Java File Operations - Code Explanation

Introduction

This Java code demonstrates basic file operations like reading, writing, and modifying files.

The operations include:
1. Writing content to a file.
2. Reading content from a file.
3. Modifying specific content in a file.

The code utilizes BufferedReader for reading and BufferedWriter for writing text to the file.

Java Code

```java
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class FileOperations {

    // Method to write text to a file
    public static void writeToFile(String fileName, String content) {
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(fileName))) {
            writer.write(content);
            System.out.println("Content written to file: " + fileName);
        } catch (IOException e) {
            System.err.println("Error writing to file: " + e.getMessage());
        }
    }

    // Method to read text from a file
    public static String readFromFile(String fileName) {
        StringBuilder content = new StringBuilder();
        try (BufferedReader reader = new BufferedReader(new FileReader(fileName))) {
            String line;
            while ((line = reader.readLine()) != null) {
                content.append(line).append("\n");
            }
            System.out.println("Content read from file: " + fileName);
        } catch (IOException e) {
            System.err.println("Error reading from file: " + e.getMessage());
        }
        return content.toString();
    }
```

Java File Operations - Code Explanation

```java
    // Method to modify text in a file
    public static void modifyFile(String fileName, String oldContent, String newContent)
{
        String fileContent = readFromFile(fileName);
        if (fileContent.contains(oldContent)) {
            fileContent = fileContent.replace(oldContent, newContent);
            writeToFile(fileName, fileContent);
            System.out.println("File modified: " + fileName);
        } else {
            System.out.println("Old content not found in the file.");
        }
    }

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

        // Writing to the file
        writeToFile(fileName, "Hello, World!\nThis is a sample text file.");

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

        // Modifying the file
        modifyFile(fileName, "sample", "modified");

        // Reading the modified file
        String modifiedContent = readFromFile(fileName);
        System.out.println("Modified File Content:\n" + modifiedContent);
    }
}
```