

<http://temp.repia.com/dhan/openapiclient-0.0.1-SNAPSHOT.jar.gz>

```
TB_ORGAN
CREATE CACHED TABLE "PUBLIC"."TB_ORGAN" (
  "OFCFS_CODE" VARCHAR(16),
  "OFCPS_NM" VARCHAR(100),
  "DEPT_CODE" VARCHAR(16),
  "DEPT_NM" VARCHAR(100),
  "USER_NM" VARCHAR(100),
  "OFFM_TELNO" VARCHAR(32),
  "EMAIL" VARCHAR(100),
  "JOB_DC" CLOB,
  "ORDR" NUMBER
);
```

```
java -Dlogback.configurationFile=resources/logback.xml -cp ../resources/ -
jar openapiclient-0.0.1-SNAPSHOT.jar HRD ORGAN refresh
```

```
#-----
#  api.properties :
#-----
#  1.  key = value
#  2.  key          가, value          가
#  3.  key          가,   value        가
#  4.          가   '\'   가(   '\'   '\'
```

```
database.type=h2
database.driver=org.h2.Driver
database.jdbcurl=jdbc:h2:tcp://127.0.0.1:9092/~/.rsa41
database.user=sa
database.password=fpvldk!@#

ORGAN.table=TB_ORGAN
```

```
logback.xml
<?xml version="1.0" encoding="UTF-8"?>
<!--
  [Layout]
  %m :
  %p : trace > debug > info > warn > error      priority
  %r :          가
  %c :   )   가 a.b.c          %c{2}   b.c가
  %n :          가          .  \r\n      \n
  %d :   가          (          .)
```

```
) %d{HH:mm:ss}            %d{dd MMMM yyyy HH:mm:ss}
%C :
)            가 org.apache.xyz.SomeClass            %C{2}    xyz.SomeClass 가

%M :            method
%F :
%l :            caller
%L :            caller
%X :            thread            NDC(nested diagnostic context)
%X :            thread            MDC(mapped diagnostic context)
%% : %
%t :            가
-->
```

```
<configuration scan="true" scanPeriod="30 seconds">
  <appender name="ROLLING"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <rollingPolicy
class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
    <!-- rollover daily -->
    <fileNamePattern>logs/openapiclient-%d{yyyy-MM-
dd}.%i.log.zip</fileNamePattern>
    <timeBasedFileNamingAndTriggeringPolicy
class="ch.qos.logback.core.rolling.SizeAndTimeBasedFNATP">
      <!-- or whenever the file size reaches 100MB -->
      <maxFileSize>100MB</maxFileSize>
    </timeBasedFileNamingAndTriggeringPolicy>
  </rollingPolicy>
  <encoder>
    <pattern>%d{HH:mm:ss.SSS} %-5level %logger{36} -
%msg%n</pattern>
  </encoder>
</appender>
  <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">
    <layout class="ch.qos.logback.classic.PatternLayout">
      <Pattern>%d{HH:mm:ss.SSS} [%-5level] - %msg%n</Pattern>
    </layout>
  </appender>

  <!-- Loggers -->
  <logger name="org.apache.catalina" level="ERROR">
</logger>

  <logger name="org.apache.commons" level="ERROR">
</logger>

  <logger name="org.springframework" level="DEBUG" >
</logger>

  <logger name="java.sql" level="DEBUG">
</logger>
```

```
<logger name="org.mybatis.spring" level="DEBUG">
</logger>

<root level="ERROR">
  <appender-ref ref="ROLLING"/>
  <appender-ref ref="STDOUT" />
</root>
</configuration>
```

```
rssStatus.sh
```

```
#!/bin/sh
```

```
trim() { echo $1; }
```

```
RSS_ROOT="/app/search/RSA"
```

```
echo "";
```

```
EXE_NAME="          "
EXE_FILE="DIC_MGR"
PID_FILE="${RSS_ROOT}/logs/dic_mgr.pid"
```

```
if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
fi
```

```
PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')
```

```
if [ ${PID} -eq ${RUN_PID} ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi
```

```
EXE_NAME="          "
EXE_FILE="DicSvr"
PID_FILE="${RSS_ROOT}/logs/dic.pid"
```

```
if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
```

```

fi

PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')

if [ ${PID} -eq ${RUN_PID} ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

EXE_NAME="          "
EXE_FILE="QP_MGR"
PID_FILE="${RSS_ROOT}/logs/qp_mgr.pid"

if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
fi

PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')

if [ ${PID} -eq ${RUN_PID} ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

EXE_NAME="          "
EXE_FILE="QuerySvr"
PID_FILE="${RSS_ROOT}/logs/1.pid"

if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
fi

PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')

if [ ${PID} -eq ${RUN_PID} ]

```

```
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

EXE_NAME=""
EXE_FILE="RSS_DAEMON"
PID_FILE="${RSS_ROOT}/logs/rssd.pid"

if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
fi

PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')

if [ ${PID} -eq ${RUN_PID} ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

EXE_NAME=""
EXE_FILE="rssAdmMgr"
PID_FILE="${RSS_ROOT}/logs/rssAdmMgr.pid"

if [ -f ${PID_FILE} ]
then
echo -n "";
#echo "${PID_FILE} exist!"
else
echo "${PID_FILE} not exist!"
fi

PID=$(trim `cat $PID_FILE`)
RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')

if [ ${PID} -eq ${RUN_PID} ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi
```

```
EXE_NAME=" 가 "
EXE_FILE="org.h2.tools.Server"
PID_FILE="${RSS_ROOT}/logs/db.pid"

RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')
#echo ${RUN_PID}

if [ "" != "${RUN_PID}" ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${RUN_PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

EXE_NAME=" "
EXE_FILE="AutoCompleteServer.jar"
PID_FILE="${RSS_ROOT}/logs/AutoComplete.pid"

RUN_PID=$(ps ax |grep ${EXE_FILE} |grep -v grep|awk '{print $1}')
#echo ${RUN_PID}

if [ "" != "${RUN_PID}" ]
then
echo " * ${EXE_NAME}(${EXE_FILE}:${RUN_PID})";
else
echo " * ${EXE_NAME}(${EXE_FILE}:Dead)";
fi

echo -e "\n\n";
```

```
backup_daily.sh
#!/bin/sh
datetime=$(date +%Y%m%d)
datetimeAgo=`date +%Y%m%d --date '30 days ago'`

echo "cd /app/search/RSA/bin"
cd /app/search/RSA/bin

echo "./RssConfBackup /app/search/RSA"
./RssConfBackup /app/search/RSA 1>/dev/null

echo "cd /app/search/RSA/backup/${datetime}"
cd /app/search/RSA/backup/${datetime}

#echo "mysqldump -u search -psearch12#$ AD > AD_${datetime}.sql"
#mysqldump -u search -psearch12#$ AD > AD_${datetime}.sql

echo "cd .."
cd ..
```

```
echo "tar cvf ./hrd176_$(date +%Y%m%d).tar.gz ./$(date +%Y%m%d)"
tar cfz ./hrd176_$(date +%Y%m%d).tar.gz ./$(date +%Y%m%d)

#echo "cp -f ./hrd176_$(date +%Y%m%d).tar.gz /data/backup/"
#cp -f ./hrd176_$(date +%Y%m%d).tar.gz /data/backup/

#echo "rm -rf ./hrd176_$(date +%Y%m%d)"
#rm -rf ./$(date +%Y%m%d)

echo "rm -rf ./hrd176_$(date +%Y%m%d).tar.gz"
rm -rf ./hrd176_$(date +%Y%m%d).tar.gz

#echo "rm -rf /data/backup/hrd176_$(date +%Y%m%d).tar.gz"
#rm -rf /data/backup/hrd176_$(date +%Y%m%d).tar.gz

#echo "cp -f /app/search/RSA/bin/html/js/* /data/bin/html/js"
#cp -f /app/search/RSA/bin/html/js/* /data/bin/html/js
echo "cp -Rf /app/search/RSA/conf/* /data/conf"
#cp -Rf /app/search/RSA/conf/* /data/conf
#echo "cp -f /app/search/RSA/logs/front/search_log/logAnal.out
/data/logs/front/search_log"
#cp -f /app/search/RSA/logs/front/search_log/logAnal.out
/data/logs/front/search_log
#echo "cp -f /app/search/RSA/logs/front/statistics_log/favor*
/data/logs/front/statistics_log"
#cp -f /app/search/RSA/logs/front/statistics_log/favor*
/data/logs/front/statistics_log
```

From:  
<http://125.132.25.164/dokuwiki/> -  
2023.12

Permanent link:  
<http://125.132.25.164/dokuwiki/doku.php?id=wiki:user:dhan:temp&rev=1630460688>

Last update: **2022/03/10 19:52**

