

# Parameterized and Up/Downstream Project

# Agenda

- What build parameters are and their importance
- Modify parameters of our freestyle projects
- Create upstream and downstream projects

# Build Parameters

- Build parameters help us set environment variables for our projects
- We can make use of build parameters to customize our build to make it run differently

# Parameterized Projects

- On the Jenkins dashboard, select New Item on the top left of the dashboard
- On the next page, select Freestyle project as the project type
- Enable this option
  - This project is parameterized
- Add few parameters
- Access parameters in Build\Execute Shell
  - `echo $<param name>`
- Build with Parameters option on the left-hand menu, which is the fifth item.

# Activity: Setting up Parameterized Projects

- You have been asked to set up a freestyle project that will contain two string parameters, your first and last names, and a build step executing a bash script that will display the greeting **Hello** followed by your full name.

# Steps for Completion

- On the Jenkins dashboard, go to the configuration panel and select New Item.
- On the New Item view, enter a project name, select Freestyle project, then select OK.
- Under the general section of the project configuration, enter an appropriate project description.
- Enable the checkbox
  - This project is parameterized
- Add the first parameter by clicking on the Add Parameter drop-down menu and selecting String Parameter
- Fill in the form with the details of the first parameter, that is, your first name, as displayed
- Follow the same procedure to create the second parameter for your last name

# Steps for Completion

- In the build section of the project configuration, select the Add build step drop-down menu and select Execute shell
- This presents us with a form requiring us to insert the command we want to execute. Populate the text area
  - `echo "Hello ${FIRST_NAME} ${LAST_NAME}"`
- Press Apply and Save at the bottom of the screen to save your project configuration. This will direct you to the project dashboard.
- On the left-hand project options menu, select Build with parameters to build the project
- Fill in the parameters if you are not comfortable with running the project with the default parameters. Select Build.
- On the build history on the left, hover over the build number and select Console Output

# Build Triggers



# Build Triggers

- To trigger the build based on event automatically

# Creating Upstream/Downstream Projects

- We will be creating two projects and using one to trigger the other
- Given two projects, project A and project B
  - If project B is configured to run once project A completes, we call project B the downstream project and project A the upstream project
- Let's create Project A and Project B

# Build Triggers

- Configuration of Project B

## Build Triggers

☐ Trigger builds remotely (e.g., from scripts)

☒ Build after other projects are built

Projects to watch

Project A,

☒ Trigger only if build is stable

☐ Trigger even if the build is unstable

☐ Trigger even if the build fails

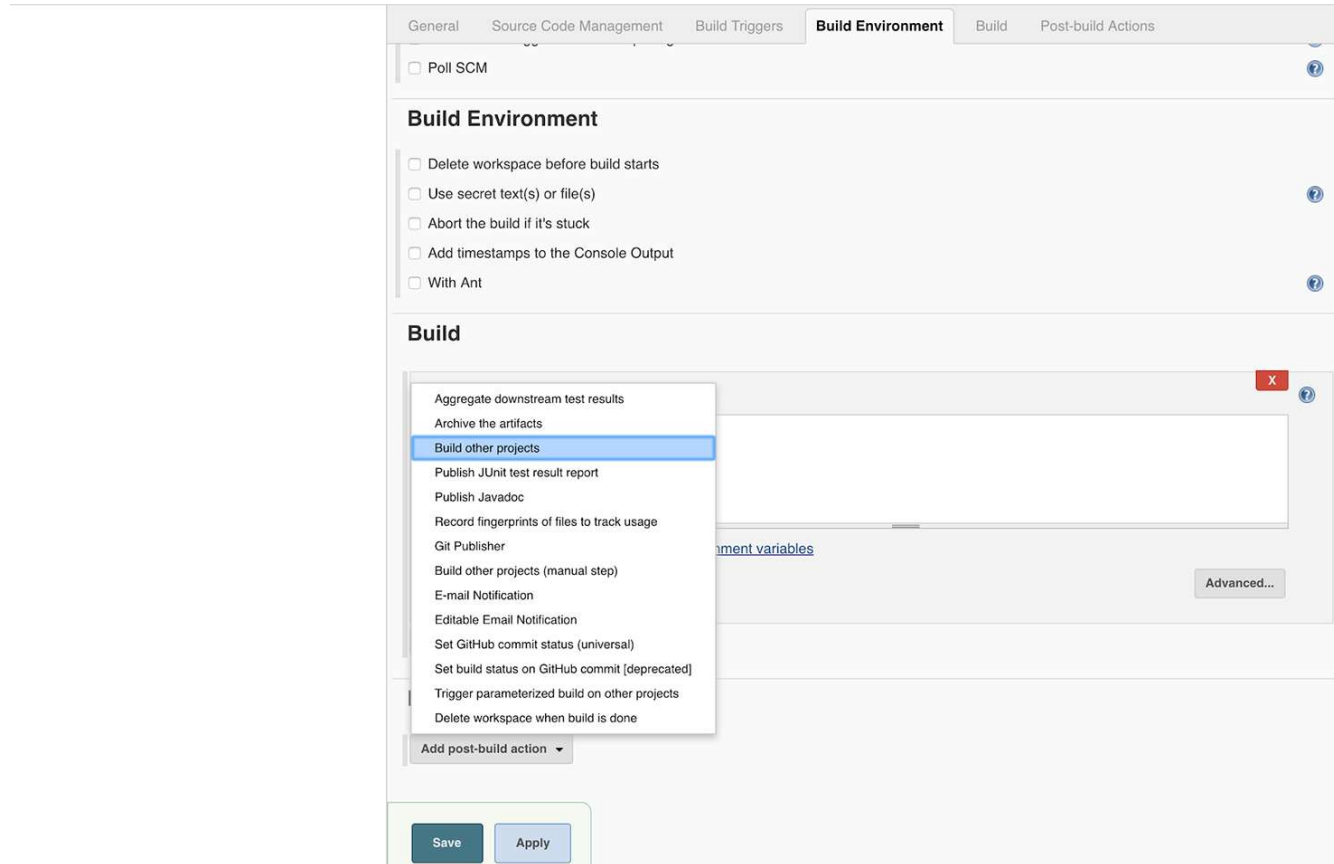
☐ Build periodically

☐ GitHub hook trigger for GITScm polling

☐ Poll SCM

# Build Triggers

- Configuration of Project A



# Running an Upstream Project

- Now that we've understood how to configure upstream and downstream projects
- Build Project A
- Select Project B under the Downstream Projects section
- Check Console Output

*Thanks*