

# How to start Java JITL Jobs in a shell

- [JITL jobs in a shell](#)
- [Example](#)

## JITL jobs in a shell

Some of the JITL jobs are prepared to be executed from shell. What you need is

- the parameters,
- the name of the java class,
- the class path for the java implementation.

The name of the java class and the parameters can be found in the documentation of the job. All documentations can be found here: [JITL - JobScheduler Integrated Template Library](#)

The class path is normally not documented. You are on the safe side, if you add all .jar files from `$(SCHEDULER_HOME)/lib` to the class path. Handling will be easier, if you navigate to `$(SCHEDULER_HOME)/lib` and start the class from that location.

## Example

Here is an example how to start the job `JobSchedulerCleanupSchedulerDbMain` from shell:

- The parameters for the hibernate configuration file and for the delete interval are specified. Also the list of required jar files in the class path.
- You can execute this from shell by navigation to `$(SCHEDULER_HOME)/lib` and the execute this command.

### Start a job from the command line (Linux)

```
java -classpath /opt/sos-berlin.com/jobscheduler/scheduler_4444/lib/3rd-party/*:/opt/sos-berlin.com/jobscheduler/scheduler_4444/lib/sos/* -Dlog4j.configuration="file:///opt/sos-berlin.com/jobscheduler/scheduler_4444/lib/log4j.properties" com.sos.jitl.housekeeping.cleanupdb.JobSchedulerCleanupSchedulerDbMain -delete_interval=10 -hibernate_configuration_file=/home/test/sos-berlin.com/jobscheduler/scheduler_4444/config/hibernate.cfg.xml
```

This is an example to provide the call with a shell script.

### Example for a shell script executing a JITL-Job

```
#!/bin/sh
#
# -----
# Company: Software- und Organisations-Service GmbH
# Purpose: start Job cleanupHistoryDb
# -----

test -z "$SCHEDULER_HOME" && SCHEDULER_HOME="/opt/sos-berlin.com/jobscheduler/scheduler_4444"
test -z "$SCHEDULER_DATA" && SCHEDULER_DATA="/home/test/sos-berlin.com/jobscheduler/scheduler_4444"
test -z "$JAVA_HOME" && JAVA_HOME="/opt/Oracle_Java/jdk1.8.0_31/jre"

test -z "$LOG_BRIDGE" && LOG_BRIDGE="log4j"
test -z "$LOG4JPROP" && test -f "$SCHEDULER_HOME/lib/log4j.properties" && LOG4JPROP="-Dlog4j.
configuration=file://$SCHEDULER_HOME/lib/log4j.properties"

JAVA_BIN="$JAVA_HOME/bin/java"

export SCHEDULER_HOME
export SCHEDULER_DATA

CUR_DIR=`pwd`
cd "$SCHEDULER_HOME"

# set_classpath
CP="lib/patches/*:lib/jdbc/*:lib/user_lib/*:lib/3rd-party/*:lib/sos/*"

echo "$JAVA_BIN" -classpath "$CP" ${LOG4JPROP} $JAVA_OPTIONS -DSCHEDULER_HOME="$SCHEDULER_HOME" -
DSCHEDULER_DATA="$SCHEDULER_DATA" -DSCHEDULER_HOT_FOLDER="$SCHEDULER_HOT_FOLDER" com.sos.jitl.housekeeping.
cleanupdb.JobSchedulerCleanupSchedulerDbMain -delete_interval=10 -hibernate_configuration_file=$SCHEDULER_DATA
/config/hibernate.cfg.xml

"$JAVA_BIN" -classpath "$CP" ${LOG4JPROP} $JAVA_OPTIONS -DSCHEDULER_HOME="$SCHEDULER_HOME" -
DSCHEDULER_DATA="$SCHEDULER_DATA" com.sos.jitl.housekeeping.cleanupdb.JobSchedulerCleanupSchedulerDbMain -
delete_interval=10 -hibernate_configuration_file=$SCHEDULER_DATA/config/hibernate.cfg.xml

cd "$CUR_DIR"
```