

Nachdenkzettel Logging

.....
Vorname, Name, Matrikelnummer

1. Kennzeichnen Sie in der Config die Stellen wo über das

- was geloggt wird
- wieviel geloggt wird
- wo geloggt wird
- wie geloggt wird entschieden wird

```
<Configuration>
  <Appenders>
    <File name="A1" fileName="A1.log" append="false">
      <PatternLayout pattern="%t %-5p %c{2} - %m%n"/>
    </File>
    <Console name="STDOUT" target="SYSTEM_OUT">
      <PatternLayout pattern="%d %-5p [%t] %C{2} (%F:%L) - %m%n"/>
    </Console>
  </Appenders>
  <Loggers>

    <!-- You may want to define class or package level per-logger rules -->
    <Logger name="se2examples.core.businessLogic.VehicleManager" level="debug">
      <AppenderRef ref="A1"/>
    </Logger>
    <Root level="debug">
      <AppenderRef ref="STDOUT"/>
    </Root>
  </Loggers>
</Configuration>
```

2. Geben Sie je ein Beispiel wann Sie den loglevel

- error → bei Exceptions
 - info → nach erfolgreichem Ausführen eines Stück Codes
→ um zu sehen, in welchem Thread sich das Programm gerade befindet
 - debug → an Problemstellen
 - warn → wenn man schauen will, wie lange die Query dauert
- verwenden

3. Sie verwenden einen FileAppender für das Logging. Jetzt soll Ihre Application im Datacenter laufen. Was machen Sie mit dem FileAppender?

Rolling File Appender → fängt bei einer gewissen Grenze (z.B. 1 GB) an, den Anfang zu überschreiben

4. Macht Logging Ihre Application langsamer? Was passiert wenn Sie `log.debug(„foobar“);` aufrufen? Wie sollte sich das Logging Subsystem verhalten?

Ja, das Programm wird langsamer, jedoch sollten dadurch keine Performance-Probleme entstehen. Es wird nur geloggt, wenn sich das Level auf mindestens Debug befindet.

5. Ein Request an Ihre Application durchläuft einen Proxy Server, dann einen Web Server, dann einen Application Server und dann die Datenbank. Auf jedem Server loggen Sie die Requests. Welches Problem tritt auf?

Man muss darauf achten, dass die Server in der Uhrzeit nicht sekundengenau miteinander übereinstimmen, da die Log-Dateien ansonsten nicht mit den Servern synchronisiert sind.

6. Was sollten Sie pro Komponente/Tier loggen?

- info: Start-Erfolg, aktueller Thread
- error: Exception-Name und Beschreibung
- debug: Problemstellen, anstatt `System.out.println` `log.debug` verwenden
- warn: Dauer der Queries

7. Aus Geschwindigkeitsgründen halten Sie teure DB-Connections auf Vorrat in einem Pool. Jeder Request vom Client braucht dann eine Connection. Der Pool hat die Methoden:

```
DB Connection con = ConnectionPool.getConnection();  
ConnectionPool.freeConnection( DBConnection dbCon);
```

Was loggen Sie in Ihrem App Server? Oder anders gefragt: Was wollen Sie beim Umgang mit dem Pool als Software-Architektin wissen?

Auf Webserver-Level werden IP-Informationen und Protokolle geloggt.