read it back.
        using (FileStream fs = File.OpenRead(path)){
            byte[] b = new byte[1024];
            UTF8Encoding temp = new UTF8Encoding(true);
            while (fs.Read(b, 0, b.Length) > 0){
                Console.WriteLine(temp.GetString(b));
            }
        }

        private static void AddText(FileStream fs, string value){
            byte[] info = new UTF8Encoding(true).GetBytes(value);
            fs.Write(info, 0, info.Length);
        }
    }
}
```

Bài tập 8: Directory class

```
using System;
using System.IO;

class Test{
    public static void Main(){
        // Specify the directories you want to manipulate.
        string path = @"c:\MyDir";
        string target = @"c:\TestDir";
        try{
            // Determine whether the directory exists.
            if (!Directory.Exists(path)){
                // Create the directory it does not exist.
                Directory.CreateDirectory(path);
            }
            if (Directory.Exists(target)){
                // Delete the target to ensure it is not
                // there. Directory.Delete(target, true);
            }

            // Move the directory.
            Directory.Move(path, target);

            // Create a file in the directory.
            File.CreateText(target + @"\myfile.txt");

            // Count the files in the target directory.
            Console.WriteLine("The number of files in {0} is {1}",
                target, Directory.GetFiles(target).Length);
        }
        catch (Exception e){
            Console.WriteLine("The process failed: {0}", e.ToString());
        }
        finally {}
    }
}
```

Bài tập 9: File class

```
using System;
using System.IO;

class Test{
    public static void Main(){
        string path = @"c:\temp\MyTest.txt";
        if (!File.Exists(path)){
            // Create a file to write to.
            using (StreamWriter sw = File.CreateText(path)){
                sw.WriteLine("Hello");
                sw.WriteLine("And");
                sw.WriteLine("Welcome")
                ;
            }
        }

        // Open the file to read from.
        using (StreamReader sr = File.OpenText(path)){
            string s = "";

            while ((s = sr.ReadLine()) != null){
                Console.WriteLine(s);
            }
        }
        try{
            string path2 = path + "temp";
            // Ensure that the target does not exist.
            File.Delete(path2);

            // Copy the file.
            File.Copy(path, path2);
            Console.WriteLine("{0} was copied to {1}.", path, path2);

            // Delete the newly created file.
            File.Delete(path2);
            Console.WriteLine("{0} was successfully deleted.", path2);
        }
        catch (Exception e){
            Console.WriteLine("The process failed: {0}", e.ToString());
        }
    }
}
```


Phần II: Tự thực hành – 60 phút

Bài tập 2.1. Đọc và lưu thông tin sinh viên vào file dữ liệu Text

Tạo một lớp **Student** gồm: **name**, **age**, **mark** và các phương thức cần thiết.

Viết một chương trình có menu sau:

Menu

1. Save to File
2. Read File
3. Exit

Your choice: _

- **Save to File:** Nhập thông tin của một sinh viên và viết thông tin đó vào một file có tên là sv.txt – mỗi thông tin của 1 sinh viên lưu vào một dòng. Sử dụng Tab để phân biệt mỗi trường của đối tượng. Các đối tượng sinh viên lưu trữ trong một mảng.
- **Read File:** Đọc và hiển thị thông tin của các sinh viên

```
Console.WriteLine(s);
```

Bài tập 2.2. Đọc file ảnh đưa ra màn hình

Xây dựng chương trình đọc file ảnh đuôi jpg, bmp và in ra màn hình.

Chương trình cho phép:

- Nhập được dẫn địa chỉ file từ màn hình
- In ra thông tin từ ảnh dưới dạng console
- Chương trình tự động đọc được các loại ảnh có đuôi khác nhau là jpg và bmp

Bài tập 2.3. Tái cấu trúc lại các Project thành Solution

Tạo một solution mà mỗi project là một tuần

Mỗi bài tập là 1 hoặc nhiều file cs

Xây dựng chương trình có menu cho phép chọn tuần và các bài tập trong tuần để thực hiện

Sau khi chạy một bài tập có thể chọn lại menu để chạy các bài tập khác.