ПО

1 using System;

using System.IO;

namespace HelloApp

{

class Program

{

static void Main(string[] args)

{

DriveInfo[] drives = DriveInfo.GetDrives();

foreach (var drive in DriveInfo.GetDrives())

{

try

{

Console.WriteLine("Имя диска: " + drive.Name);

Console.WriteLine("Файловая система: " + drive.DriveFormat);

Console.WriteLine("Тип диска: " + drive.DriveType);

Console.WriteLine("Объем доступного свободного места (в байтах): " + drive.AvailableFreeSpace);

Console.WriteLine("Готов ли диск: " + drive.IsReady);

Console.WriteLine("Корневой каталог диска: " + drive.RootDirectory);

Console.WriteLine("Общий объем свободного места, доступного на диске (в байтах): " + drive.TotalFreeSpace);

Console.WriteLine("Размер диска (в байтах): " + drive.TotalSize);

Console.WriteLine("Метка тома диска: " + drive.VolumeLabel);

}

catch { }

Console.WriteLine();

}

}

}

}

2 ) using System;

using System.IO;

using System.Text.Json;

namespace ConsoleApp1

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Задание 2");

string path = @"D:\Documents";

DirectoryInfo dirInto = new DirectoryInfo(path);

FileInfo fileInf = new FileInfo(@"D:\Documents\prak.txt");

if (dirInfo.Exists)

{

dirInfo.Create();

}

Console.WriteLine("Введите строку для записи в файл:");

string text = Console.ReadLine();

using (FileStream fstream = new FileStream(@"D:\Documents\prak.txt",

FileMode.OpenOrCreate))

{

byte[] array = System.Text.Encoding.Default.GetBytes(text);

fstream.Write(array, 0, array.Length);

}

using (FileStream fstream = File.OpenRead(@"D:\Documents\prak.txt"))

{

byte[] array = new byte[fstream.Lenght];

fstream.Read(array, 0, array.Length);

string textFromFile = System.Text.Encoding.Default.GetString(array);

Console.WriteLine($"Текст из файла: {textFromFile}");

}

if (fileInf.Exists)

{

fileInf.Delete();

Console.WriteLine("Файл удален!");

}

}

3 using System;

using System.IO;

using System.Text.Json;

namespace ConsoleApp2

{

class Person

{

public string Name { get; set; }

public int Age { get; set; }

}

class Program

{

static async Task Main(string[] args)

{

using (FileStream fs = new FileStream("user.json", FileMode.OpenOrCreate))

{

Person tom = new Person() { Name = "Tom", Age = 35 };

await JsonSerializer.SerializeAsync<Person>(fs, tom);

Console.WriteLine("Data has been saved to file");

}

using (FileStream fs = new FileStream("user.json", FileMode.OpenOrCreate))

{

Person restoredPerson = await JsonSerializer.DeserializeAsync<Person>(fs);

Console.WriteLine($"Name: {restoredPerson.Name} Age: {restoredPerson.Age}");

}

Console.WriteLine("Если хотите удалить файл нажмите a, если хотите оставить b");

string a = Console.ReadLine();

if (a == "a")

{

string path = @"D:/Documents/user.json";

FileInfo fileInf = new FileInfo(path);

if (fileInf.Exists)

{

fileInf.Delete();

}

Console.WriteLine("Файл удалился");

}

else

{

Console.WriteLine("Файл остался");

}

}

}

}

4

using System;

using System.IO;

using System.Text;

class Test

{

public static void Main()

{

Console.WriteLine("1 - вывод состояния дисков \n2- Создать файл\n3- Записать текст в файл \n4- Считать файл\n5- Удалить файл ");

Console.Write("Введите номер функции - ");

int caseSwitch = Convert.ToInt32(Console.ReadLine());

switch (caseSwitch)

{

case 1:

DriveInfo[] Drivesincomp = DriveInfo.GetDrives();

foreach (DriveInfo d in Drivesincomp)

{

Console.WriteLine("Drive {0}", d.Name);

Console.WriteLine(" Drive type: {0}", d.DriveType);

if (d.IsReady == true)

{

Console.WriteLine(" Volume label: {0}", d.VolumeLabel);

Console.WriteLine(" File system: {0}", d.DriveFormat);

Console.WriteLine( " Available space to current user:{0, 15} bytes", d.AvailableFreeSpace);

Console.WriteLine( " Total available space: {0, 15} bytes", d.TotalFreeSpace);

Console.WriteLine( " Total size of drive: {0, 15} bytes ", d.TotalSize);

}

}

break;

case 2:

string path = @"D:\Documents\STASYAN\TESTPROG.xml";

try

{

using (FileStream fs = File.Create(path))

{

byte[] info = new UTF8Encoding(true).GetBytes("Создан");

fs.Write(info, 0, info.Length);

}

using (StreamReader sr = File.OpenText(path))

{

string s = "";

while ((s = sr.ReadLine()) != null)

{

Console.WriteLine(s);

}

}

}

catch (Exception ex)

{

Console.WriteLine(ex.ToString());

}

break;

case 3:

try

{

StreamWriter sw = new StreamWriter("D:\\Documents\\STASYAN\\TESTPROG.xml");

sw.WriteLine("Люблю кушать печенье! и Люблю смотреть кино!");

sw.Close();

}

finally

{

Console.WriteLine("Записанно!");

}

break;

case 4:

String line;

try

{

StreamReader sr = new StreamReader("D:\\Documents\\STASYAN\\TESTPROG.xml");

line = sr.ReadLine();

while (line != null)

{

Console.WriteLine(line);

line = sr.ReadLine();

}

sr.Close();

Console.ReadLine();

}

finally

{

Console.WriteLine("Прочтено!");

}

break;

case 5:

string path1 = @"D:\Documents\STASYAN\TESTPROG.xml";

FileInfo fi1 = new FileInfo(path1);

try

{

using (StreamWriter sw = fi1.CreateText()) { }

string path2 = path1;

FileInfo fi2 = new FileInfo(path2);

fi2.Delete();

fi2.Delete();

Console.WriteLine("{0} was successfully deleted.", path2);

}

catch (Exception e)

{

Console.WriteLine("The process failed: {0}", e.ToString());

}

break;

}

Console.ReadKey();

}

}

5 using System;

using System.IO;

using System.Text;

class Test

{

public static void Main()

{

Console.WriteLine("1 - вывод состояния дисков \n2- Создать файл\n3- Записать текст в файл \n4- Считать файл\n5- Удалить файл ");

Console.Write("Введите номер функции - ");

int caseSwitch = Convert.ToInt32(Console.ReadLine());

switch (caseSwitch)

{

case 1:

DriveInfo[] Drivesincomp = DriveInfo.GetDrives();

foreach (DriveInfo d in Drivesincomp)

{

Console.WriteLine("Drive {0}", d.Name);

Console.WriteLine(" Drive type: {0}", d.DriveType);

if (d.IsReady == true)

{

Console.WriteLine(" Volume label: {0}", d.VolumeLabel);

Console.WriteLine(" File system: {0}", d.DriveFormat);

Console.WriteLine( " Available space to current user:{0, 15} bytes", d.AvailableFreeSpace);

Console.WriteLine( " Total available space: {0, 15} bytes", d.TotalFreeSpace);

Console.WriteLine( " Total size of drive: {0, 15} bytes ", d.TotalSize);

}

}

break;

case 2:

string path = @"D:\Documents\STASYAN\TESTPROG.zip";

try

{

using (FileStream fs = File.Create(path))

{

byte[] info = new UTF8Encoding(true).GetBytes("Создан");

fs.Write(info, 0, info.Length);

}

using (StreamReader sr = File.OpenText(path))

{

string s = "";

while ((s = sr.ReadLine()) != null)

{

Console.WriteLine(s);

}

}

}

catch (Exception ex)

{

Console.WriteLine(ex.ToString());

}

break;

case 3:

try

{

StreamWriter sw = new StreamWriter("D:\\Documents\\STASYAN\\TESTPROG.zip");

sw.WriteLine("Люблю кушать печенье! и Люблю смотреть кино!");

sw.Close();

}

finally

{

Console.WriteLine("Записанно!");

}

break;

case 4:

String line;

try

{

StreamReader sr = new StreamReader("D:\\Documents\\STASYAN\\TESTPROG.zip");

line = sr.ReadLine();

while (line != null)

{

Console.WriteLine(line);

line = sr.ReadLine();

}

sr.Close();

Console.ReadLine();

}

finally

{

Console.WriteLine("Прочтено!");

}

break;

case 5:

string path1 = @"D:\Documents\STASYAN\TESTPROG.zip";

FileInfo fi1 = new FileInfo(path1);

try

{

using (StreamWriter sw = fi1.CreateText()) { }

string path2 = path1;

FileInfo fi2 = new FileInfo(path2);

fi2.Delete();

fi2.Delete();

Console.WriteLine("{0} was successfully deleted.", path2);

}

catch (Exception e)

{

Console.WriteLine("The process failed: {0}", e.ToString());

}

break;

}

Console.ReadKey();

}

}