Matthew Johnson

4/28/16

**CUS1156 programming project 3**

**import** java.io.File;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** java.io.StringWriter;

**import** java.text.SimpleDateFormat;

**import** java.util.Date;

/\*\*

\* This program outputs the month, date, and year of a file

\* **@author** mJohn142

\*

\*/

**public** **class** FileLogger **extends** SimpleLogger {

**private** **static** FileLogger *inst*=**null**;

**private** FileWriter outFile;

**private** FileLogger()

{

System.***err***.println("in FileLogger constructor");

}

@Override

**public** **void** log(Level logLevel, String component, String msg) **throws** LoggerException

{

**if** (currLevel.permitsLogging(logLevel))

{

SimpleDateFormat sdf = **new** SimpleDateFormat("MM/dd/YYYY:HH:mm:ss");

Date now = **new** Date();

String logStr = sdf.format(now);

String lb = System.*getProperty*("line.separator");

logStr = logStr + ":" + logLevel.toString() + ":" + component + ":" + msg + lb;

**try** {

outFile = **new** FileWriter(**new** File("log.txt"), **true**);

outFile.write(logStr);

outFile.close();

} **catch** (IOException e) {

**throw** **new** LoggerException("could not log to file", e);

}

}

}

**public** **static** FileLogger getInstance() {

**return** **null**;

}

@Override

**public** **void** log(Level error, String component, Throwable t) **throws** LoggerException {

// **TODO** Auto-generated method stub

}

}

**public** **interface** Formatter {

**public** String format(LogMessage logMsg);

}

**public** **enum** Level {

***SEVERE***(4), ***ERROR***(3), ***WARNING***(2), ***INFO***(1), ***DEBUG***(0);

**private** **int** value;

**private** Level(**int** val)

{

value = val;

}

**public** **boolean** permitsLogging(Level logLevel)

{

**if** (value <= logLevel.value)

**return** **true**;

**else**

**return** **false**;

}

**public** String toString()

{

String ret;

**switch** (value) {

**case** 0: ret = "DEBUG";

**break**;

**case** 1: ret = "INFO";

**break**;

**case** 2: ret = "WARNING";

**break**;

**case** 3: ret = "ERROR";

**break**;

**case** 4: ret = "SEVERE";

**break**;

**default**: ret = "";

}

**return** ret;

}

}

**public** **class** LoggerException {

/\*\*

\* **@param** args

\*/

}

}

**public** **class** LoggerFactory {

**public** **static** SimpleLogger getLogger(String type) {

**if**(type.equals("file")){

**return** FileLogger.*getInstance*();

}

**else**

{

**return** **null**;

}

}

}

**public** **class** LogMessage(Level)

{

}

**public** **abstract** **class** PlainStringFormatter **implements** Formatter

{

}

**public** **abstract** **class** SimpleLogger {

**protected** Level currLevel = Level.***ERROR***;

**protected** Formatter formatter;

**public** **void** setLevel(Level l)

{

currLevel = l;

}

**public** Level getLevel()

{

**return** currLevel;

}

**public** **void** setFormatter(Formatter f)

{

formatter = f;

}

**public** **abstract** **void** log(Level logLevel, String component, String msg) **throws** LoggerException;

**public** **abstract** **void** log(Level error, String component, Throwable t) **throws** LoggerException;

}

**public** **abstract** **class** XMLFormatter **implements** Formatter

{

}