**CTS – DIGITAL NUTURE – 4.0 - DEEPSKILLING PROGRAM**

**WEEK – 1 🡪 DESIGN PATTERNS**

**Exercise 1: Implementing the Singleton Pattern**

**Code (Implementation):**

***Logger.java***

package com.cts.dp.handson.week1;

public class Logger

{

private static Logger *instance*;

private Logger()

{

System.***out***.println("Logger instance created");

}

public static Logger getInstance()

{

if (*instance* == null)

{

*instance* = new Logger();

}

return *instance*;

}

public void log(String message)

{

System.***out***.println("Log message: " + message);

}

}

***LoggerTest.java***

package com.cts.dp.handson.week1;

public class LoggerTest

{

public static void main(String[] args)

{

Logger logger1 = Logger.*getInstance*();

Logger logger2 = Logger.*getInstance*();

logger1.log("First message");

logger2.log("Second message");

if (logger1 == logger2)

{

System.***out***.println("Both logger instances are the same (Singleton confirmed).");

}

else

{

System.***out***.println("Different instances exist (Singleton failed).");

}

}

}

**Output:**

**A screenshot of a computer

AI-generated content may be incorrect.**

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

**Code (Implementation):**

***PdfDocument.java***

package com.cts.dp.handson.week1;

public interface PdfDocument

{

public void open();

}

***WordDocument.java***

package com.cts.dp.handson.week1;

public interface WordDocument

{

public void open();

}

***ExcelDocument.java***

package com.cts.dp.handson.week1;

public interface ExcelDocument

{

public void open();

}

***PdfDocumentImpl.java***

package com.cts.dp.handson.week1;

public class PdfDocumentImpl implements PdfDocument

{

*@Override*

public void open()

{

System.***out***.println("Opening PDF Document...");

}

}

***WordDocumentImpl.java***

package com.cts.dp.handson.week1;

public class WordDocumentImpl implements WordDocument

{

*@Override*

public void open()

{

System.***out***.println("Opening Word Document...");

}

}

***ExcelDocumentImpl.java***

package com.cts.dp.handson.week1;

public class ExcelDocumentImpl implements ExcelDocument

{

*@Override*

public void open()

{

System.***out***.println("Opening Excel Document...");

}

}

***DocumentFactory.java***

package com.cts.dp.handson.week1;

public abstract class DocumentFactory

{

public abstract Object createDocument();

}

***PdfFactory.java***

package com.cts.dp.handson.week1;

public class PdfFactory extends DocumentFactory

{

*@Override*

public PdfDocument createDocument()

{

return new PdfDocumentImpl();

}

}

***WordFactory.java***

package com.cts.dp.handson.week1;

public class WordFactory extends DocumentFactory

{

*@Override*

public WordDocument createDocument()

{

return new WordDocumentImpl();

}

}

***ExcelFactory.java***

package com.cts.dp.handson.week1;

public class ExcelFactory extends DocumentFactory

{

*@Override*

public ExcelDocument createDocument()

{

return new ExcelDocumentImpl();

}

}

***DocumentTest.java***

package com.cts.dp.handson.week1;

public class DocumentTest

{

public static void main(String[] args)

{

WordFactory wordFactory = new WordFactory();

WordDocument wordDoc = wordFactory.createDocument();

wordDoc.open();

PdfFactory pdfFactory = new PdfFactory();

PdfDocument pdfDoc = pdfFactory.createDocument();

pdfDoc.open();

ExcelFactory excelFactory = new ExcelFactory();

ExcelDocument excelDoc = excelFactory.createDocument();

excelDoc.open();

}

}

**Output:**

