## Assignments to be done in this session

1. Write a console application which will read text files from mentioned file system location. And list subdirectories from mentioned folder on file system using System.IO namespace and use DirectoryInfo, Directory, File and FileInfo Classes with all the methods present in these classes.

using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");

Console.WriteLine("Reading Text File");

string path = @"D:\DemoFileAssignment.txt";

string FileData = File.ReadAllText(path);

Console.WriteLine(FileData);

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");

Console.WriteLine("Listing subdirectories \n");

ListSubDirectories(@"C:\Users\");

}

static void ListSubDirectories(string subdirectories)

{

string[] directories = Directory.GetDirectories(subdirectories);

foreach(string subdir in directories)

{

Console.WriteLine(subdir);

}

}

}

}



Directory and subdirectory is created..

using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("Directory and its properties");

string path = @"D:\My Directory Assignment";

DirectoryInfo directory = new DirectoryInfo(path);

directory.Create();

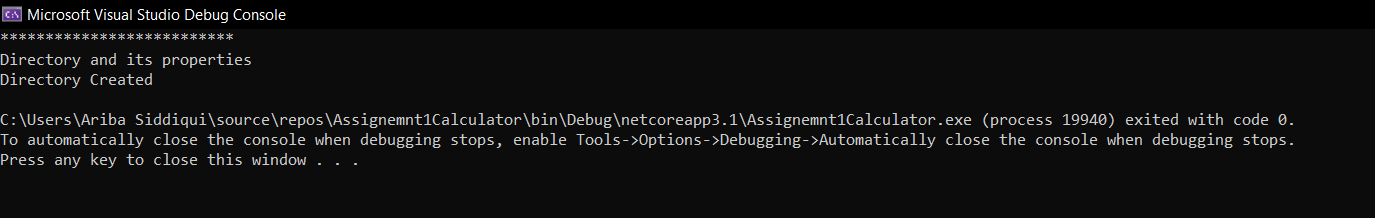
directory.CreateSubdirectory("Directory 1");

Console.WriteLine("Directory Created");

}

}

}



Movedto

using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("Directory and its properties");

string path = @"D:\My Directory Assignment";

string path2 = @"D:\My Directory Assignment11";

DirectoryInfo directory = new DirectoryInfo(path);

// directory.Create();

// directory.CreateSubdirectory("Directory 1");

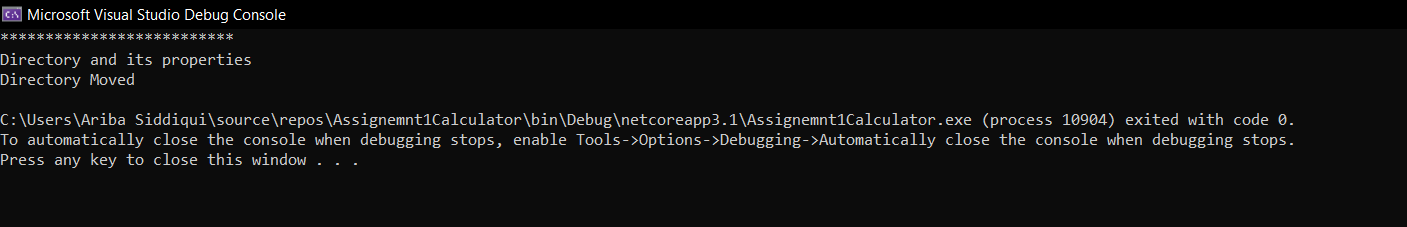
directory.MoveTo(path2); //Moveto is use to move directory to new directory and delete the previous directory

Console.WriteLine("Directory Moved");

}

}

}



Delete Directory

using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("Directory and its properties");

string path2 = @"D:\My Directory Assignment11";

DirectoryInfo directory = new DirectoryInfo(path2);

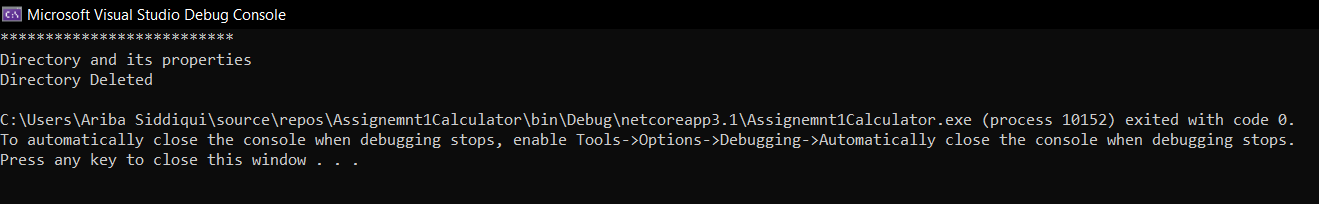
directory.Delete(true); // for deleting directory

Console.WriteLine("Directory Deleted");

}

}

}



using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("Directory and its properties");

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

string path = @"D:\DemoFileAssignment.txt";

//string path = @"D:\My Directory Assignment";

string path2 = @"D:\C#FileHandlingAssignment";

DirectoryInfo directory = new DirectoryInfo(path2);

Console.WriteLine("Directory Name :" + directory.Name);

Console.WriteLine("Directory FullName:" + directory.FullName);

Console.WriteLine("Directory Last access time:" + directory.LastAccessTime);

Console.WriteLine("Directory Creation Time: " + directory.CreationTime);

Console.WriteLine("Directory Attribute:" + directory.Attributes);

Console.WriteLine("Directory last edit:" + directory.LastWriteTime);

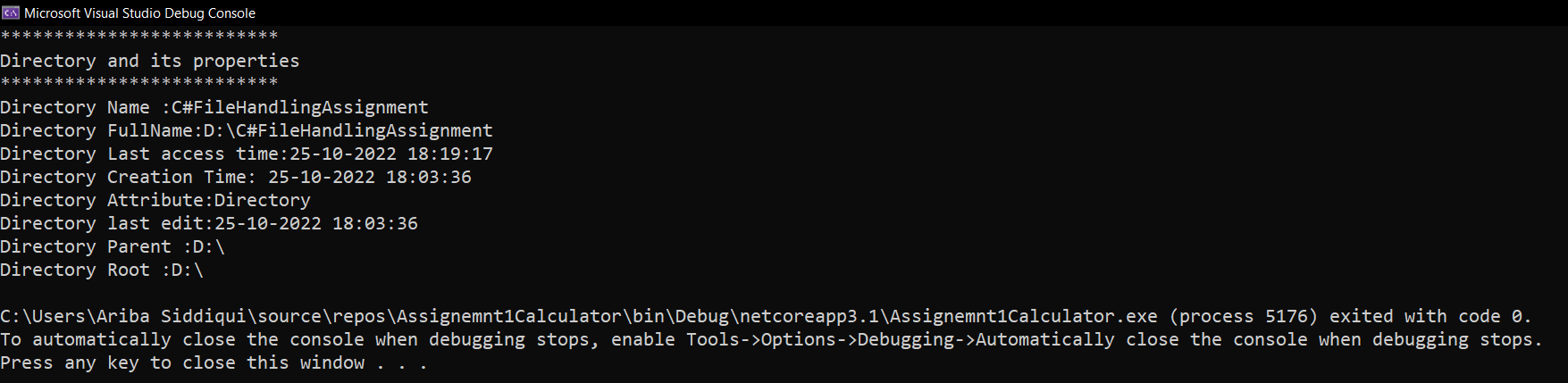
Console.WriteLine("Directory Parent :" + directory.Parent);

Console.WriteLine("Directory Root :" + directory.Root);

}

}

}



**FileInfo Properties**

using System;

using System.IO;

namespace FileIOAssignment

{

class ReadTxtFile

{

public static void Main(string[] args)

{

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("File and its properties");

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

string path = @"D:\DemoFileAssignment.txt";

FileInfo file = new FileInfo(path);

Console.WriteLine("File Name:" + file.Name);

Console.WriteLine("File Full Name:" + file.FullName);

Console.WriteLine("File Extension:" + file.Extension);

Console.WriteLine("File DirectoryName:" + file.DirectoryName);

Console.WriteLine("File Attributes:" + file.Attributes);

Console.WriteLine("File Creation Time:" + file.CreationTime);

Console.WriteLine("File Directory:" + file.Directory);

Console.WriteLine("File IsReadOnly or not:" + file.IsReadOnly);

Console.WriteLine("File LastAccessTime:" + file.LastAccessTime);

Console.WriteLine("File Length:" + file.Length);

}

}

}



Create a simple user interface to accept account related information of a customer.[ Account class from Lab session on Delegates and Events can be used]. Save the information about the customers in a file using StreamWriter and retrieve the information using StreamReader.

**INPUT:**

**Stream Writer Example**

using System;

using System.IO;

namespace FileHandling

{

#region Stream Writer Class

class StreamWriterExample

{

public static int AccountNumber;

public static string CustomerName;

public static int CustomerMobile;

public static string CustomerAddress;

static void Main()

{

#region Taking input

try

{

Console.WriteLine("Enter Customer Details....");

Console.WriteLine("-----------------------------");

Console.Write("Enter Account Number:");

AccountNumber = int.Parse(Console.ReadLine());

Console.Write("Enter Customer Name:");

CustomerName = Console.ReadLine();

Console.Write("Enter Customer Mobile Number:");

CustomerMobile = int.Parse(Console.ReadLine());

Console.Write("Enter Customer Address:");

CustomerAddress = Console.ReadLine();

}

catch(Exception ex)

{

Console.WriteLine(ex.Message);

}

#endregion

#region File Operation

try

{

string path = @"D:/DemoFileAssignment.txt";

FileStream fileStream = new FileStream(path, FileMode.Append, FileAccess.Write);

StreamWriter writer = new StreamWriter(fileStream);

writer.WriteLine("Account Number:" + AccountNumber);

writer.WriteLine("Customer Name:" + CustomerName);

writer.WriteLine("Customer Mobile Number:" + CustomerMobile);

writer.WriteLine("Customer Address:" + CustomerAddress);

Console.WriteLine("Data written successfully....");

writer.Close();

fileStream.Close();

}

catch(Exception ex )

{

Console.WriteLine(ex.Message);

}

#endregion

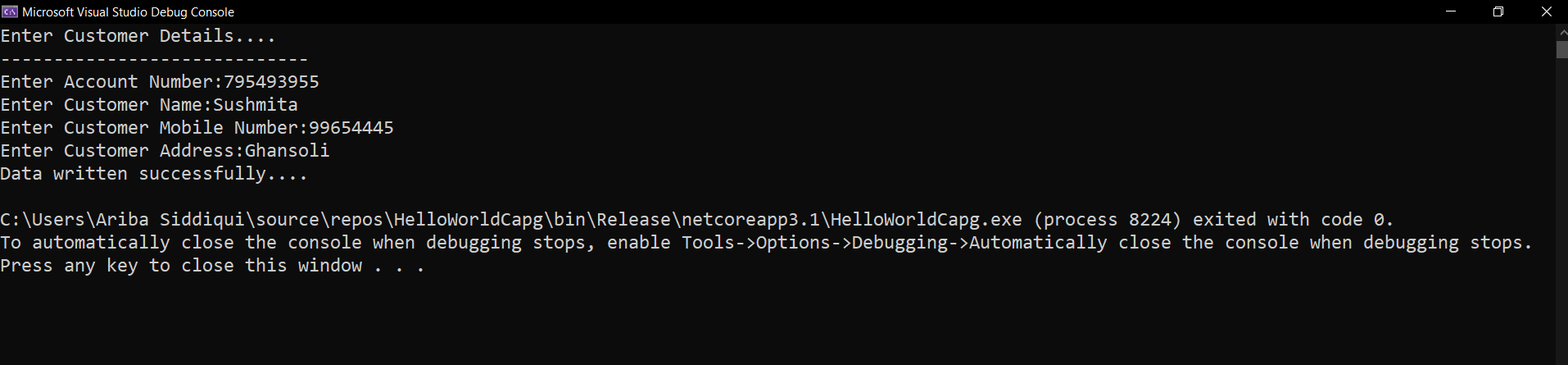
}

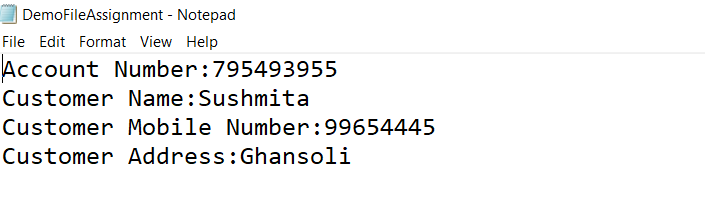
}

}

#endregion

**Output:**





**Stream Reader Code:**

using System;

using System.IO;

namespace FileHandling

{

class StreamReaderExample

{

static void Main()

{

#region File Operation

try

{

string path = @"D:/DemoFileAssignment.txt";

FileStream fileStream = new FileStream(path, FileMode.Open, FileAccess.Read);

StreamReader reader = new StreamReader(fileStream);

Console.WriteLine(reader.ReadToEnd());

Console.WriteLine("Data Read successfully....");

reader.Close();

fileStream.Close();

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

#endregion

}

}

}

