|  |
| --- |
| Day15 Morning Assignment  By  Anusha Bellala  11-2-2022 |

|  |
| --- |
| 1. Research and write atleast 10 methods present in File Class. Illustrate with code example. |
| Code: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day15Project1  {  internal class Program  {  static void Main(string[] args)  {  // Assigning a File Path using Verbatim String Manipulation method.  string fileName = @"F:\c# programs\Day15 Morning Assignment\Hello.txt";  // Creating a File with Given File Path  StreamWriter sw = File.CreateText(fileName);  // When we Use WriteLine Method in StreamWriter class,  // It will Write in New Line each time it is called.  sw.WriteLine("Hi this is my first File Operations using C# code");  // The below statement is to have an empty line break in the file.  sw.WriteLine("");  sw.WriteLine("Writing Data using StringWriter, by using WriteLine Method.");  // When We Use Write Method in StreamWriter Class,  // it will start from the place of Cursor Exits after the last update, If Any.  sw.Write("This Line is by Write Method");  sw.Write("This is Second Line using Write Method\n");  sw.Close();  // Appending a Text  sw = File.AppendText(fileName);  sw.WriteLine("This");  sw.WriteLine("is Extra");  sw.WriteLine("Text");  Console.WriteLine("\nFile Appending is Done by Append\_Text Method.");  // We Need to Close the File, When ever we Create/open/Read a File, in file Operations.  sw.Close();  // Reading a File Using OpenText() Method  StreamReader sr = File.OpenText(fileName);  string s;  Console.WriteLine("\n");  while ((s = sr.ReadLine()) != null)  Console.WriteLine(s);  sr.Close();  string fileNewPath = @"F:\c# programs\FileOperations.txt";  // Moving a File From One Path to Another Path  if (File.Exists(fileNewPath))  File.Delete(fileNewPath);  Console.WriteLine("\nAlready the File is Present, So Deleting the old file & Creating a New File.");  File.Move(fileName, fileNewPath);  Console.WriteLine("\nFile Moved to New Path, Successfuly [F:\\c# programs\\FileOperations.txt]");  // Copying the File From NewPAth to Old path to make a Duplicate.  /\*if (File.Exists(fileName))  File.Delete(fileName);  Console.WriteLine("\nAlready the File is Present, So Deleting the old file & Creating a New File.");\*/  File.Copy(fileNewPath, fileName);  Console.WriteLine("\nFile Copying is Done Successfully, to old Path\n");  // Opening a Text File  File.OpenText(fileName).Close();  Console.WriteLine("\nFile opened Successfully, without any Errors");  string appendText = "This is an Extra text from Append\_All\_Text Method";  File.AppendAllText(fileName, appendText, Encoding.UTF8);  Console.WriteLine("\nFile Appended with Extra Text , Successfully");  string readText = File.ReadAllText(fileName);  Console.WriteLine("\n\n Reading All Text From the File\n");  Console.WriteLine(readText);  Console.ReadLine();  }  }  } |
| Output: |

|  |
| --- |
| 2. WACP to copy files from one folder to other folder. Schedule this job to be executed at daily some time. |
| Code: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day15Project2  {  internal class Program  {  static void Main(string[] args)  {  string filePath = @"F:\c# programs\Day15 Morning Assignment\Hello.txt";  string TaskSchedularPath = @"F:\c# programs\TaskSchedularPath.txt";  if (File.Exists(TaskSchedularPath))  File.Delete(TaskSchedularPath);  File.Copy(filePath, TaskSchedularPath);  Console.WriteLine("File Copying is Done");  Console.ReadKey();  }  }  } |
| Output:  1.)Open Task Schedular in Windows App Menu.  2.)Select “Create Task” Option from the Right-Side Menu. |

|  |
| --- |
| 3.)Give Name as : File Copying & Description as : File Copying @ 00.01.20AmEveryday from one folder to another file/folder. |

|  |
| --- |
| 4.)Select Triggers Tab & choose Time and Date as per your choice.  Select Daily in the Left-side Pane. (I chosen One Time as per my choice of running  only once. ) And Press “OK” button |

|  |
| --- |
| 5.)Now the New Trigger is Created. |

|  |
| --- |
| 6.)Go To “Actions” Tab. Click on “New” to create an action to be trigged by the task schedular. |

|  |
| --- |
| 7.)Now Chose “Browse” button, to add the Exe file of Our Project Code to be executed  by the task schedular on the given trigger time. |

|  |
| --- |
| 8.)Select the Exe file of your Project code Created when we Rebuild. |

|  |
| --- |
| 9.)Press “OK” Button When Selection of EXE file is Done. |

|  |
| --- |
| 10.)Final Step. Select the “OK” Button Which is Highlighted**.** |

|  |
| --- |
| 11.)Task Schedular is Scheduled, and When the timer Hits the trigged Value given in Step (4). The Output is Shown in the below image. |

|  |
| --- |
| 3. WACP to write data into file (and append the data) using Stream writer class |
| Code: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day15Project3  {  internal class Program  {  static void Main(string[] args)  {  // We are Creating a File using streamWriter Class. It will over Ride Each time we run the code  StreamWriter sw = new StreamWriter(@"F:\c# programs\Day15 Morning Assignment\StreamWriterExample.txt");  sw.WriteLine("Hi this is Anusha");  sw.WriteLine("This is a File Operation using StreamWriter with WriteLine Method");  sw.Close();  Console.WriteLine("\n Writing File Is done, by using Stream Writer Class by writeLine Method.\n");  // We are Creating a File using streamWriter Class. It will not over Ride Each time when we run the code.  // Instead it will Append the Text of Lines, Each time We Run the Code.  StreamWriter writer = new StreamWriter(@"F:\c# programs\Day15 Morning Assignment\StreamWriterAppendExample.txt", true);  writer.WriteLine("This is From New object of Stream Writer,");  writer.WriteLine("using Append by assigning True, while creating object for StreamWriter");  writer.Close();  Console.WriteLine("\n Appending is done, by using Stream Writer Class, By Enabling Append Method(true).");  Console.ReadKey();  }  }  } |
| Output: |

|  |
| --- |
| 4. Research and write C# program to read data from file. |
| Code: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day15Project4  {  internal class Program  {  static void Main(string[] args)  {  // We are Creating a File using streamWriter Class. It will not over Ride Each time when we run the code.  // Instead it will Append the Text of Lines, Each time We Run the Code.  StreamWriter writer = new StreamWriter(@"F:\c# programs\Day15 Morning Assignment\StreamWriterAppendExample.txt", true);  writer.WriteLine("This is From New object of Stream Writer,");  writer.WriteLine("using Append by assigning True, while creating object for StreamWriter");  writer.Close();  Console.WriteLine("\n Appending is done, by using Stream Writer Class, By Enabling Append Method(true).");  // We Are Reading the File Content using Stream Writer Class From System.IO namespace.  StreamReader reader = new StreamReader(@"F:\c# programs\Day15 Morning Assignment\StreamWriterAppendExample.txt");  reader.ReadToEnd();  reader.Close();  Console.WriteLine("\n Reading is done, by using Stream Writer Class, by ReadToEnd Method.");  Console.ReadKey();  }  }  } |
| Output: |

|  |
| --- |
| 5. Modify the quiz application to save the name and score in the flat file. No need to display the score to end user. |
| Code: |
| using System;  using System.Collections.Generic;  using System.IO;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day15Project5  {  internal class Program  {  static void Main(string[] args)  {  int score = 0, ans;  string name;  Console.WriteLine("Enter your name:");  name = Console.ReadLine();  Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  Console.WriteLine("Hi {0}, Welcome to quiz by Anusha", name);  Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  Console.WriteLine("Q1: What's your favorite food for chinese?");  Console.WriteLine("1.FriedRice 2.lemonRice 3.Biryani 4.PannerBiryani");  Console.WriteLine("Enter u r choice:");  ans = Convert.ToInt32(Console.ReadLine());  if (ans == 1)  score += 20;  Console.WriteLine("Q2: What's your favorite fast food?");  Console.WriteLine("1.FriedRice 2.lemonRice 3.Manchuria 4.PannerBiryani");  Console.WriteLine("Enter u r choice:");  ans = Convert.ToInt32(Console.ReadLine());  if (ans == 3)  score += 20;  Console.WriteLine("Q3: What's your favorite Drink?");  Console.WriteLine("1.FriedRice 2.ThumsUp 3.Biryani 4.Cocktail");  Console.WriteLine("Enter u r choice:");  ans = Convert.ToInt32(Console.ReadLine());  if (ans == 2)  score += 20;  Console.WriteLine("Q4: What's your favorite snack?");  Console.WriteLine("1.Cakes 2.lemonRice 3.Noodles 4.PannerBiryani");  Console.WriteLine("Enter u r choice:");  ans = Convert.ToInt32(Console.ReadLine());  if (ans == 1)  score += 20;  Console.WriteLine("Q5: What's your favorite kind of ethnic food?");  Console.WriteLine("1.Cakes 2.lemonRice 3.Noodles 4.PalakPaneer");  Console.WriteLine("Enter u r choice:");  ans = Convert.ToInt32(Console.ReadLine());  if (ans == 4)  score += 20;  StreamWriter sw = new StreamWriter("F:\\c# programs\\Scores\_Results.txt", true);  sw.WriteLine(" Name : {0} \n Score : {1}\n\n", name, score);  sw.Close();  Console.ReadLine();  }  }  } |
| Output: |

