Logging

Es können `logI("text")` für INFO-Meldungen etc. verwendet werden. `logE` kann als zweiter Parameter zusätzlich noch eine Exception übergeben werden.

Zur besseren Übersicht im Log wird immer der Scriptname vorangestellt.

`logName` noch anpassen

es muss noch `com.batix.Log` importiert werden

Zum Kopieren (für Script)

```
/* Logging start */
def logName = "myScript"
//noinspection GroovyUnusedAssignment
def logD = { String msg, HttpServletRequest request = null -> Log.debug("[${logName}] ${msg}",
request) }
//noinspection GroovyUnusedAssignment
def logI = { String msg, HttpServletRequest request = null -> Log.info("[${logName}] ${msg}",
request) }
//noinspection GroovyUnusedAssignment
def logN = { String msg, HttpServletRequest request = null -> Log.notice("[${logName}]
${msg}", request) }
//noinspection GroovyUnusedAssignment
def logW = { String msg, HttpServletRequest request = null -> Log.warn("[${logName}] ${msg}",
request) }
//noinspection GroovyUnusedAssignment
def logE = { String msg, Exception ex = null, HttpServletRequest request = null ->
Log.error("[${logName}] ${msg}", request, ex) }
/* Logging end */
```

Zum Kopieren (für Klassen)

```
/* Logging start */
private static def logName = "myScript"
//noinspection GroovyUnusedAssignment
private static void logD(String msg, HttpServletRequest request = null) {
Log.debug("[${logName}] ${msg}", request) }
//noinspection GroovyUnusedAssignment
private static void logI(String msg, HttpServletRequest request = null) {
Log.info("[${logName}] ${msg}", request) }
//noinspection GroovyUnusedAssignment
private static void logN(String msg, HttpServletRequest request = null) {
Log.notice("[${logName}] ${msg}", request) }
//noinspection GroovyUnusedAssignment
private static void logW(String msg, HttpServletRequest request = null) {
Log.warn("[${logName}] ${msg}", request) }
//noinspection GroovyUnusedAssignment
private static void logE(String msg, Exception ex = null, HttpServletRequest request = null) {
Log.error("[${logName}] ${msg}", request, ex) }
/* Logging end */
```

Beispiel

```
def user = BxUser.findInstance(request)
if (!user) {
  logE("no user found")
  return
}

logI("found user ${user.id}")
```