import hudson.model.\*

import com.tikal.jenkins.plugins.multijob.\*;

import jenkins.\*

import hudson.\*

import hudson.model.\*

import jenkins.model.\*

import java.util.regex.Matcher

import java.util.regex.Pattern

import java.\*

import java.lang.String

String job\_name = build.getEnvironment().get("JOB\_NAME");

def job =Jenkins.getInstance().getItemByFullName(job\_name);

println("Groovy script Started")

//job.getLastBuild()

build.getSubBuilds().each { subBuild->

print( subBuild.getBuildNumber()+' ')

println (subBuild.getJobName())

if ( subBuild.getJobName().equals("build\_linux\_master")){

def job1 =Jenkins.getInstance().getItemByFullName(subBuild.getJobName()).getBuildByNumber(subBuild.getBuildNumber())

job1.getSubBuilds().each { subBuild1->

if ( subBuild1.getJobName().equals("build\_src\_linux64\_master")){

def plat\_vom\_mh\_src\_lin64 = new ParametersAction([

new StringParameterValue("build\_src\_lin64",subBuild1.getBuildNumber().toString())

])

build.addAction(plat\_src\_lin64)

}

if ( subBuild1.getJobName().equals("build\_vom\_inux\_master")){

def build\_src\_lin = new ParametersAction([

new StringParameterValue("build\_vom\_ms\_src\_lin",subBuild1.getBuildNumber().toString())

])

build.addAction(build\_src\_lin)

}

}

}

}

println("Groovy script Ended")