Practice-4.2

Code:

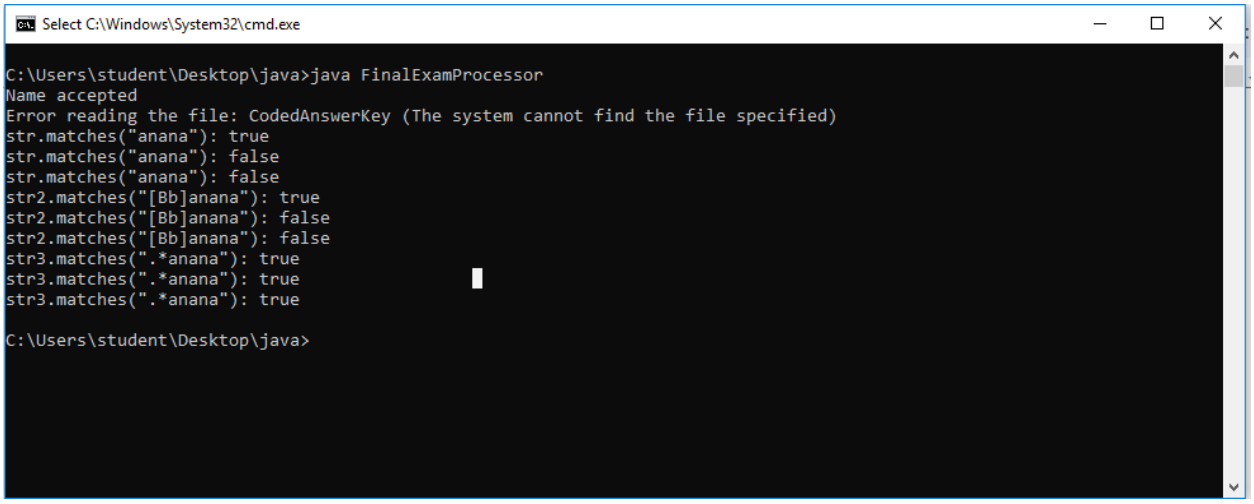
```
import java.util.regex.*;
import java.io.*;

public class FinalExamProcessor {

    // Method to validate the name format
    public static void validateName(String input) {
        // Regular expression to validate name format: "Firstname Lastname"
        String regex = "^[\\p{L}]+\\s[\\p{L}]+$";
        Pattern pattern = Pattern.compile(regex);
        Matcher matcher = pattern.matcher(input);

        if (!matcher.matches()) {
            System.out.println("Incorrect format for name");
        } else {
            System.out.println("Name accepted");
        }
    }

    // Method to process the coded answer key
    public static String processAnswerKey(String filename) throws
IOException {
        BufferedReader codedAnswers = new BufferedReader(new
FileReader(filename));
        String line;
        StringBuilder answers = new StringBuilder();

        // Regular expression to match valid answer choices
        Pattern pattern = Pattern.compile("^[a-dA-D]$");

        while ((line = codedAnswers.readLine()) != null) {
            Matcher matcher = pattern.matcher(line);
```

```
            if (matcher.matches()) {
                answers.append(line);
            }
        }

        codedAnswers.close();
        return answers.toString();
    }

    // Method to finalize answers based on teacher's instructions
    public static String finalAnswers(String answers) {
        // Replace characters as per teacher's instructions
        String finalAnswers = answers.replaceAll("e", "b")
                                     .replaceAll("E", "A")
                                     .replaceAll("f", "c")
                                     .replaceAll("F", "D");
        // Convert the string to lowercase
        return finalAnswers.toLowerCase();
    }

    // Method to test regular expressions
    public static void testRegex() {
        // Testing "?anana"
        String str = "anana";
        System.out.println("str.matches(\"anana\"): " + str.matches("anana"));
// true
        str = "banana";
        System.out.println("str.matches(\"anana\"): " + str.matches("anana"));
// false
        str = "gabanana";
        System.out.println("str.matches(\"anana\"): " + str.matches("anana"));
// false

        // Testing "[Bb]anana"
        String str2 = "banana";
```

```
        System.out.println("str2.matches(\"[Bb]anana\"): " +
str2.matches("[Bb]anana")); // true
        str2 = "anana";
        System.out.println("str2.matches(\"[Bb]anana\"): " +
str2.matches("[Bb]anana")); // false
        str2 = "shanana";
        System.out.println("str2.matches(\"[Bb]anana\"): " +
str2.matches("[Bb]anana")); // false

        // Testing ".*anana"
        String str3 = "montanana";
        System.out.println("str3.matches(\".*anana\"): " +
str3.matches(".*anana")); // true
        str3 = "anana";
        System.out.println("str3.matches(\".*anana\"): " +
str3.matches(".*anana")); // true
        str3 = "_anana";
        System.out.println("str3.matches(\".*anana\"): " +
str3.matches(".*anana")); // true
    }

    public static void main(String[] args) {
        // Example of validating a name
        String name = "John Doe";
        validateName(name); // Output: Name accepted or Incorrect format for
name

        try {
            // Process the coded answer key
            String filename = "CodedAnswerKey";

            // Uncomment the following block to create a mock file if needed
            /*
            BufferedWriter writer = new BufferedWriter(new
FileWriter(filename));
```

```
            writer.write("A\n");
            writer.write("b\n");
            writer.write("C\n");
            writer.write("D\n");
            writer.write("x\n"); // This line should not be included in the final
answer
            writer.close();
            */

            String answers = processAnswerKey(filename);
            System.out.println("Decoded answers: " + answers);

            // Finalize the answers
            String finalAnswerString = finalAnswers(answers);
            System.out.println("Final answers: " + finalAnswerString);

        } catch (IOException e) {
            System.out.println("Error reading the file: " + e.getMessage());
        }

        // Test regular expressions
        testRegex();
    }
}
```

Output:


Select C:\Windows\System32\cmd.exe
C:\Users\student\Desktop\java>java FinalExamProcessor
Name accepted
Error reading the file: CodedAnswerKey (The system cannot find the file specified)
str.matches("anana"): true
str.matches("anana"): false
str.matches("anana"): false
str2.matches("[Bb]anana"): true
str2.matches("[Bb]anana"): false
str2.matches("[Bb]anana"): false
str3.matches(".*anana"): true
str3.matches(".*anana"): true
str3.matches(".*anana"): true
C:\Users\student\Desktop\java>