Academic Year: 2023-24 **Semester: II** **Class: FYMCA**

Course Code: MC506 **Course Name: Java Programming**

Name: Durgesh Dilip Mandge **UCID : 2023510032**

Experiment No.4

Date: 22-04-2024

1. W.A.P for the following operations of a file :- a. File Handling using Character Stream classes b. Write content onto file c. Reading Content from Files d. To get current path of file. e. file related operations like rename, last modification, specified director.

```
import java.io.*;

public class FileOperations {
    public static void main(String[] args) {
        String fileName = "sample.txt"; // File name

        // Write content onto file
        writeToFile(fileName, "Hello, this is some content to write to the
file.");

        // Reading content from file
        String content = readFromFile(fileName);
        System.out.println("Content read from file: " + content);

        // Get current path of file
        String currentPath = getCurrentPath(fileName);
        System.out.println("Current path of the file: " + currentPath);

        // File related operations
        renameFile(fileName, "renamed_sample.txt");
        displayLastModified(fileName);
        listFilesInDirectory(".");
    }

    // Write content onto file
```

```
    public static void writeToFile(String fileName, String content) {
        try (FileWriter writer = new FileWriter(fileName)) {
            writer.write(content);
            System.out.println("Content successfully written to file.");
        } catch (IOException e) {
            System.out.println("An error occurred while writing to the
file: " + e.getMessage());
        }
    }

    // Reading content from file
    public static String readFromFile(String fileName) {
        StringBuilder content = new StringBuilder();
        try (FileReader reader = new FileReader(fileName)) {
            int character;
            while ((character = reader.read()) != -1) {
                content.append((char) character);
            }
        } catch (IOException e) {
            System.out.println("An error occurred while reading from the
file: " + e.getMessage());
        }
        return content.toString();
    }

    // Get current path of file
    public static String getCurrentPath(String fileName) {
        File file = new File(fileName);
        return file.getAbsolutePath();
    }

    // Rename file
    public static void renameFile(String oldName, String newName) {
        File oldFile = new File(oldName);
        File newFile = new File(newName);
        if (oldFile.renameTo(newFile)) {
            System.out.println("File renamed successfully.");
        } else {
            System.out.println("Failed to rename the file.");
        }
```

```
    }

    // Display last modification time of file
    public static void displayLastModified(String fileName) {
        File file = new File(fileName);
        long lastModified = file.lastModified();
        System.out.println("Last modified time of the file: " +
lastModified);
    }

    // List files in specified directory
    public static void listFilesInDirectory(String directoryPath) {
        File directory = new File(directoryPath);
        File[] files = directory.listFiles();
        if (files != null) {
            System.out.println("Files in directory " + directoryPath +
":");
            for (File file : files) {
                System.out.println(file.getName());
            }
        } else {
            System.out.println("Specified directory does not exist or is
not a directory.");
        }
    }
}
```

OUTPUT:

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  c:; cd 'c:\Users\smart\Documents\SPIT-MCA\SPIT-lab
java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\smart\AppData\Roaming\Code\User\workspaceStora
Lab4_a91ea8f8\bin' 'FileOperations'
Content successfully written to file.
Content read from file: Hello, this is some content to write to the file.
Current path of the file: C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4\sample.txt
Failed to rename the file.
Last modified time of the file: 1713796912273
Files in directory .:
FileOperations.java
renamed_sample.txt
sample.txt
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```

2. Create a db "Employee". Perform DDL operation using JAVA use createstatement

```
import java.sql.*;

public class DDLExample {
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost:3306/";

    // Database credentials
    static final String USER = "root";
    static final String PASS = "nita";

    public static void main(String[] args) {
        Connection conn = null;
        Statement stmt = null;

        try {
            // Register JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");

            // Open a connection
            System.out.println("Connecting to database...");
            conn = DriverManager.getConnection(DB_URL, USER, PASS);

            // Execute DDL operations
            System.out.println("Creating database...");
            stmt = conn.createStatement();
            String sql = "CREATE DATABASE IF NOT EXISTS Employee";
            stmt.executeUpdate(sql);
            System.out.println("Database created successfully");

            // Select the database
            sql = "USE Employee";
            stmt.executeUpdate(sql);

            // Create table
            System.out.println("Creating table...");
            sql = "CREATE TABLE IF NOT EXISTS EmployeeDetails " +
```

```
                    "(id INT PRIMARY KEY AUTO_INCREMENT, " +
                    " name VARCHAR(255), " +
                    " age INT)";
            stmt.executeUpdate(sql);
            System.out.println("Table created successfully");

            // Alter table to add column
            System.out.println("Altering table to add column...");
            sql = "ALTER TABLE EmployeeDetails ADD COLUMN department
VARCHAR(255)";
            stmt.executeUpdate(sql);
            System.out.println("Table altered successfully");

            // Drop table
            System.out.println("Dropping table...");
            sql = "DROP TABLE EmployeeDetails";
            stmt.executeUpdate(sql);
            System.out.println("Table dropped successfully");

        } catch (SQLException se) {
            // Handle errors for JDBC
            se.printStackTrace();
        } catch (Exception e) {
            // Handle errors for Class.forName
            e.printStackTrace();
        } finally {
            // Finally block to close resources
            try {
                if (stmt != null) stmt.close();
            } catch (SQLException se2) {
            } // nothing we can do
            try {
                if (conn != null) conn.close();
            } catch (SQLException se) {
                se.printStackTrace();
            } // end finally try
        } // end try
        System.out.println("DDL operations completed");
    }
}
```

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  c:; cd 'c:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java'
Files\Java\jdk-21\bin\java.exe' '@C:\Users\smart\AppData\Local\Temp\cp_5ym5plbgz89v881a65csmag58.argfile' 'DDLExample'
Connecting to database...
Creating database...
Database created successfully
Creating table...
Table created successfully
Altering table to add column...
Table altered successfully
Dropping table...
Table dropped successfully
DDL operations completed
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```

3. Open a file "try.txt". using fileinputstream. show the content of the same. Using buffereader, copy the content of file first.txt to another file second.txt

```
import java.io.*;

public class FileCopyExample {
    public static void main(String[] args) {
        String sourceFileName = "try.txt";
        String destinationFileName = "second.txt";

        // Reading content of "try.txt" using FileInputStream and
displaying it
        System.out.println("Content of try.txt:");
        try (FileInputStream fis = new FileInputStream(sourceFileName);
             InputStreamReader isr = new InputStreamReader(fis);
             BufferedReader br = new BufferedReader(isr)) {
            String line;
            while ((line = br.readLine()) != null) {
                System.out.println(line);
            }
        } catch (IOException e) {
            System.err.println("Error reading the file: " +
e.getMessage());
        }

        // Copying content of "first.txt" to "second.txt" using
BufferedReader
        try (BufferedReader reader = new BufferedReader(new
FileReader("first.txt"));
             BufferedWriter writer = new BufferedWriter(new
FileWriter(destinationFileName))) {
```

```
            String line;
            while ((line = reader.readLine()) != null) {
                writer.write(line);
                writer.newLine();
            }
            System.out.println("Content of first.txt copied to second.txt
successfully.");
        } catch (IOException e) {
            System.err.println("Error copying the file: " +
e.getMessage());
        }
    }
}
```

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  c:; cd 'c:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2
Files\Java\jdk-21\bin\java.exe' '@C:\Users\smart\AppData\Local\Temp\cp_5ym5plbgz89v881a65csmag58.argfile' 'FileCopyExam
Content of try.txt:
Nita Dilip Mandge
Durgesh Dilip Mandge
Content of first.txt copied to second.txt successfully.
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```

4. Open the file "even.txt" read the number present in the file. show the square of the number as output. if the number is not present show the exception

```
import java.io.*;

public class NumberSquared {
    public static void main(String[] args) {
        String fileName = "even.txt";

        try (BufferedReader br = new BufferedReader(new
FileReader(fileName))) {
            String line;
            while ((line = br.readLine()) != null) {
                try {
                    int number = Integer.parseInt(line);
                    if (number % 2 == 0) {
                        System.out.println("Square of " + number + ": " +
(number * number));
                    } else {
```

```
                        System.out.println(number + " is not an even
number.");
                    }
                } catch (NumberFormatException e) {
                    System.err.println("Error: The line does not contain a
valid integer: " + line);
                }
            }
        } catch (FileNotFoundException e) {
            System.err.println("Error: File not found: " + fileName);
        } catch (IOException e) {
            System.err.println("Error reading the file: " +
e.getMessage());
        }
    }
}
```

OUTPUT:

even.txt not there

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  & 'C:\Program Files\Java\jdk-21\bi
p_5ym5plbgz89v881a65csmag58.argfile' 'NumberSquared'
Error: File not found: even.txt
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```

even.txt is there

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  c:; cd 'c:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Jav
Files\Java\jdk-21\bin\java.exe' '@C:\Users\smart\AppData\Local\Temp\cp_5ym5plbgz89v881a65csmag58.argfile' 'NumberSquared'
Square of 2: 4
3 is not an even number.
Square of 4: 16
5 is not an even number.
Square of 6: 36
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```

5. using buffereader read the content of the first file. show the output on the console. only thing checks the content. if it is vowel skip that character

```
import java.io.*;

public class SkipVowels {
    public static void main(String[] args) {
```

```
        String fileName = "first.txt";

        try (BufferedReader br = new BufferedReader(new
FileReader(fileName))) {
            int character;
            while ((character = br.read()) != -1) {
                char ch = (char) character;
                if (!isVowel(ch)) {
                    System.out.print(ch);
                }
            }
        } catch (FileNotFoundException e) {
            System.err.println("Error: File not found: " + fileName);
        } catch (IOException e) {
            System.err.println("Error reading the file: " +
e.getMessage());
        }
    }

    // Method to check if a character is a vowel
    private static boolean isVowel(char ch) {
        ch = Character.toLowerCase(ch);
        return ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch ==
'u';
    }
}
```

OUTPUT:

```
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>  & 'C:\Program Files\
p_5ym5plbgz89v881a65csmag58.argfile' 'SkipVowels'
Hll frm frst fl ths s 4th prctcl f jv
PS C:\Users\smart\Documents\SPIT-MCA\SPIT-lab\Sem 2\Java\Lab4>
```