LOGGING

Uhm yeh stuff went south with log4j. And some LIBs like JDA uses SLF4J and need a logging backend.

Here is a small guide of how to use the java's own Logging backend `java.util.loggin`

logging.properties

load file

is a file which specifies on how stuff gets logged, similar to log4j

you have several ways to tell java where this file is:

1. default config

I mean obviously ... usually located at `JAVA_HOME/jre/lib/logging.properties` or `/PATH_TO_JDK/conf/logging.properties`

2. use a flag on starting your application

tbh I forgot the flag but ... it's there!

3. use the LoggerManager

load your configuration here as example from `/main/resources/`

```
LogManager logManager = java.util.logging.LogManager.getLogManager();
logManager.readConfiguration(FOOBAABAZ.class.getClassLoader().getResourceAsStream("logging.properties"));
```

4. Set static JVM Properties

well that possible too but ... again i forgot how to ...

configure

following website helped me to understand what you could configure:

1. jenkov.com (general config)
2. oracle.com SimpleFormatter (docs for formatting)
3. oracle.com Formatter (docs for formatting)
4. geeksforgeeks.org (some examples)

I ended up with the following config:

```
handlers=java.util.logging.FileHandler, java.util.logging.ConsoleHandler

java.util.logging.FileHandler.level=ALL
java.util.logging.FileHandler.limit=10485760
java.util.logging.FileHandler.count=2
java.util.logging.FileHandler.append=true
java.util.logging.FileHandler.pattern=FOOBAABAZ%g.log
java.util.logging.FileHandler.formatter=java.util.logging.SimpleFormatter

java.util.logging.ConsoleHandler.level=ALL
java.util.logging.ConsoleHandler.formatter=java.util.logging.SimpleFormatter

java.util.logging.SimpleFormatter.format=[%1$TF-%1$TT][%2$s][%4$s] : %5$s %6$s %n

FOO.BAA.BAZ.level=WARNING
```

Version #4
Erstellt: 2021-12-11 15:19:08 UTC von Hiajen
Zuletzt aktualisiert: 2021-12-11 15:38:33 UTC von Hiajen