EXERCISE – 1 **Implementing the Singleton Pattern**

using System;

public sealed class Logger

{

private static Logger \_logger\_instance = null;

private static readonly object \_lock\_object = new object();

private int \_log\_count = 0;

private Logger()

{

Console.WriteLine("Logger instance created");

}

public static Logger Get\_Instance()

{

lock (\_lock\_object)

{

if (\_logger\_instance == null)

{

\_logger\_instance = new Logger();

}

return \_logger\_instance;

}

}

public void Log\_Message(string message)

{

\_log\_count++;

Console.WriteLine($"[Log Entry #{\_log\_count}] {DateTime.Now}: {message}");

}

}

class Program

{

static void Main()

{

Logger first\_logger = Logger.Get\_Instance();

first\_logger.Log\_Message("Application started");

Logger second\_logger = Logger.Get\_Instance();

second\_logger.Log\_Message("User logged in");

if (ReferenceEquals(first\_logger, second\_logger))

{

Console.WriteLine("Both references point to the SAME Logger instance");

}

else

{

Console.WriteLine("DIFFERENT Logger instances detected");

}

}

}  
OUTPUT  
A screenshot of a computer

AI-generated content may be incorrect.

**Exercise 2: Implementing the Factory Method Pattern**

**using System;**

**public interface document**

**{**

**void open();**

**void save();**

**}**

**public class word\_document : document**

**{**

**public void open()**

**{**

**Console.WriteLine("opening word document");**

**}**

**public void save()**

**{**

**Console.WriteLine("saving word document");**

**}**

**}**

**public class pdf\_document : document**

**{**

**public void open()**

**{**

**Console.WriteLine("opening pdf document");**

**}**

**public void save()**

**{**

**Console.WriteLine("saving pdf document");**

**}**

**}**

**public class excel\_document : document**

**{**

**public void open()**

**{**

**Console.WriteLine("opening excel document");**

**}**

**public void save()**

**{**

**Console.WriteLine("saving excel document");**

**}**

**}**

**public abstract class document\_factory**

**{**

**public abstract document create\_document();**

**}**

**public class word\_document\_factory : document\_factory**

**{**

**public override document create\_document()**

**{**

**return new word\_document();**

**}**

**}**

**public class pdf\_document\_factory : document\_factory**

**{**

**public override document create\_document()**

**{**

**return new pdf\_document();**

**}**

**}**

**public class excel\_document\_factory : document\_factory**

**{**

**public override document create\_document()**

**{**

**return new excel\_document();**

**}**

**}**

**class program**

**{**

**static void Main()**

**{**

**document\_factory word\_factory = new word\_document\_factory();**

**document word\_doc = word\_factory.create\_document();**

**word\_doc.open();**

**word\_doc.save();**

**document\_factory pdf\_factory = new pdf\_document\_factory();**

**document pdf\_doc = pdf\_factory.create\_document();**

**pdf\_doc.open();**

**pdf\_doc.save();**

**document\_factory excel\_factory = new excel\_document\_factory();**

**document excel\_doc = excel\_factory.create\_document();**

**excel\_doc.open();**

**excel\_doc.save();**

**}**

**}**

**OUTPUT :**

**A computer screen shot of a blue screen

AI-generated content may be incorrect.**