**Advanced workbook**

Aufgabenstellung zum Kurs:

DLBCSDSJCL02\_D – Datenstruktur und Java-Klassenbibliotheken

**Bitte überprüfe die Gültigkeit dieser Workbookaufgaben im Kurs auf myCampus.**

**Hinweis zum Urheberrecht und zur Plagiatsprüfung:**

**Es wird darauf hingewiesen, dass der IU Internationale Hochschule GmbH das Urheberrecht der Prüfungsaufgaben/Aufgabenstellungen obliegt. Einer Veröffentlichung der Aufgabenstellungen auf Drittplattformen wird ausdrücklich widersprochen. Im Falle einer Zuwiderhandlung stehen der Hochschule u.a. Unterlassungsansprüche zu. Zudem weisen wir darauf hin, dass jede eingereichte schriftliche Ausarbeitung mittels einer Plagiatssoftware überprüft wird. Wir empfehlen daher auch, keinesfalls ausgearbeitete Lösungen zu teilen, da dies den Verdacht eines Plagiates begründen kann.**

****Aufgabenstellung 1:****

Angenommen, Du möchtest eine Anwendung entwickeln, die Informationen über Dateien in einem bestimmten Verzeichnis sammelt und analysiert. Verwende die File-Klasse in Java, um die folgenden Aufgaben zu lösen:

* Auflistung aller in einem Verzeichnis enthaltenen Dateien: Schreibe eine Methode, die den Pfad zu einem Verzeichnis als Eingabe akzeptiert und alle enthaltenen Dateien auflistet. Verwende dazu die Methoden der File-Klasse, um das Verzeichnis zu öffnen und die Liste der Dateien abzurufen. Gib die Dateinamen auf der Konsole aus.
* Überprüfung der Größe der Dateien in dem Verzeichnis: Erweitere Deine Methode, um die Größe jeder Datei in dem Verzeichnis zu überprüfen. Verwende dazu die length()-Methode der File-Klasse, um die Größe jeder Datei abzurufen. Summiere die Größen aller Dateien und gib die Gesamtgröße auf der Konsole aus.
* Berechnung der Gesamtgröße des Verzeichnisses: Modifiziere Deine Methode, um die Größe des gesamten Verzeichnisses zu berechnen. Durchlaufe rekursiv alle Unterverzeichnisse und summiere die Größen aller Dateien in jedem Unterverzeichnis auf. Gib die Gesamtgröße des Verzeichnisses auf der Konsole aus.

Erkläre ausführlich, wie Du die File-Klasse verwendest, um die oben genannten Aufgaben zu lösen. Diskutiere die Methoden und Funktionen der File-Klasse, die für diese Aufgaben relevant sind, und gib die Codebeispiele für die Implementierung der einzelnen Schritte an. Erörtere mögliche Herausforderungen bei der Verwendung der File-Klasse und stelle Lösungsansätze vor, um diese Herausforderungen zu bewältigen.

****Aufgabenstellung 2:****

Angenommen, Du entwickelst eine Anwendung in Java, die es Benutzern ermöglicht, Text in eine Datei zu schreiben: Entwirf eine Methode namens "writeToFile", die einen Text (als String) und einen Dateinamen (als String) als Parameter akzeptiert und den Text in die angegebene Datei schreibt. Stelle sicher, dass die Methode die erforderlichen Dateioperationen durchführt, wie das Erstellen der Datei, falls sie nicht existiert, das Überschreiben der Datei, falls sie bereits existiert, und das Schließen der Datei nach dem Schreiben.

Implementiere anschließend eine Beispielanwendung, in der Du die "writeToFile"-Methode verwendest, um eine Benutzereingabe in eine Textdatei zu schreiben. Lies den Text von der Konsole ein und fordere den Benutzer auf, einen Dateinamen einzugeben. Rufe dann die "writeToFile"-Methode auf, um den Text in die angegebene Datei zu schreiben.

****Aufgabenstellung 3:****

**In Deinem Unternehmen, welches auf die Softwareentwicklung in Java fokussiert ist, fing am vergangenen Monatsersten ein Praktikant (Student der Informatik) an. Nachdem er sein erstes „kleines“ Entwicklungsprojekt abgeschlossen hat, fällt Dir auf, dass sein Code etwas schwer lesbar ist und Du befürchtest, dass es Deinen Kollegen ähnlich ergehen könnte. Um eine effiziente und produktive Zusammenarbeit Deines Teams sicherstellen zu können, möchtest Du Deinem Praktikanten einige Hinweise zur Codekonvention geben. Welche fünf besonders bedeutenden Code-Konventionen könntest Du ihm mit auf den Weg gehen? Wie kannst Du ihm die fünf Konventionen möglichst anschaulich näherbringen?**

****Aufgabenstellung 4:****

Angenommen, Du entwickelst eine Anwendung, die einen Texteditor implementiert. Eine der Funktionen des Texteditors besteht darin, dem Benutzer zu ermöglichen, Teilstrings in einem Dokument zu suchen und zu vergleichen. Erkläre, wie Du in Java Teilstring-Vergleiche implementieren würdest und welche Funktionen und Klassen Du verwenden würdest, um diese Aufgabe zu lösen.

Beschreibe zunächst, wie Du einen Teilstring in einem gegebenen Text finden würdest. Diskutiere verschiedene Ansätze wie die Verwendung der String-Klasse und der Pattern-Klasse aus dem java.util.regex-Paket, um Teilstring-Matching zu ermöglichen. Erkläre, wie reguläre Ausdrücke verwendet werden können, um komplexe Suchmuster zu definieren.

Diskutiere dann die verschiedenen Möglichkeiten, Teilstrings zu vergleichen. Erkläre die Verwendung von Methoden wie equals(), equalsIgnoreCase(), startsWith(), endsWith() usw. der String-Klasse, um Teilstrings zu vergleichen und zu überprüfen, ob sie bestimmte Eigenschaften haben.

Gib anschließend Beispiele für die Verwendung dieser Teilstring-Vergleichsfunktionen in einem Texteditor. Beschreibe, wie Du eine Suchfunktion implementieren würdest, um den Benutzern das Finden von Teilstrings zu ermöglichen, und wie Du die Vergleichsfunktionen verwenden würdest, um bestimmte Muster oder Eigenschaften von Teilstrings zu überprüfen.

****Aufgabenstellung 5:****

**Erkläre drei Pakete der Java-Klassenbibliothek jeweils mittels eines selbst gewählten Beispiels.**

****Aufgabenstellung 6:****

Du arbeitest an der Entwicklung eines Flugbuchungssystems und musst Strings in Java effizient manipulieren, insbesondere beim Einlesen und Verarbeiten von Daten aus CSV-Dateien. Beschreibe, wie Du String-Manipulationstechniken und den Einsatz eines StringBuffers nutzen kannst, um die Daten aus den CSV-Dateien zu analysieren und die erforderlichen Informationen zu extrahieren.

Angenommen, Du hast eine CSV-Datei mit Flugdaten, in der jeder Datensatz Informationen wie Flugnummer, Abflugort, Zielort und Flugdatum enthält. Erläutere, wie Du den CSV-Text einlesen und analysieren würdest, um die Flugdaten effizient zu extrahieren und in Ihrem Flugbuchungssystem zu verwenden.

Erkläre, wie Du den CSV-Text mithilfe der String-Manipulationstechniken in Java verarbeiten würdest. Beschreibe den Prozess des Aufteilens des CSV-Texts in einzelne Datensätze und die Extraktion der relevanten Informationen. Erörtere auch, wie Du den StringBuffers verwenden könntest, um effizient Strings zu manipulieren und zu modifizieren, insbesondere bei der Zusammenstellung von Flugdetails oder der Erstellung von Berichten.

Gib konkrete Codebeispiele, wie Du die oben genannten Techniken in Java anwenden würdest, um den CSV-Text einzulesen, die Daten zu analysieren und die relevanten Informationen zu extrahieren. Erläutere auch, wie Du die String-Manipulationsmethoden wie split(), substring(), trim() und die Methoden des StringBuffers wie append() und toString() verwenden würdest.

Diskutiere die Vorteile der Verwendung von String-Manipulationstechniken und des StringBuffers bei der Verarbeitung von CSV-Daten. Gehe auf Aspekte wie Speichereffizienz, Leistung und Codelesbarkeit ein. Erwähne auch mögliche Herausforderungen und Best Practices bei der Verarbeitung von Strings und der Handhabung von CSV-Dateien in Java.