

# While Action

The While action will continue to execute its children until a certain condition is met. The condition is specified in the While action and uses the same condition syntax as [Agent Requirements](#), [Stage Gates](#) and [Configuration Conditions](#).

## While

While Action

While

Comments

Required Field

Name

While [" "]

☒ Enabled

Left

Operator

Exists

☒ Write the expanded expression to the build log for each iteration

Validate

Save

Cancel

Help

### Name

A friendly name for this action (will be displayed in the [actions workflow area](#)).

### Enabled

Determines if this action will be run within the relevant stage.

### Left

Left value in condition expression.

### Operator

Operator to be applied to left and right values of expression.

### Right

Right value in condition expression.

### Case-sensitive text comparison

Tick this to ensure that the case of string values on both sides must be compared when evaluating the expression.

### Write the expanded expression to the build log for each iteration

Tick this to write the expanded expression to the build log for each iteration.

## Example

This example demonstrates a way to use the while action to repeat the same tasks. All it does is loops 10 times and each time it loops it executes the [Delay action](#). First you need to declare a variable which has the starting value of the loop and also doubles as the counter variable which increments in value each pass of the loop. To do this, create a [Configuration Variable](#) called "Counter" and set the Type to Numeric and Value to 0.

Edit Variable

Required Field

Variable Name:

Counter

Description:

Type:

Number

?

Prompt Type:

(none)

Value:

0

Sensitive:

☐

Enabling this option will mask the variable's value from build logs and other sensitive areas.

Save

Cancel

Help

Now head back to the [Stage Editor](#) and click the While action. You'll need to select **Less Than** from the Operator drop down list to get the **Right Value** input field to appear. The **Right Value** is straight forward, put in the number 10 or whatever number you would like to count up to. The **Left Value** will need to reference the **Counter** Configuration variable we create in the last step.

While Action

While

Comments

Name

While [" "]

☒ Enabled

Left

%Counter%

Operator

Less than

Right

10

☐ Case-sensitive text comparison

☒ Write the expanded expression to the build log for each iteration

Validate

Save

Cancel

Help

It's important to note, as explained on the [Variables, Objects & Expressions](#) page the way the **Counter** variable is referenced here in this action. Even though it's a Configuration variable, I have chosen to remove the Configuration namespace prefix and reference it using only its name. The next step is to increment the **Counter** variable using the [Set Variable action](#). Select the **Counter** variable from the list and make sure you change the **Operation** type to **Append or increment**. Then set the **New Value** to the amount you want the counter incremented by, in this case we'll use 1.

**Note:** The Configuration Variable **Counter** which we change the value of in the Set Variable action is not actually modified, the value of it is copied into the Build namespace and modified there. To reference the new value in the Build namespace we use the syntax **%Counter%**. That is why we excluded the Configuration namespace prefix when specifying the Counter variable in the While action. We want the While action to use the updated value in the Build namespace. If we were to use **%Configuration.Counter%** then the value would always be what we set it to on the [Variables](#) page which was 0.

Set Variable Action

Set Variable

Comments

Required Field

Name

Set Variable [Counter]

☒

Enabled

Variable

Counter

Select the name of the build variable to set. Expression variables which cannot be modified are not listed.

Operation

Append or increment

Replace value or increase/append the value. Note that DateTime/Date/Time variables can only be replaced.

New Value

1

The new value of the variable. Note that any date time variable values should be entered as UTC. Full date time format is yyyy-MM-dd'T'HH:mm:ss'Z' e.g. 2099-12-31T23:59:59Z

☐ Expand variable references in new value

Specify whether New Value should be expanded. eg. When checked, \$Server.HostName\$ = <server\_hostname>

Validate

Save

Cancel

Help

**Important:** The set variable action and any other action you wanted executed in each pass of the loop should be placed as a child of the while action.

Once you've setup the While action and the Set Variable action, you can begin putting actions in the while loop you want executed each pass. For simplicity I'll add the [Delay action](#). Your final workflow should look like this.

| Enabled | Action       | Name                               |
|---------|--------------|------------------------------------|
| ✓       | While        | While ['%Counter%' Less than '10'] |
| ✓       | Set Variable | Set Variable [Counter]             |
| ✓       | Delay        | Delay [60s]                        |