```java
package edu.cpp.cs5800.flyweight;
import static edu.cpp.cs5800.flyweight.CharacterFormat.*;

public class Driver {
    public static void main(String[] args) {
        String docFileName = "document.txt";

        Document document = createDocument();
        System.out.print(document.getContent());

        document.save(docFileName);
        System.out.println(">> Successfully saved the document file: " + docFileName + "\n"
);

        Document loadedDoc = loadDocument(docFileName);
        System.out.println(loadedDoc.getContent());

        editDocument(loadedDoc);
        System.out.println(loadedDoc.getContent());

    }

    private static Document createDocument() {
        System.out.println(">> Creating a new document");
        Document document = new Document();
        document.addCharacter('H', Font.ARIAL, Color.RED, 12);
        document.addCharacter('e', Font.ARIAL, Color.BLACK, 12);
        document.addCharacter('l', Font.CALIBRI, Color.BLUE, 14);
        document.addCharacter('l', Font.CALIBRI, Color.BLUE, 14);
        document.addCharacter('o', Font.VERDANA, Color.BLACK, 16);
        document.addCharacter('W', Font.VERDANA, Color.RED, 16);
        document.addCharacter('o', Font.ARIAL, Color.RED, 12);
        document.addCharacter('r', Font.CALIBRI, Color.BLUE, 14);
        document.addCharacter('l', Font.VERDANA, Color.BLACK, 16);
        document.addCharacter('d', Font.ARIAL, Color.BLACK, 12);
        document.addCharacter('C', Font.CALIBRI, Color.BLUE, 14);
        document.addCharacter('S', Font.VERDANA, Color.BLACK, 16);
        document.addCharacter('5', Font.ARIAL, Color.RED, 12);
        document.addCharacter('8', Font.CALIBRI, Color.BLUE, 14);
        document.addCharacter('0', Font.VERDANA, Color.RED, 16);
        document.addCharacter('0', Font.ARIAL, Color.BLACK, 12);
        return document;
    }

    private static Document loadDocument(String fileName) {
        System.out.println(">> Loading the document from file: " + fileName);
        Document document = new Document();
        document.load(fileName);
        return document;
    }

    private static void editDocument(Document document) {
        System.out.println(">> Editing document");
        document.edit(1, 'E', Font.CALIBRI, Color.BLUE, 16);
        document.edit(2, 'L', Font.ARIAL, Color.BLUE, 12);
        document.edit(3, 'L', Font.ARIAL, Color.BLACK, 14);
        document.edit(4, 'O', Font.VERDANA, Color.RED, 12);
    }
}

```

```java
package edu.cpp.cs5800.flyweight;

import java.io.*;
import java.util.ArrayList;
import java.util.List;

public class Document {
    private List<Character> characters = new ArrayList<>();

    public boolean load(String path) {
        characters.clear();
        try {
            BufferedReader reader = new BufferedReader(new FileReader(path));
            String line;
            while ((line = reader.readLine()) != null) {
                Character ch = parseCharacter(line);
                if (ch != null) {
                    this.addCharacter(ch);
                }
            }
        } catch (IOException e) {
            return false;
        }

        return true;
    }

    public boolean addCharacter(char symbol,
                                CharacterFormat.Font font,
                                CharacterFormat.Color color,
                                int size) {
        Character ch = CharacterFactory.getCharacter(symbol, font, color, size);
        return characters.add(ch);
    }

    public boolean addCharacter(Character character) {
        return characters.add(character);
    }

    public Character edit(int index,
                          char symbol,
                          CharacterFormat.Font font,
                          CharacterFormat.Color color,
                          int size) {
        Character ch = CharacterFactory.getCharacter(symbol, font, color, size);
        return characters.set(index, ch);
    }

    public boolean save(String path) {
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(path))) {
            for (Character character : characters) {
                writer.write(character.getSymbol() + "," + character.getFormat()
                        .getFont() + "," + character.getFormat()
                        .getColor() + "," + character.getFormat().getFontSize());
                writer.newLine();
            }
        } catch (IOException e) {
            return false;
        }
        return true;
    }

    public String getContent() {
        StringBuilder content = new StringBuilder();
        for (Character character : characters) {
            content.append(character);
            content.append('\n');
        }
        return content.toString();
    }

    private Character parseCharacter(String content) {
        String[] parts = content.split(",");
        if (parts.length == 4) {
            char symbol = parts[0].charAt(0);
            CharacterFormat.Font font = CharacterFormat.Font.valueOf(parts[1]);
            CharacterFormat.Color color = CharacterFormat.Color.valueOf(parts[2]);
            int size = Integer.parseInt(parts[3]);
            return CharacterFactory.getCharacter(symbol, font, color, size);
        } else {
            return null;
        }
    }
}

```

```java
package edu.cpp.cs5800.flyweight;

public class Character {
    private final char symbol;
    private final CharacterFormat format;

    public Character(char symbol, CharacterFormat format) {
        this.symbol = symbol;
        this.format = format;
    }

    public char getSymbol() {
        return symbol;
    }
    public CharacterFormat getFormat() {
        return format;
    }

    @Override
    public String toString() {
        return symbol + "(" + format + ")";
    }
}

```

```java
package edu.cpp.cs5800.flyweight;

public class CharacterFormat {
    public enum Color {
        RED, BLACK, BLUE
    }

    public enum Font {
        ARIAL, CALIBRI, VERDANA
    }

    private final Color color;
    private final Font font;
    private final int fontSize;

    public CharacterFormat(Font font, Color color, int fontSize) {
        this.color = color;
        this.font = font;
        this.fontSize = fontSize;
    }

    public Color getColor() {
        return color;
    }

    public Font getFont() {
        return font;
    }

    public int getFontSize() {
        return fontSize;
    }

    @Override
    public String toString() {
        return font + "," + color + "," + fontSize;
    }
}
```

```java
package edu.cpp.cs5800.flyweight;

import java.util.HashMap;
import java.util.Map;

public class CharacterFactory {
    private static Map<String, CharacterFormat> formats = new HashMap<>();

    public static Character getCharacter(char symbol, CharacterFormat.Font font, CharacterFormat.Color color, int fontSize) {
        String formatType = font.toString() + color.toString() + fontSize;
        CharacterFormat format = formats.get(formatType);
        if (format == null) {
            format = new CharacterFormat(font, color, fontSize);
        }

        return new Character(symbol, format);
    }
}
```