

Aufgabe 1: Bubble Sort [20 Punkte]

.... Aus der Vorlesung / Vortrag ...

Aufgabe 2: Stream [10 Punkte]

- a.) Erstellen Sie mit Java folgende Ordner-Struktur: /log/uebungsklausur/ [2]
- b.) Instanziiieren Sie die Klasse "FileWriter" zur Ausgabe von Text in eine Datei namens "meine_log_datei.txt"! [2]
- c.) Kapseln Sie die Instanz in einem neu erzeugten Objekt des Types "PrintWriter"! [2]
- d.) Schreiben Sie folgenden Text in die Datei hinein [2]:
Error Log:
- e.) Fügen Sie der Datei drei folgendermaßen formatierte Einträge hinzu [2]:
Error 500: Internal Error. 15.06.2022, 10:35 Uhr
Bestimmen Sie dabei die aktuelle Zeit durch eine geeignete Methode.

Aufgabe 3: Erstellen und suchen einer Person [20 Punkte]

- a.) Erstellen Sie eine Klasse namens „Kontakt“ mit den drei privaten Zeichenkettenattributen „name“, „strasse“, „ort“! [2]
- b.) Fügen Sie ein passendes Laufkriterium ein! (Hinweis: Bedingung der for-Schleife) [2]
- c.) Erstellen Sie einen initialisierenden Konstruktor, der alle drei Attribute berücksichtigt! [2]
- d.) Erstellen Sie öffentliche Zugriffsmethoden für alle drei Attribute! [2]
- e.) Überschreiben Sie die geerbte „toString“-Methode! Lassen Sie sie eine vernünftig formatierte Ausgabe aller drei Attributwerte zurückliefern! [2]
- f.) Erzeugen Sie in der neu zu erstellenden Startklasse ein Feld (Array), das fünf „Person“- Objekte speichern kann! [2]
- g.) Weisen Sie ihm fünf „Person“-Objekte zu! Nutzen Sie dafür den initialisierenden Konstruktor! [2]
- h.) Erstellen Sie eine statische Methode „print“, der ein „Person“-Feld (Array) als Parameter übergeben wird! Verwenden Sie darin eine „for-each“-Schleife, um alle „Person“-Objekte auf der Konsole auszugeben! [2]
- i.) Erstellen Sie eine weitere statische Methode „search“, die neben dem „Person“-Feld (Array) auch eine Such-Zeichenkette entgegennimmt! [2]
- j.) Durchsuchen Sie darin die Namen sämtlicher „Person“-Objekte per Zählschleife! Geben Sie den Index eines gefundenen Objektes als Rückgabewert zurück! [2] (Hinweis: Methode „contains“)
- k.) Testen Sie die Methoden „print“ und „search“ durch Aufruf in der „main“-Methode! [2]

Aufgabe 4: Object Serialisation [20 Punkte]

- a.) Erstellen Sie eine Klasse „Person“ mit den Attributen „vorname“, „nachname“, „wohnort“ vom Typ „String“ und fügen Sie entsprechende getter- und setter-Methoden ein. [4]
- b.) Die Klasse soll serialisierbar sein und über eine eigene UID verfügen. [2]
- c.) Der Konstruktor berücksichtigt alle drei Attribute und weist die Werte entsprechend zu. [2]
- d.) Erstellen Sie zwei Objekte des Types "Person"! Geben Sie alle Informationen beider Personen über die Konsole aus! [4]
- e.) Speichern Sie die zuvor erstellten Objekte in eine Datei (z.B. personen.ser). [4]
- f.) Lesen Sie die Objekte aus der Datei wieder in das Programm ein und speichern Sie sie in jeweils neuen Objektvariablen! [2]

g.) Geben Sie alle Informationen der eingelesenen Personen über die Konsole aus! [2]

Aufgabe 5: XML Prozessierung [30 Punkte]

Gegeben sind zwei Klassen:

Studierender:

privat String: name, jahrgang, nummer

StudierendenListe:

privat Liste: studListe

privat String: seminargruppe, studiengang

- a.) Erstellen Sie die Klassen mit Attributen und den entsprechenden Zugriffsmethoden. [2]
- b.) Erstellen Sie eine weitere Klasse (Start) mit einer „main“-Methode [2]
- c.) Erzeugen Sie eine Variable, die mehrere Studierender-Objekte enthalten kann. [2]
- d.) Erzeugen Sie drei Studierender-Objekte mit Werten Ihrer Wahl. [2]
- e.) Fügen Sie alle Studierenden in den in c.) erzeugten Behälter hinzu. [2]
- f.) Erzeugen Sie ein Objekt der Klasse „StudierendenListe“ und fügen Sie alle Studierender-Objekte mit einer entsprechenden Methode dem Behälter-Element (studListe) hinzu. [2]
- g.) Fügen Sie in Ihrer Start-Klasse (b.) eine weitere Methode mit dem Namen „converttoxml“ hinzu. Diese Methode soll Ihre StudierendenListe verarbeiten können. [2]
- h.) Erzeugen Sie eine Datei „studierendenliste.xml“ und speichern Sie dort in Form einer XML-Struktur Ihre übergebene StudierendenListe ab. [6]
- i.) Fügen Sie ein geeignetes Exception-Handling in Ihrer converttoxml-Methode ein. [2]
- j.) Rufen Sie in Ihrer main-Methode die converttoxml-Methode auf und testen Sie Ihre Anwendung. [2]
- k.) Fügen Sie die passenden XMLRoot-Element Bezeichnungen in den Klassen Studierender und StudierendenListe ein. [4]
- l.) Im XML-Dokument soll das Attribut name (Studierender-Klasse) als XML-Tag <nachname> ausgegeben werden. Das Attribut selbst darf nicht geändert werden! [2]