

```

import java.util.*;
import java.nio.charset.StandardCharsets;
import java.nio.file.*;
import java.io.*;
public class ReadFileIntoList
{
    public static List<String> readFileInList(String fileName)
    {
        List<String> lines = Collections.emptyList();
        try
        {
            lines =
                Files.readAllLines(Paths.get(fileName), StandardCharsets.UTF_8);
        }

        catch (IOException e)
        {
            e.printStackTrace();
        }
        return lines;
    }
    public static void main(String[] args)
    {
        List l = readFileInList("c://temp//testFile2.txt");

        Iterator<String> itr = l.iterator();
        while (itr.hasNext())
            System.out.println(itr.next());
    }
}

```

```

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class TextFileModificationProgram
{
    static void modifyFile(String filePath, String oldString, String newString)
    {
        File fileToBeModified = new File(filePath);
        String oldContent = "";
    }
}

```

```

BufferedReader reader = null;
FileWriter writer = null;
try
{
    reader = new BufferedReader(new FileReader(fileToBeModified));
    String line = reader.readLine();
    while (line != null)
    {
        oldContent = oldContent + line + System.LineSeparator();
        line = reader.readLine();
    }
    String newContent = oldContent.replaceAll(oldString, newString);
    writer = new FileWriter(fileToBeModified);
    writer.write(newContent);
}
catch (IOException e)
{
    e.printStackTrace();
}
finally
{
    try
    {
        reader.close();
        writer.close();
    }
    catch (IOException e)
    {
        e.printStackTrace();
    }
}
}

public static void main(String[] args)
{
    modifyFile("c://temp//testFile2.txt", "85", "95");
    System.out.println("done");
}
}

```

```

import java.io.IOException;
import java.nio.file.*;

public class Test
{
    public static void main(String[] args)
    {
        try
        {
            Files.deleteIfExists(Paths.get("c://temp//testFile2.txt"));
        }
    }
}

```

```

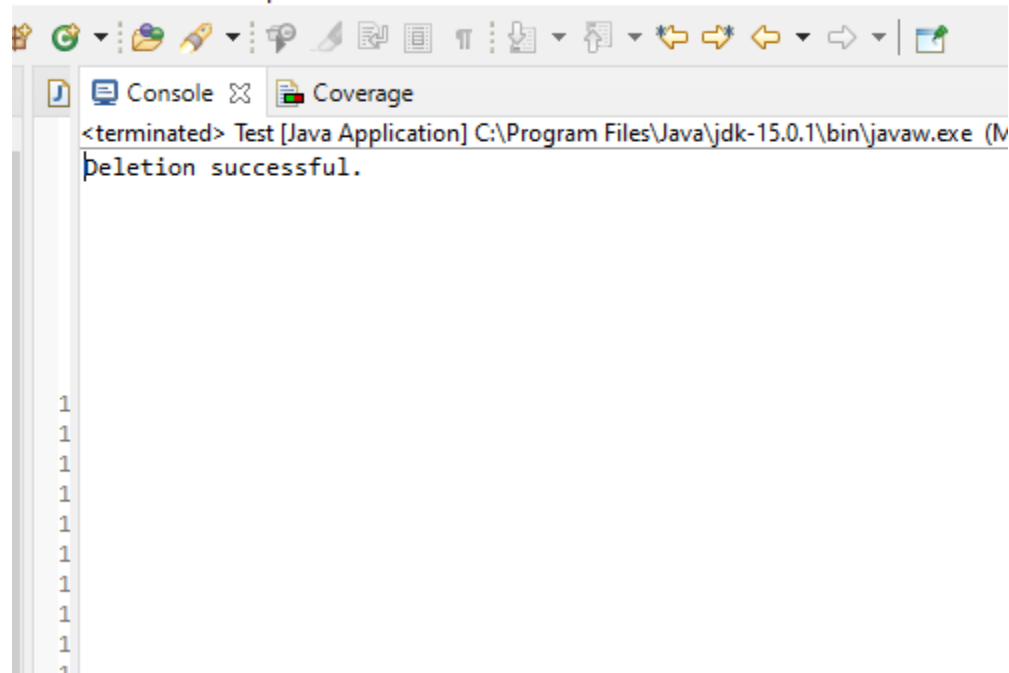
        catch(NoSuchFileException e)
        {
            System.out.println("No such file/directory exists");
        }
        catch(DirectoryNotEmptyException e)
        {
            System.out.println("Directory is not empty.");
        }
        catch(IOException e)
        {
            System.out.println("Invalid permissions.");
        }

        System.out.println("Deletion successful.");
    }
}

```

se IDE

Run Window Help



```

import java.io.File;
import java.io.FileOutputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.nio.file.StandardOpenOption;
import java.util.Arrays;
import java.util.List;

public class CreateNewFile
{
    public static void main(String[] args) throws IOException

```

```

{
    createFileUsingFileClass();
    createFileUsingFileOutputStreamClass();
    createFileIn_NIO();
}

private static void createFileUsingFileClass() throws IOException
{
    File file = new File("c://temp//testFile1.txt");

    //Create the file
    if (file.createNewFile()){
        System.out.println("File is created!");
    }else{
        System.out.println("File already exists.");
    }

    //Write Content
    FileWriter writer = new FileWriter(file);
    writer.write("Test data");
    writer.close();
}

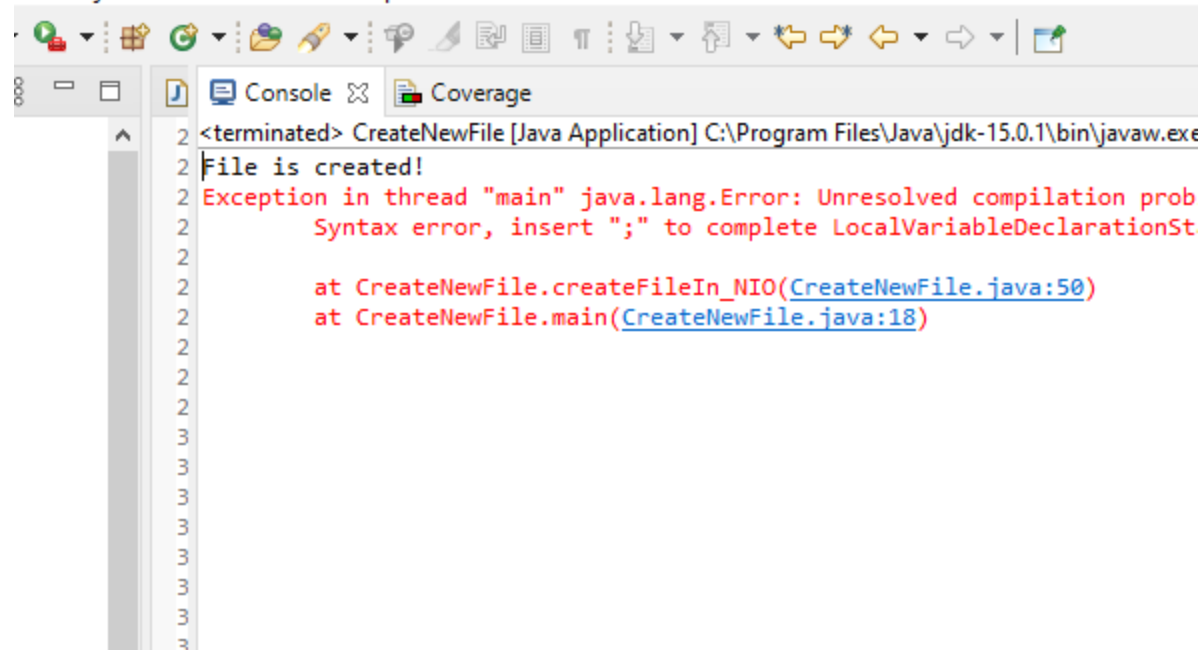
private static void createFileUsingFileOutputStreamClass() throws IOException
{
    String data = "Test data";
    FileOutputStream out = new FileOutputStream("c://temp//testFile2.txt");
    out.write(data.getBytes());
    out.close();
}

private static void createFileIn_NIO() throws IOException
{
    String data = "Test data";
    Files.write(Paths.get("c://temp//testFile3.txt"), data.getBytes());
    List<String> lines = Arrays.asList("1st line", "2nd line")
    Files.write(Paths.get("file6.txt"),
                lines,
                StandardCharsets.UTF_8,
                StandardOpenOption.CREATE,
                StandardOpenOption.APPEND);
}
}

```

teNewFile.java - Eclipse IDE

File Project Run Window Help



The screenshot shows the Eclipse IDE interface. The top menu bar includes File, Project, Run, Window, and Help. Below the menu is a toolbar with various icons for file operations, running, and debugging. The main window is divided into two panes: 'Console' and 'Coverage'. The 'Console' pane is active and displays the following output:

```
<terminated> CreateNewFile [Java Application] C:\Program Files\Java\jdk-15.0.1\bin\javaw.exe
File is created!
Exception in thread "main" java.lang.Error: Unresolved compilation prob
    Syntax error, insert ";" to complete LocalVariableDeclarationSt
    at CreateNewFile.createFileIn_NIO(CreateNewFile.java:50)
    at CreateNewFile.main(CreateNewFile.java:18)
```

The output indicates that the application successfully created a file but then crashed with a syntax error. The error message is truncated, but it suggests a missing semicolon in a local variable declaration. The stack trace points to line 50 in `CreateNewFile.java` and line 18 in the `main` method.