**STRESS TESTING**

Shell Script: StressTestApp.sh

*#!/usr/bin/env bash -e*

*NUM\_PROCESSES=0*

*NUM\_LOOPS=0*

*WAITALL\_DELAY=1*

*while [ "$1" != "" ]; do*

*case "$1" in*

*"-NumBackGroundProcesses")*

*NUM\_PROCESSES=$2*

*shift; shift*

*;;*

*"-NumberLoop")*

*NUM\_LOOPS=$2*

*shift; shift*

*;;*

*\*)*

*echo "Unexpected argument: $1"*

*exit 1*

*;;*

*esac*

*done*

*if [ $NUM\_PROCESSES -eq 0 ] ; then*

*echo "Must specify at-least one background processes."*

*exit 1*

*fi*

*if [ $NUM\_LOOPS -eq 0 ] ; then*

*echo "Must specify at-least one loop."*

*exit 1*

*fi*

*echo $NUM\_PROCESSES $NUM\_LOOPS*

*declare -a pids*

*# run processes and store pids in array*

*for i in `seq 1 $NUM\_PROCESSES`; do*

*./gradlew testRelease $NUM\_LOOPS &*

*pids[${i}]=$!*

*echo "Test ${pids[i]} started at $(date)."*

*done*

*waitall() { # PID...*

*## Wait for children to exit and indicate whether all exited with 0 status.*

*local errors=0*

*while :; do*

*for pid in "$@"; do*

*shift*

*if kill -0 "$pid" 2>/dev/null; then*

*set -- "$@" "$pid"*

*elif wait "$pid"; then*

*debug "Test $pid finished successfully at $(date)."*

*else*

*error "Test $pid failed at $(date)."*

*((++errors))*

*fi*

*done*

*(("$#" > 0)) || break*

*sleep ${WAITALL\_DELAY:-1}*

*done*

*((errors == 0))*

*}*

*debug() { echo "DEBUG: $\*" >&2; }*

*error() { echo "ERROR: $\*" >&2; }*

*# wait for all pids*

*waitall ${pids[\*]}*

To Run the above script pass below parameters:

First parameter is NUM\_PROCESSES i.e. NumBackGroundProcesses

Second parameter is NUM\_LOOPS i.e. NumberLoop

*./StressTestApp.sh -NumBackGroundProcesses 20 -NumberLoop 100*

Output:

The following output is for 20 background processors and for 100 number of loops. We are also capturing log of start test and end test. Here, we are showing end date even if there is an error or any test failure.

For 100 background processors and for 100 number of loops it takes around 115mins to complete entire activity.

Below are the screenshots for below command i.e.

*./StressTestApp.sh -NumBackGroundProcesses 20 -NumberLoop 100*





