

Übung 8 – SE Grundlagen

Aufgabe 1 (0 Punkt):

- Erstellen Sie eine neue Datei „test.txt“ auf Ihrer Festplatte (in einem Ordner mit Schreibrechten).
- Erstellen Sie ein neues Objekt vom Typ `java.io.File` und laden Sie Ihre Datei `test.txt`.
- Erstellen Sie einen `FileReader` und wrappen Sie diesen mittels eines `BufferedReader`.
- Lesen sie mittels der `readLine()` Methode des `BufferedReader` die einzelnen Zeile in eine `ArrayList <String>` ein.
- Geben Sie die `ArrayList<String>` aus (for-each Schleife).

Aufgabe 2 (0 Punkt):

- Recherchieren Sie, wieso man `Reader` / `Streams` immer am Ende mit der `close()` Methode beenden muss.
- Implementieren Sie die `close()` Methode für die beiden `Reader` aus Aufgabe 1.

Aufgabe 3 (0 Punkt):

- Schreiben Sie eine Datei `test2.txt` auf Ihre Festplatte.
- Nutzen Sie hierfür die `FileWriter` Klasse.
- Schreiben Sie die Zahlen von 1 bis 5 in Ihre Textdatei (eine Zahl pro Zeile).
- Mittels `System.getProperty("line.separator");` erhalten Sie einen String der immer einen Linebreak erzeugt.
- Schließen Sie den `FileWriter` ab.

Aufgabe 4 (0 Punkt):

- Lesen Sie vom Nutzer ein Verzeichnis ein (z.B.: `C:\testfolder`)
- Lesen Sie mittels der `list()` Methode von `java.io.File` die einzelnen Dateien des Ordners ein und geben Sie den Text analog zu Aufgabe 1 aus.

Aufgabe 5 (0 Punkt):

- Beheben Sie alle Schwachstellen in Ihrem Programm, so dass falsche Eingaben oder sonstige Fehler nicht zu einem Absturz Ihres Programmes führen.

Leistungsbeurteilung

Aufgabe 1 (1 Punkt):

- Erstellen Sie eine Methode, die dem Nutzer ein Geheimnis verrät.
- Um das Geheimnis zu erfahren, muss der Nutzer sich einloggen.
- Wenn der Nutzer bereits einen Account hat, kann er sich einloggen, andernfalls muss er sich registrieren.
- Bei der Registrierung darf der Benutzer sich einen Benutzernamen und ein Passwort aussuchen, dieses wird auf dem PC in einer Textdatei gespeichert.
- Wenn der Benutzer sich einloggen will, muss er einen Benutzernamen und ein Passwort eingeben. Wenn diese Kombination in der Passwortdatei vorhanden ist erfährt er das Geheimnis, andernfalls beginnt das Programm von vorne.

Aufgabe 2 (1 Punkt):

- Erstellen Sie eine Methode, mittels der ein String verschlüsselt werden kann.
 - Wie die Verschlüsselung aussieht, liegt beim Entwickler.
- Die Methode muss für einen konkreten String immer denselben verschlüsselten Wert liefern!
- Aus dem verschlüsselten Text soll sich das Passwort nicht erraten lassen.
- Implementieren Sie eine einfache (eigene) Verschlüsselung und nutzen Sie diese für Ihr Geheimnis Programm.

Aufgabe 3 (1 Punkt):

- Erstellen Sie ein Logfile für Ihre Datei. In dieser Datei soll jeder Login Versuch dokumentiert werden.
- Es soll pro Login Versuch eine Zeile genutzt werden. In dieser soll stehen:
- Datum
 - Uhrzeit
 - Benutzername
 - Die ersten beiden und die letzten beiden Zeichen des verwendeten Passworts, sowie dazwischen 3x das Zeichen '#'
 - True oder false, je nachdem, ob der Versuch erfolgreich war (true) oder nicht (False)
 - Z.B.: 01.01.2014 13:37 erhartermichael le###or false

Aufgabe 4 (1 Punkt):

- Schreiben Sie eine Methode, mittels der man die Anzahl der nicht erfolgreichen Login Versuche auslesen kann.

Aufgabe 5 (1 Punkt):

- Schreiben Sie ein Programm, das aus einer Textdatei Integer Zahlen zwischen 1 und 50 einliest. Diese Zahlen können häufiger auftreten.
- Erstellen Sie dann eine Methode, der Sie eine gesuchte Zahl n übergeben.
- Die Methode soll ausgeben, wie oft die Zahl n in der eingelesenen Datei vorhanden ist.