

README

LoggerController

一个日志输出控制类

使用说明

详见如下代码

```
1 logRoot = os.path.abspath(os.path.join(__file__, "../log")) #指定日志存放的根路径
2 logController = LogController(logRoot) #使用根目录初始化日志控制器
3 logController.addLogger("socket", LOG_TYPE_DEBUG) #添加一个名为socket的日志控制类，记录DEBUG级别以
  上的日志
4 SocketLogger = logController.getLogger("socket") #获取socket的日志控制类
5 logController.addLogger("monitor", LOG_TYPE_ERROR) #添加一个名为monitor的日志控制类，记录Error级别
  以上的日志
6 MonitorLogger = logController.getLogger("monitor") #获取monitor的日志控制类
7 SocketLogger.logDebug("hello socket %s, %s", "world", "debug man") #SocketLogger输出Debug日志
8 SocketLogger.logInfo("hello socket %s, %s", "world", "info man") #SocketLogger输出info日志
9 SocketLogger.logError("hello socket %s, %s", "world", "error man") #SocketLogger输出error日
  志
10 MonitorLogger.logDebug("hello monitor %s, %s", "world", "debug man") #MonitorLogger输出Debug
   日志
11 MonitorLogger.logInfo("hello monitor %s, %s", "world", "info man") #MonitorLogger输出info日
   志
12 MonitorLogger.logError("hello monitor %s, %s", "world", "error man") #MonitorLogger输出
   error日志
13 #可以看到在logRoot下会有两个文件夹，monitor记录了只有Error的日志，Socket记录了Debug、Info、Error的日志
14
```

更详细的使用说明可参考Demo.py的使用范例