Assignment 2 Week 1

Part-B

⑤ Write C# program.

1. How do I save a stream to a file in C#?

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Week1Assignment2ansreal

{

class Class2

{

public static void Main()

{

byte[] buff = File.ReadAllBytes(@"C:\Users\v-rapj\Downloads\MMSposter-large.jpg");

MemoryStream ms = new MemoryStream(buff);

FileStream file = new FileStream("d:\\file.txt", FileMode.Create, FileAccess.Write);

ms.WriteTo(file);

file.Close();

ms.Close();

Console.ReadKey();

}

}

}

1. You need to write data from a collection to a log file.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Week1Assignment2ansreal

{

class Class4

{

List<string> name = new List<string>();

public void copycollect()

{

using (StreamWriter fileWrite = new StreamWriter(@"D:\mycollect.txt"))

foreach (string na in name)

{

fileWrite.WriteLine(na);

Console.WriteLine(na);

}

}

public void mycollect()

{

name.Add("Raj");

name.Add("Nishanth");

name.Add("Akshay");

name.Add("Harsh");

name.Add("Anita");

}

static void Main()

{

Class4 obj = new Class4();

obj.mycollect();

obj.copycollect();

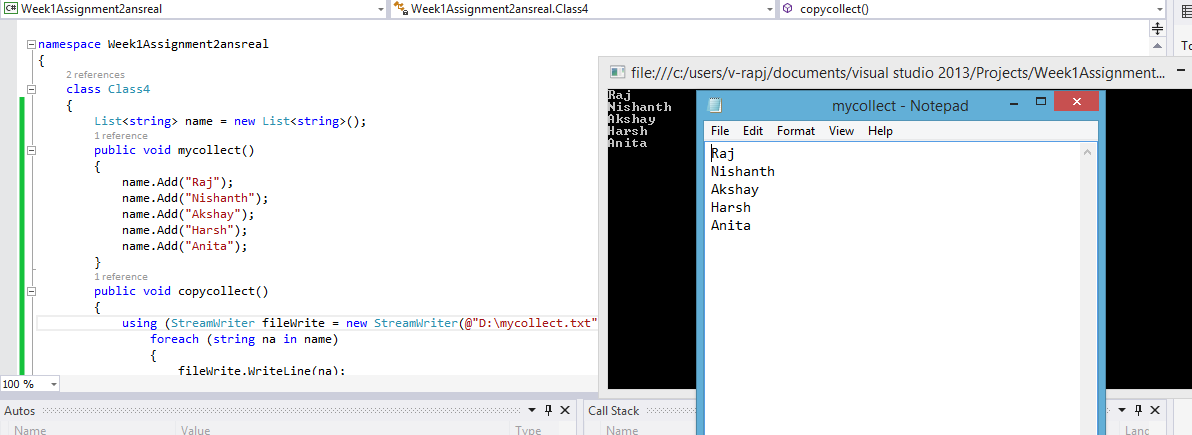
Console.ReadKey();

}

}

}

}



1. You need to read or write compressed data.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

using System.IO.Compression;

namespace Week1Assignment2ansreal

{

class Class6

{

static void Main()

{

GZipStream zipper = new GZipStream((File.OpenWrite(@"D:\mycompress.gzip")), CompressionMode.Compress);

StreamWriter writin = new StreamWriter(zipper);

writin.WriteLine("MS connect Programming");

writin.Close();

GZipStream zipop = new GZipStream((File.OpenRead(@"D:\mycompress.gzip")), CompressionMode.Decompress);

StreamReader rewrite = new StreamReader(zipop);

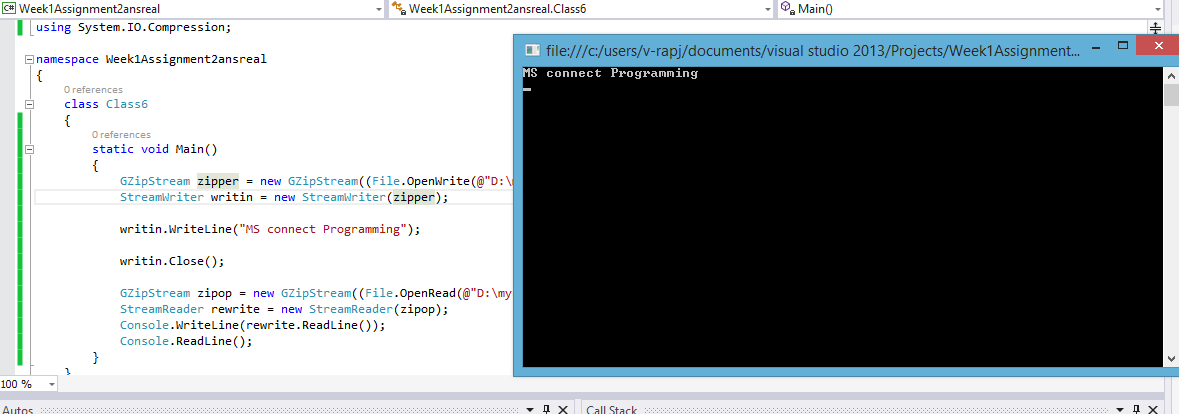
Console.WriteLine(rewrite.ReadLine());

Console.ReadLine();

}

}

}



4. You need to modify the access control list of a file or directory in the computer.

using System;

using System.IO;

using System.Security.AccessControl;

using System.Security.Principal;

namespace Week1Assignment2ansreal

{

class Class9

{

public static void Main()

{

DirectoryInfo direct = new DirectoryInfo(@"D:\file.txt");

DirectorySecurity access = direct.GetAccessControl();

access.AddAccessRule(new FileSystemAccessRule(new SecurityIdentifier(WellKnownSidType.WorldSid, null), FileSystemRights.FullControl, InheritanceFlags.ObjectInherit | InheritanceFlags.ContainerInherit, PropagationFlags.NoPropagateInherit, AccessControlType.Deny));

direct.SetAccessControl(access);

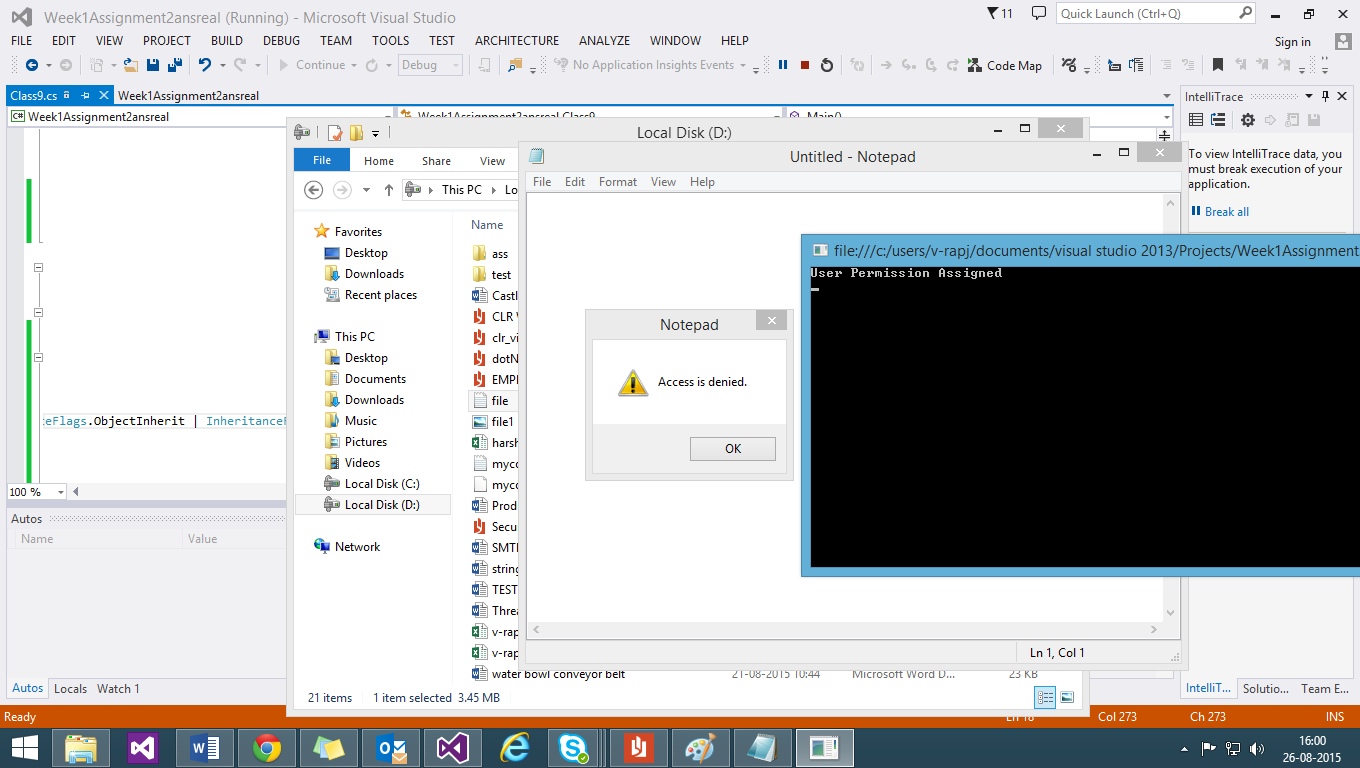
Console.WriteLine("User Permission Assigned");

Console.ReadKey();

}

}

}



6. Get the total free space on a Drive

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Week1Assignment2ansreal

{

class Class7

{

DriveInfo[] alldrives;

public static void Main()

{

Class7 driv = new Class7();

driv.GetDrive();

Console.Write("Enter the exact drive number for getting its free space :\t");

int drivenum=Int32.Parse(Console.ReadLine());

driv.GetTotalFreeSpace(drivenum);

Console.ReadKey();

}

public void GetDrive()

{

Console.WriteLine("The drives in the system are :");

alldrives= DriveInfo.GetDrives();

for (int iter = 0; iter < alldrives.Length;iter++ )

{

Console.WriteLine("Press "+(iter+1)+" for "+alldrives[iter]);

}

}

private void GetTotalFreeSpace(int number)

{

foreach (DriveInfo drive in alldrives)

{

if (drive.IsReady && drive.Name == alldrives[number-1].ToString())

{

Console.WriteLine("{0} has free space of {1}",drive.Name,drive.AvailableFreeSpace);

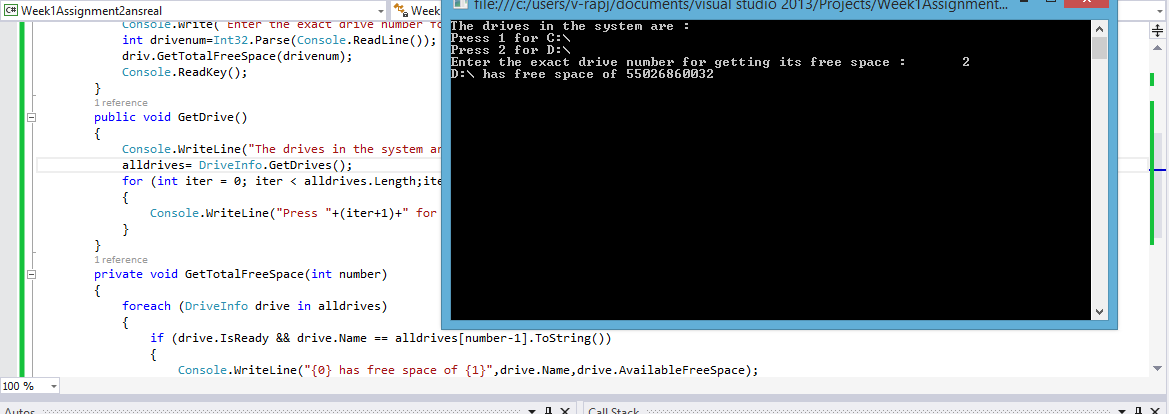
}

}

}

}

}



7. Determine if a path is a directory or a file.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Week1Assignment2ansreal

{

class Class8

{

public static void Main()

{

Console.Write("Enter the Path :\t");

string path=Console.ReadLine();

Class8 check = new Class8();

check.GetInfo(path.ToUpper());

Console.ReadKey();

}

public void GetInfo(string input)

{

try

{

FileAttributes attr = File.GetAttributes(input);

if (attr.HasFlag(FileAttributes.Directory))

Console.WriteLine("Its a directory");

else

Console.WriteLine("Its a file");

}

catch

{

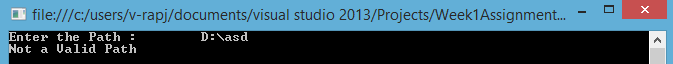
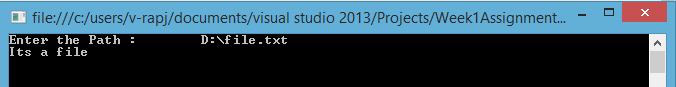
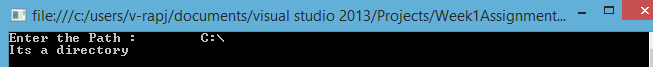
Console.WriteLine("Not a Valid Path");

}

}

}

}



8. You need to add nodes to an XML document

using System;

using System.Threading.Tasks;

using System.IO;

using System.Xml;

namespace Week1Assignment2ansreal

{

class Class11

{

public static void Main()

{

XmlDocument doc = new XmlDocument();

doc.Load(@"D:\file2.xml");

XmlNode expiry = doc.CreateNode(XmlNodeType.Element, "Expiry", null);

expiry.InnerText = "05/11/2015";

doc.DocumentElement.AppendChild(expiry);

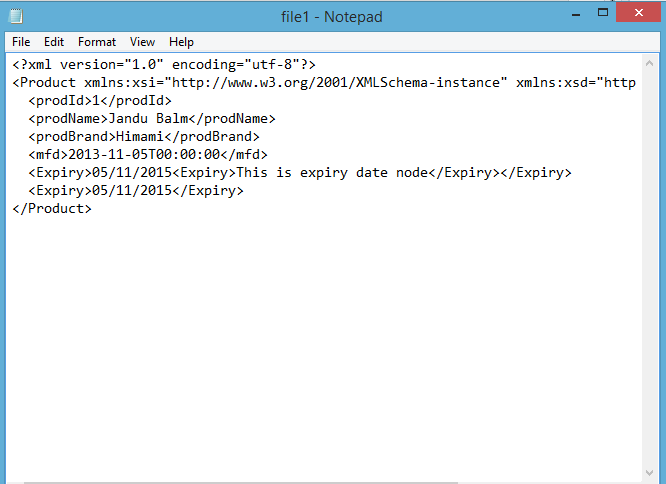
doc.Save(@"D:\file1.xml");

Console.ReadKey();

}

}

}



9. You need to retrieve a specific node from an XMLDocument

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

using System.Xml;

namespace Week1Assignment2ContinuePartB

{

class Class1

{

public void fetch()

{

XmlDocument docObj = new XmlDocument();

docObj.Load(@"D:\file1.xml");

XmlNodeList myxList = docObj.SelectNodes("Product"); ;

foreach (XmlNode xlist in myxList)

{

XmlNode mynode = xlist.SelectSingleNode("Expiry");

if (mynode != null)

{

string node = xlist["Expiry"].InnerText;

Console.WriteLine("Name: " + node);

}

}

}

public static void Main()

{

Class1 objClass = new Class1();

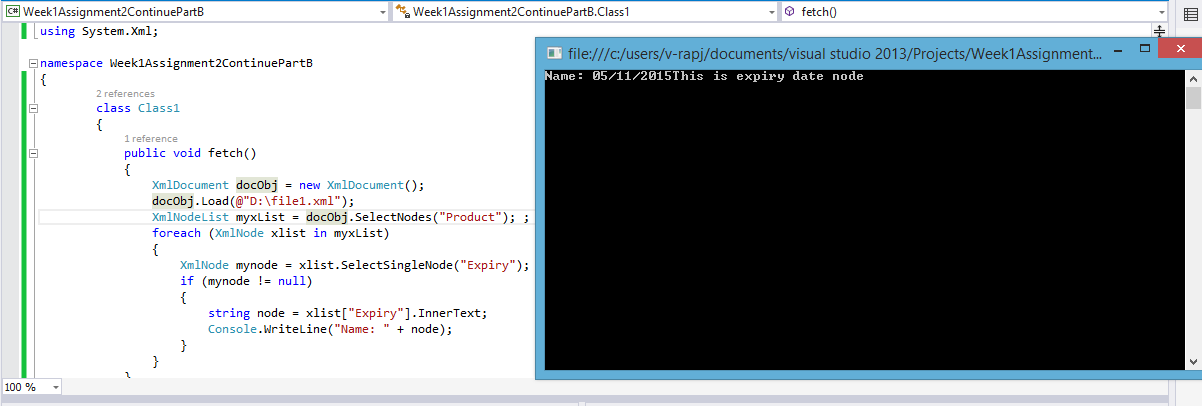
objClass.fetch();

Console.ReadLine();

}

}

}



10. Read/Write XML without loading entire document.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

using System.Xml;

namespace Week1Assignment2ContinuePartB

{

class Class1

{

public void fetch()

{

XmlDocument docObj = new XmlDocument();

docObj.Load(@"D:\file1.xml");

XmlNodeList myxList = docObj.SelectNodes("Product"); ;

foreach (XmlNode xlist in myxList)

{

XmlNode mynode = xlist.SelectSingleNode("Expiry");

if (mynode != null)

{

string node = xlist["Expiry"].InnerText;

Console.WriteLine("Name: " + node);

}

}

}

public static void Main()

{

Class1 objClass = new Class1();

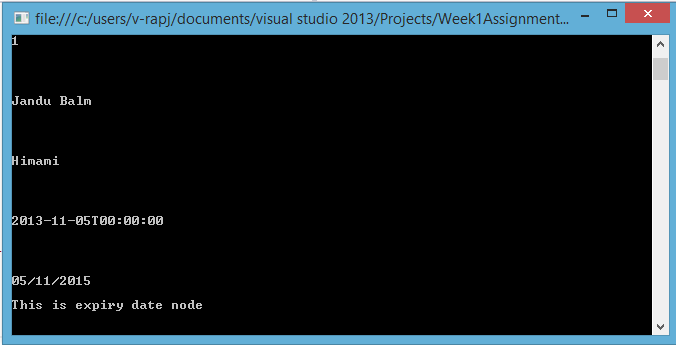
objClass.fetch();

Console.ReadLine();

}

}

}



11. Write a C# Program to demonstrate the use of XML Serialization.

using System;

using System.Xml.Serialization;

using System.IO;

namespace Week1Assignment2ansreal

{

public class Class10

{

static void Main(string[] args)

{

Product myprod = new Product();

myprod.prodId = 1;

myprod.prodName = "Jandu Balm";

myprod.prodBrand = "Himami";

string date="5 / 11 / 2013";

myprod.mfd = Convert.ToDateTime(date);

XmlSerializer xmlconvert = new XmlSerializer(typeof(Product));

using (TextWriter writer = new StreamWriter(@"D:\file1.xml"))

{

xmlconvert.Serialize(writer, myprod);

}

Console.ReadKey();

}

public class Product

{

public int prodId { get; set; }

public string prodName { get; set; }

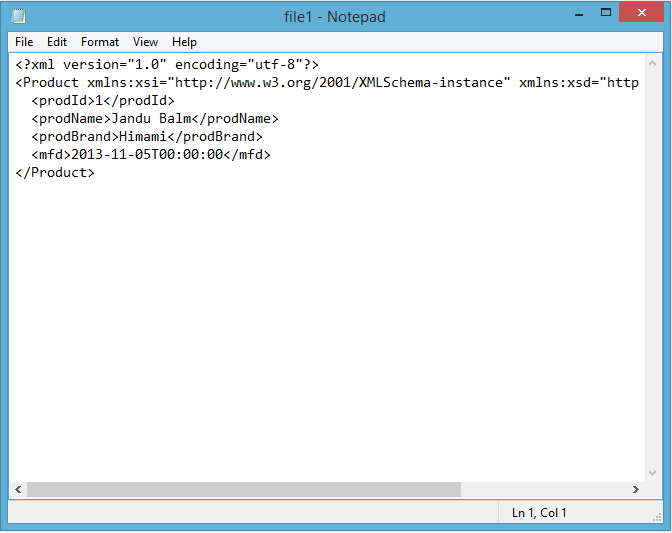
public string prodBrand { get; set; }

public DateTime mfd { get; set; }

}

}

}



13. Create C# program demonstrates threads in an array.

using System;

using System.Collections.Generic;

using System.Threading.Tasks;

using System.Threading;

namespace Week1Assignment2ansreal

{

class Class12

{

public void first()

{

int cnt = 0;

while (cnt < 5)

{

Console.WriteLine("first thread is executed {0} times", ++cnt);

Thread.Sleep(1);

}

}

public void second()

{

int cnt = 0;

while (cnt < 5)

{

Console.WriteLine("second thread is executed {0} times", ++cnt);

Thread.Sleep(1);

}

}

static void Main()

{

Class12 myobj=new Class12();

ThreadStart myfirst = new ThreadStart(myobj.first);

ThreadStart mysecond = new ThreadStart(myobj.second);

Thread[] mythread = new Thread[2];

mythread[0] = new Thread(myfirst);

mythread[1] = new Thread(mysecond);

foreach (Thread newthread in mythread)

{

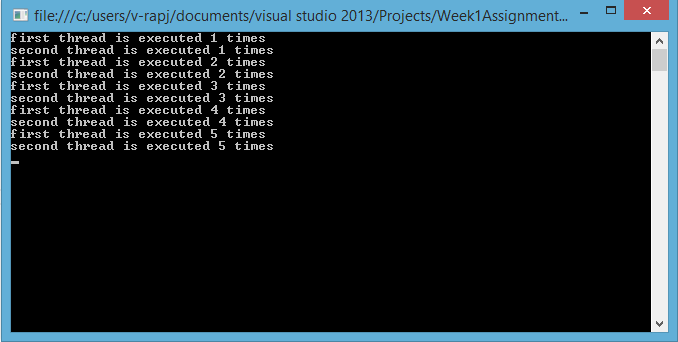
newthread.Start();

}

Console.ReadKey();

}

}

}

14. Create C# program demonstrates parameterized threads.

class Class14

{

static void Main()

{

Thread myThread = new Thread(Class14.DoWork);

myThread.Start(42);

Console.ReadLine();

}

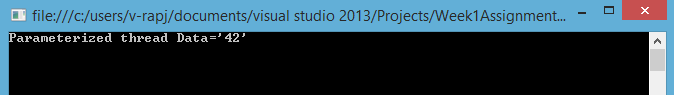
public static void DoWork(object data)

{

Console.WriteLine("Parameterized thread Data='{0}'",data);

}

}



15. Create C# program demonstrates threadpool.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Threading;

namespace Week1Assignment2ansreal

{

class Class13

{

static void Main()

{

Thread mythread = new Thread(Go);

mythread.Start();

Task.Factory.StartNew(Go);

Go();

Console.ReadLine();

}

static void Go()

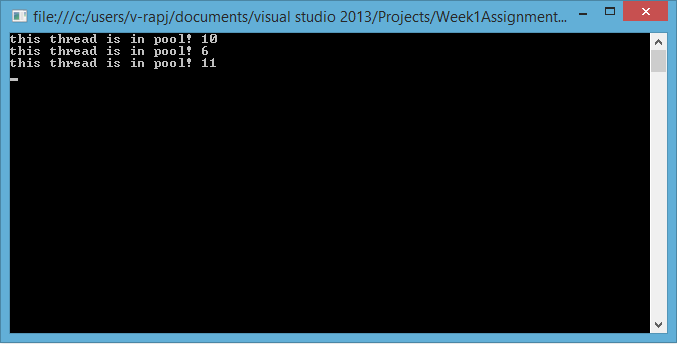
{

Console.WriteLine("this thread is in pool! {0} ", Thread.CurrentThread.GetHashCode());

}

}

}



17. Create C# program demonstrates deadlocks & locking mechanism.

class Class5

{

static void Main()

{

lock (typeof(long))

{

Thread.Sleep(1000);

lock (typeof(char))

Console.WriteLine("Executed Thread 1");

}

lock (typeof(char))

{

Thread.Sleep(1000);

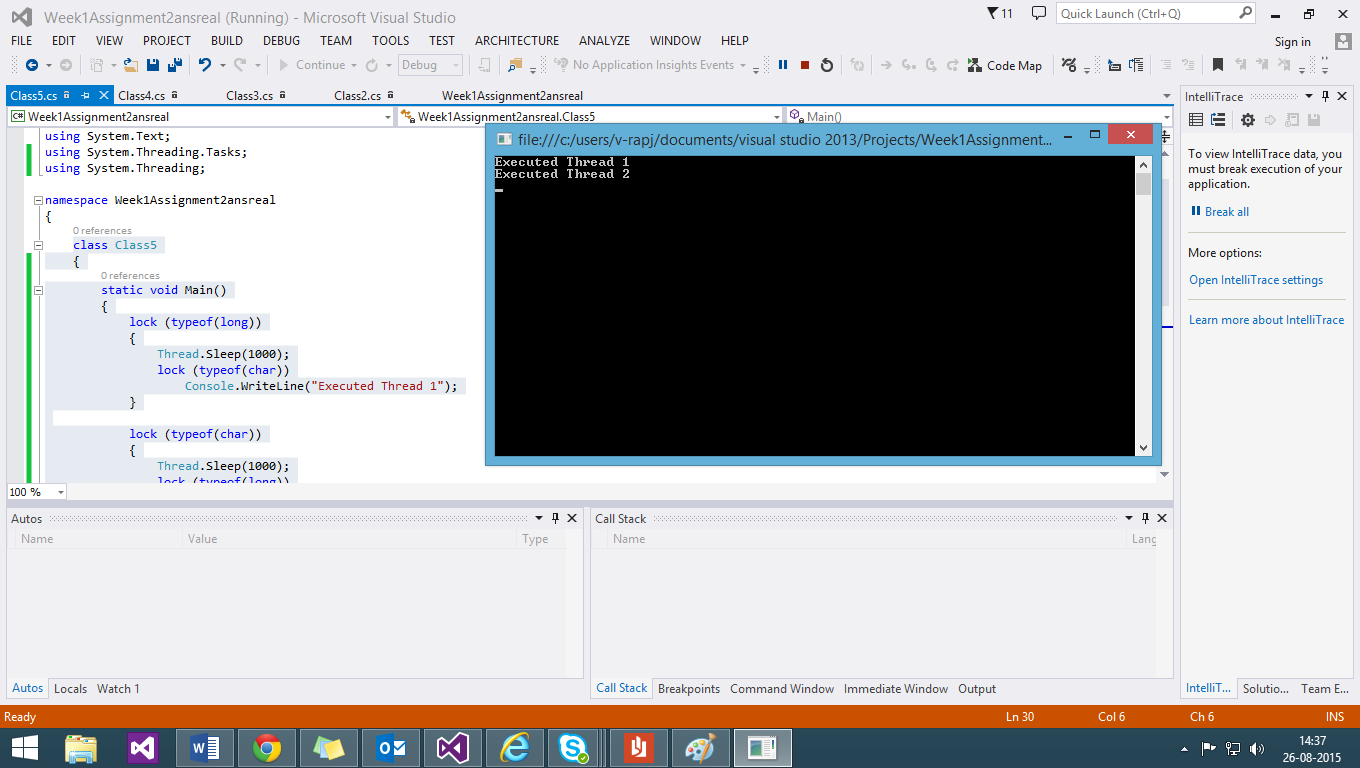
lock (typeof(long))

Console.WriteLine("Executed Thread 2");

}

Console.ReadLine();

}

}

18. Create C# program to demonstrate custom exception handler.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Week1Assignment2ContinuePartB

{

class Program : System.SystemException

{

static void Main(string[] args)

{

number numobj = new number();

try

{

Console.WriteLine("Custom Divide by Zero Exception");

Console.Write("Enter the Numerator:");

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

Console.Write("Enter the Denominator:");

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

numobj.div(numerator, denominator);

}

catch (DivideByZeroException except)

{

Console.WriteLine("MyDivideByZeroException: {0}", except.Message);

}

Console.ReadKey();

}

public class DivideByZeroException : ApplicationException

{

public DivideByZeroException(string msg): base(msg)

{

}

}

public class number

{

public void div(int num, int denom)

{

if (denom == 0)

throw (new DivideByZeroException("cannot divide by zero"));

else

{

double result = (double)num / (double)denom;

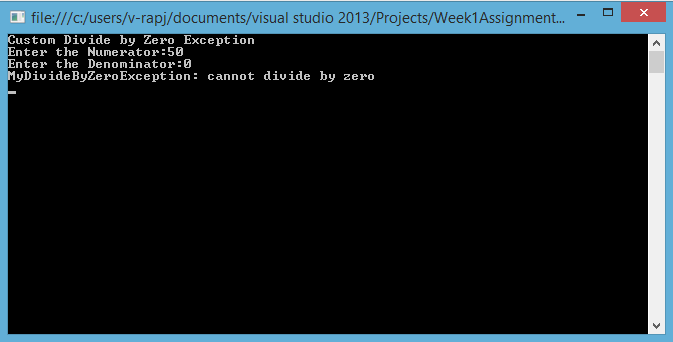
Console.WriteLine("Number is: {0}", result);

}

}

}

}

}

19. Create C# program to generate bitmap (or jpg) image on the fly from runtime

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Drawing;

namespace Week1Assignment2ContinuePartB

{

class Class2

{

public static void Main()

{

using (var bmp = new Bitmap(100, 100))

using (var gr = Graphics.FromImage(bmp))

{

gr.FillRectangle(Brushes.Azure, new Rectangle(0, 0, bmp.Width, bmp.Height));

var path = System.IO.Path.Combine(

Environment.GetFolderPath(Environment.SpecialFolder.Desktop),

"Example1.png");

bmp.Save(path);

Console.WriteLine(bmp.ToString());

}

Console.ReadKey();

}

}

}

OUTPUT:

