**NAME:MUHAMMAD HAMZA**

**SID:10040**

**DP ASSIGNMENT 02**

**STRATEGY**

**CODE:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace DPaSS02\_10040

{

public interface ICompression

{

void CompressFolder(string compressedArchiveFileName);

}

public class RarCompression : ICompression

{

public void CompressFolder(string compressedArchiveFileName)

{

Console.WriteLine("Folder is compressed using Rar approach: '" + compressedArchiveFileName + ".rar' file is created");

}

}

public class ZipCompression : ICompression

{

public void CompressFolder(string compressedArchiveFileName)

{

Console.WriteLine("Folder is compressed using zip approach: '" + compressedArchiveFileName + ".zip' file is created");

}

}

public class CompressionContext

{

private ICompression Compression;

public CompressionContext(ICompression Compression)

{

this.Compression = Compression;

}

public void SetStrategy(ICompression Compression)

{

this.Compression = Compression;

}

public void CreateArchive(string compressedArchiveFileName)

{

Compression.CompressFolder(compressedArchiveFileName);

}

}

class Program

{

static void Main(string[] args)

{

CompressionContext ctx = new CompressionContext(new ZipCompression());

ctx.CreateArchive("muhammadHamza10040");

ctx.SetStrategy(new RarCompression());

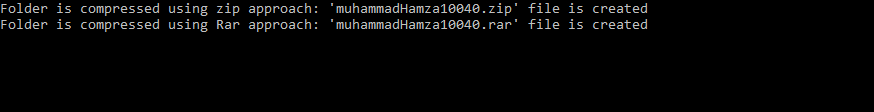
ctx.CreateArchive("muhammadHamza10040");

Console.ReadLine();

}

}

}

**OUTPUT:** 

**SINGLETON**

**CODE**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace SingleTon

{

sealed class SingleTon

{

private SingleTon()

{

}

public static SingleTon getInstance = null;

public static SingleTon myObject()

{

if (getInstance==null)

{

return new SingleTon();

}

return getInstance;

}

public void myMethod()

{

Console.WriteLine("This is MY Method \n Muhammad Hamza 10040 \n");

}

}

class Program

{

static void Main(string[] args)

{

SingleTon obj1 = SingleTon.myObject();

obj1.myMethod();

SingleTon obj2 = SingleTon.myObject();

obj2.myMethod();

Console.ReadLine();

}

}

}

**SS**