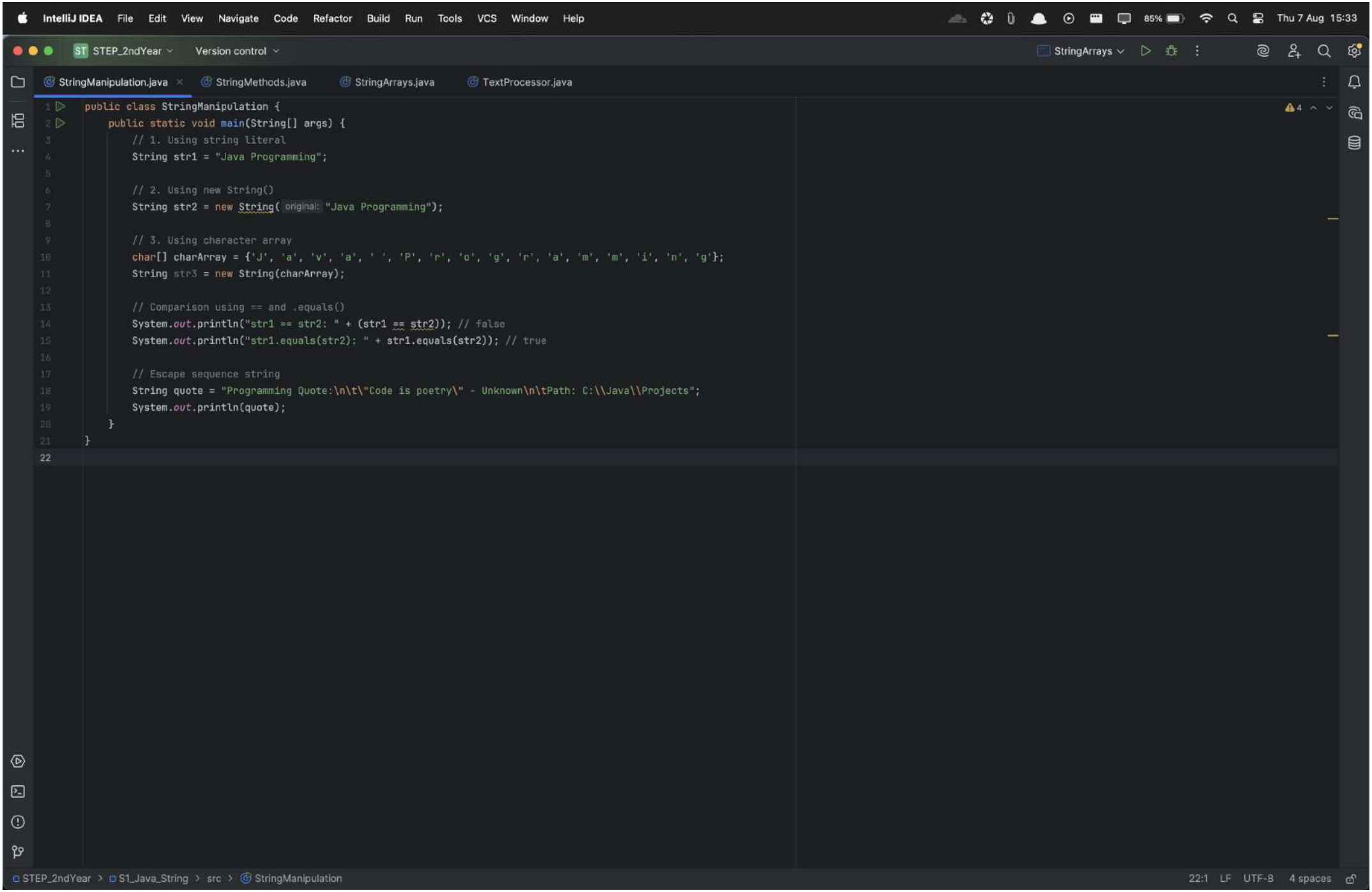
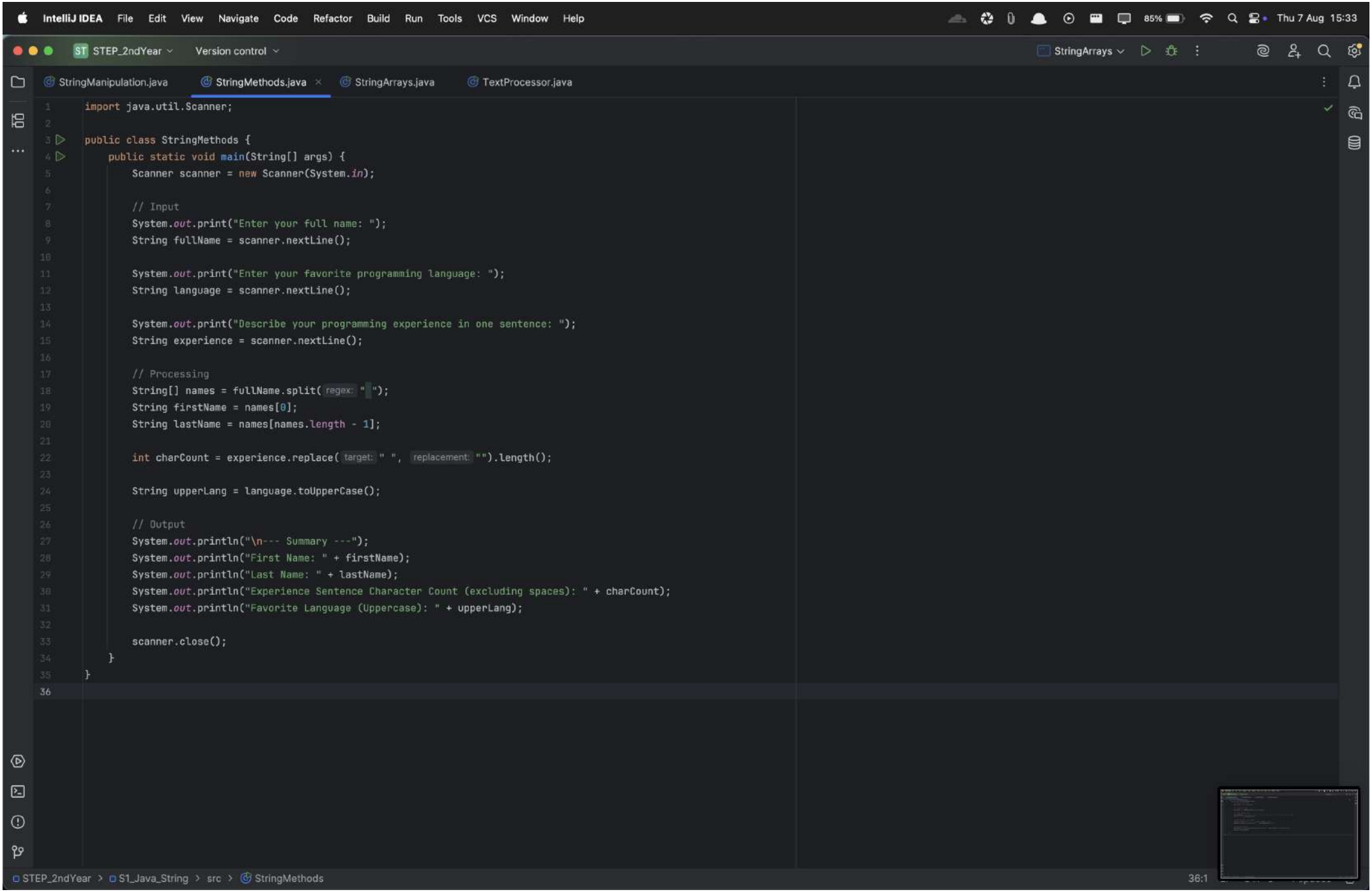
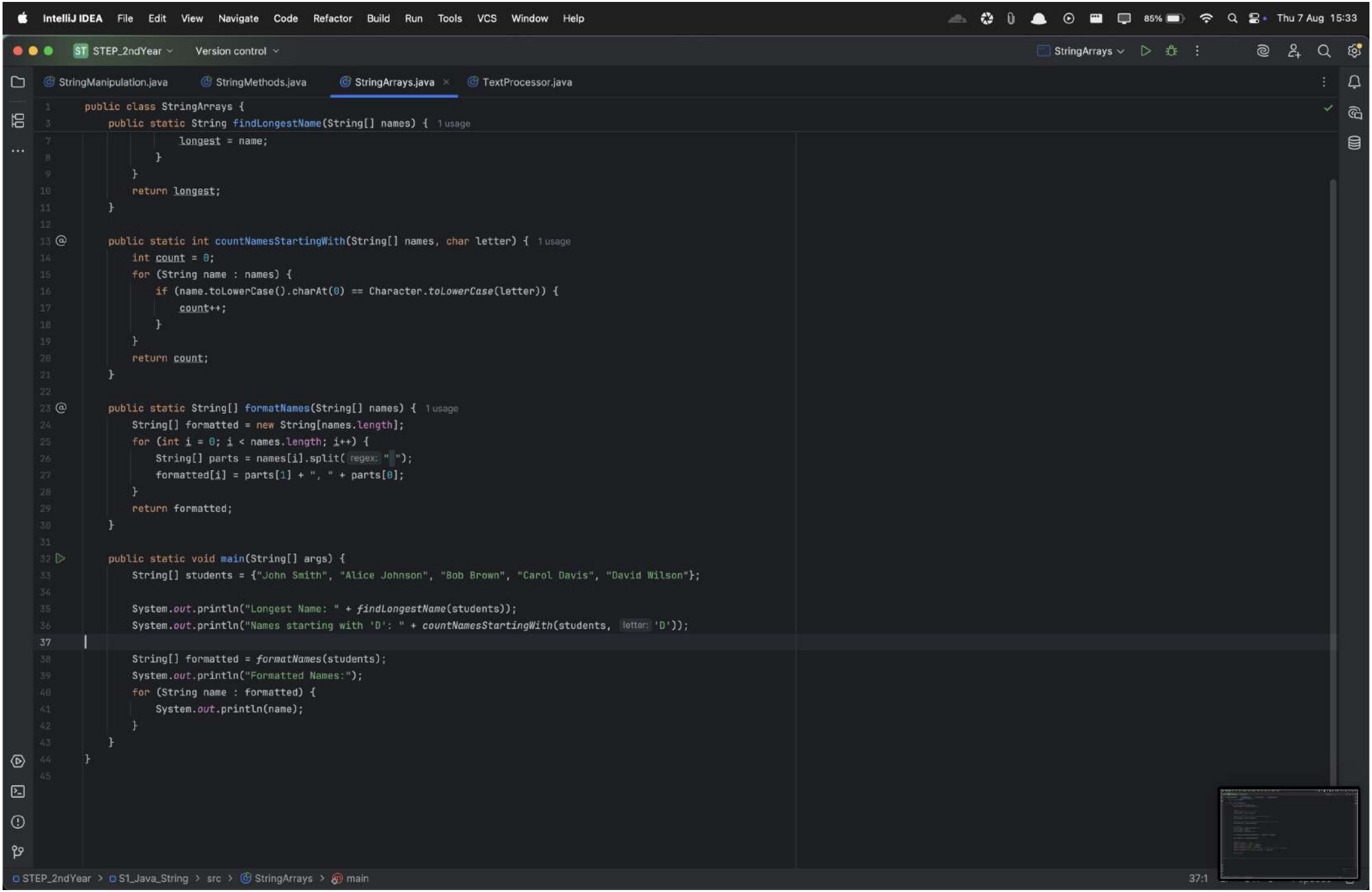
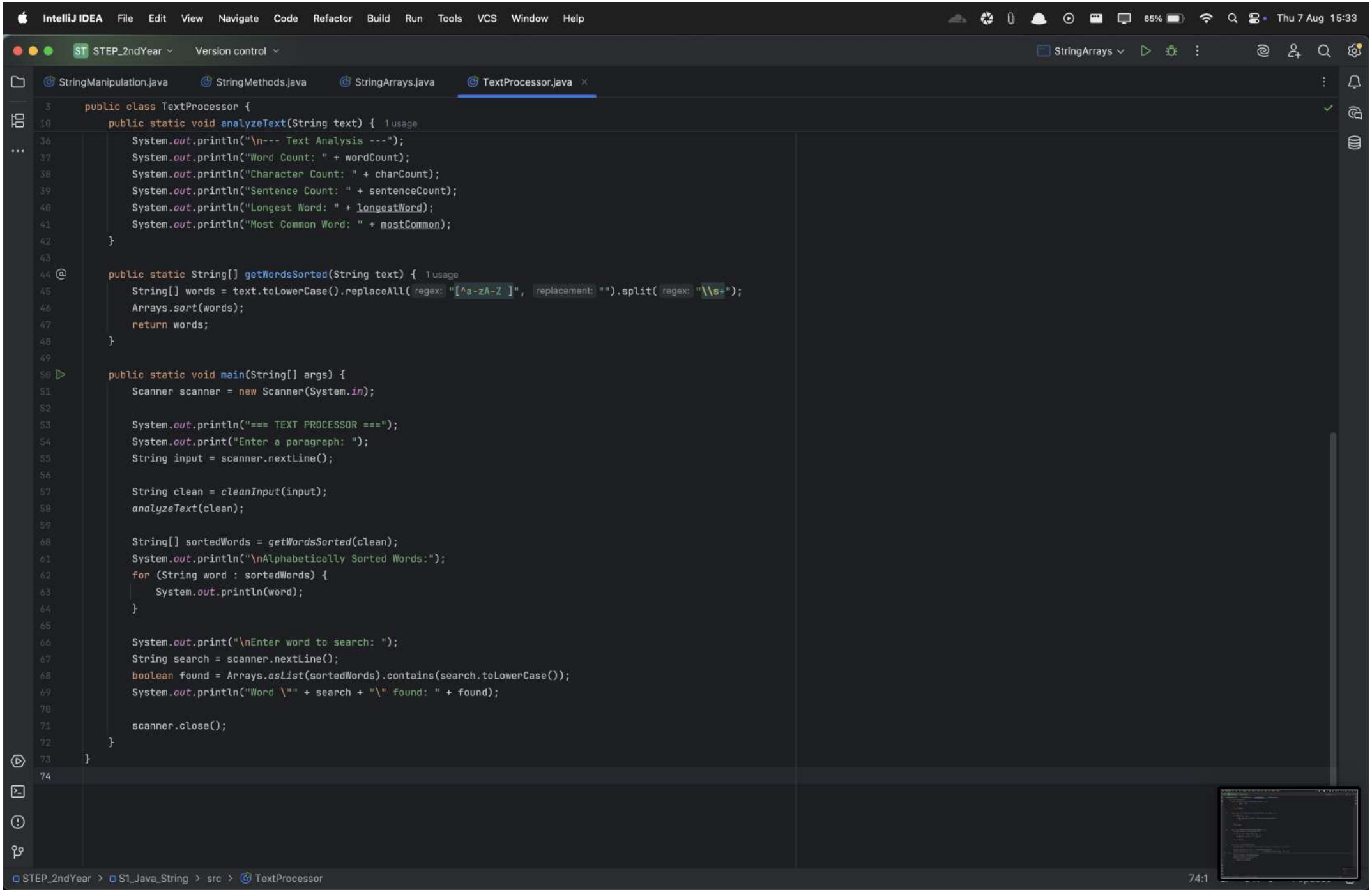

IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu 7 Aug 15:33
STEP_2ndYear
Version control
StringArrays
StringManipulation.java
StringMethods.java
StringArrays.java
TextProcessor.java
public class StringManipulation {
public static void main(String[] args) {
// 1. Using string literal
String str1 = "Java Programming";
// 2. Using new String()
String str2 = new String(original: "Java Programming");
// 3. Using character array
char[] charArray = {'J', 'a', 'v', 'a', ' ', 'P', 'r', 'o', 'g', 'r', 'a', 'm', 'm', 'i', 'n', 'g'};
String str3 = new String(charArray);
// Comparison using == and .equals()
System.out.println("str1 == str2: " + (str1 == str2)); // false
System.out.println("str1.equals(str2): " + str1.equals(str2)); // true
// Escape sequence string
String quote = "Programming Quote:\n\t\"Code is poetry\" - Unknown\n\tPath: C:\\Java\\Projects";
System.out.println(quote);
}
}
STEP_2ndYear > S1_Java_String > src > StringManipulation
22:1 LF UTF-8 4 spaces


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu 7 Aug 15:33
STEP_2ndYear
Version control
StringArrays
StringManipulation.java
StringMethods.java
StringArrays.java
TextProcessor.java
import java.util.Scanner;
public class StringMethods {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
// Input
System.out.print("Enter your full name: ");
String fullName = scanner.nextLine();
System.out.print("Enter your favorite programming language: ");
String language = scanner.nextLine();
System.out.print("Describe your programming experience in one sentence: ");
String experience = scanner.nextLine();
// Processing
String[] names = fullName.split(regex: " ");
String firstName = names[0];
String lastName = names[names.length - 1];
int charCount = experience.replace(target: " ", replacement: "").length();
String upperLang = language.toUpperCase();
// Output
System.out.println("\n--- Summary ---");
System.out.println("First Name: " + firstName);
System.out.println("Last Name: " + lastName);
System.out.println("Experience Sentence Character Count (excluding spaces): " + charCount);
System.out.println("Favorite Language (Uppercase): " + upperLang);
scanner.close();
}
}
STEP_2ndYear > S1_Java_String > src > StringMethods
36:1


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu 7 Aug 15:33
STEP_2ndYear
Version control
StringArrays
StringManipulation.java
StringMethods.java
StringArrays.java
TextProcessor.java
public class StringArrays {
public static String findLongestName(String[] names) { 1 usage
longest = name;
}
}
return longest;
}
public static int countNamesStartingWith(String[] names, char letter) { 1 usage
int count = 0;
for (String name : names) {
if (name.toLowerCase().charAt(0) == Character.toLowerCase(letter)) {
count++;
}
}
return count;
}
public static String[] formatNames(String[] names) { 1 usage
String[] formatted = new String[names.length];
for (int i = 0; i < names.length; i++) {
String[] parts = names[i].split(regex: " ");
formatted[i] = parts[1] + ", " + parts[0];
}
return formatted;
}
public static void main(String[] args) {
String[] students = {"John Smith", "Alice Johnson", "Bob Brown", "Carol Davis", "David Wilson"};
System.out.println("Longest Name: " + findLongestName(students));
System.out.println("Names starting with 'D': " + countNamesStartingWith(students, letter: 'D'));
String[] formatted = formatNames(students);
System.out.println("Formatted Names:");
for (String name : formatted) {
System.out.println(name);
}
}
}
STEP_2ndYear > S1_Java_String > src > StringArrays > main
37:1


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu 7 Aug 15:33
STEP_2ndYear
Version control
StringArrays
StringManipulation.java
StringMethods.java
StringArrays.java
TextProcessor.java
public class TextProcessor {
public static void analyzeText(String text) { 1 usage
System.out.println("\n--- Text Analysis ---");
System.out.println("Word Count: " + wordCount);
System.out.println("Character Count: " + charCount);
System.out.println("Sentence Count: " + sentenceCount);
System.out.println("Longest Word: " + longestWord);
System.out.println("Most Common Word: " + mostCommon);
}
public static String[] getWordsSorted(String text) { 1 usage
String[] words = text.toLowerCase().replaceAll(regex: "[^a-zA-Z]", replacement: "").split(regex: "\\s+");
Arrays.sort(words);
return words;
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("=== TEXT PROCESSOR ===");
System.out.print("Enter a paragraph: ");
String input = scanner.nextLine();
String clean = cleanInput(input);
analyzeText(clean);
String[] sortedWords = getWordsSorted(clean);
System.out.println("\nAlphabetically Sorted Words:");
for (String word : sortedWords) {
System.out.println(word);
}
System.out.print("\nEnter word to search: ");
String search = scanner.nextLine();
boolean found = Arrays.asList(sortedWords).contains(search.toLowerCase());
System.out.println("Word \"" + search + "\" found: " + found);
scanner.close();
}
}
STEP_2ndYear > S1_Java_String > src > TextProcessor
74:1