

Creating A Freestyle Project

Objective: In this lab, you will create a **Freestyle UI-driven Jenkins Project**. Then, you will build the project and see the results.

Jenkins Freestyle Project is a repeatable

- build job
- script
- or pipeline

that contains **steps** and **post-build** actions.

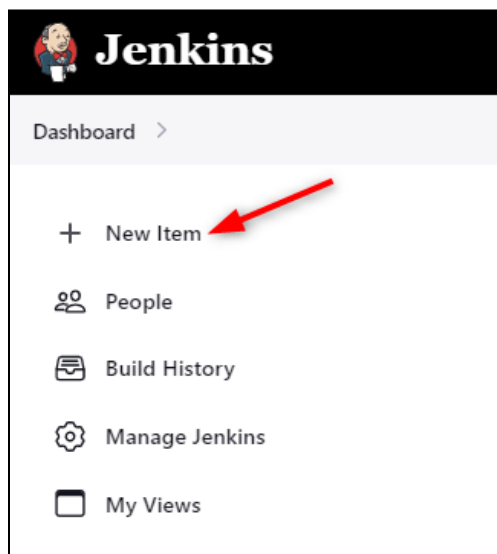
It allows users to configure build triggers (**automatically starts/schedules a build whenever the developer makes changes to SCM**). Users can configure Jenkins plugins to build steps and post-build actions.

Create a new project

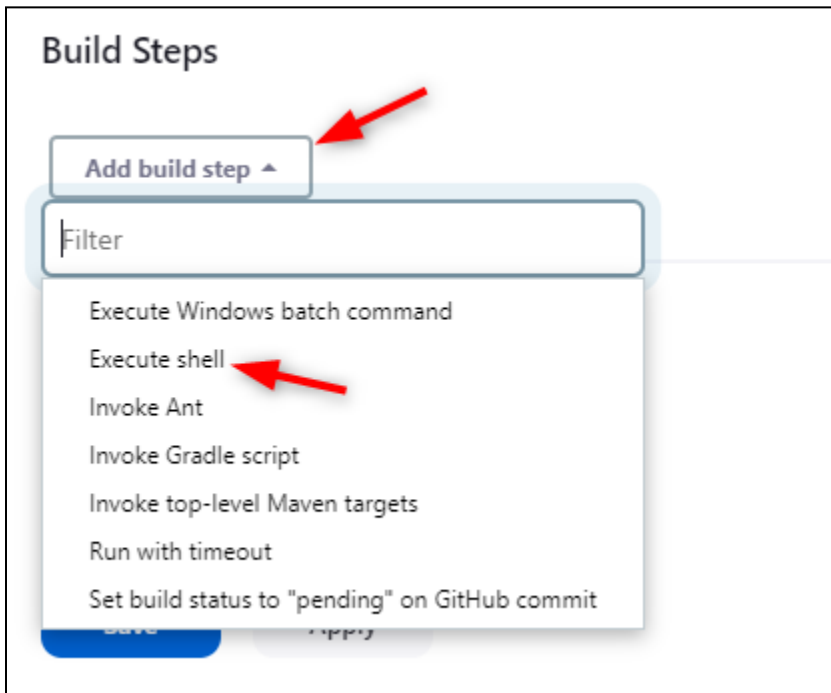
1. If you aren't already on the main Jenkins dashboard, click the "**Jenkins**" title on the top left



2. Click on "New Item"



3. Enter "cow-world" in the "Enter an item name" box
4. Click "Freestyle project"
5. Click OK
6. Under Project configuration enter a description of your liking
7. Scroll to the bottom. Under 'Build Steps', click 'Add build step'
8. Choose 'Execute shell'



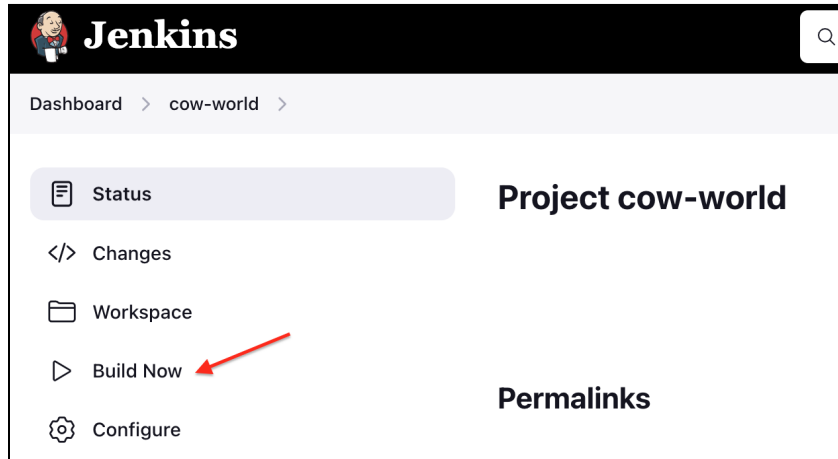
9. In the Execute shell **Command** window, enter the code:

```
echo "This project is called $JOB_NAME"  
/usr/games/cowsay "Hello World"
```

10. Click **Save**

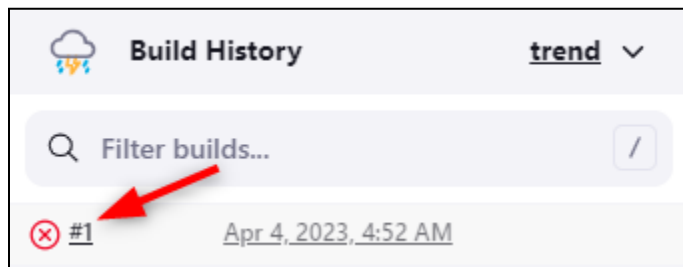
11. Click 'Build Now'

NOTE: This build will fail. This is expected behavior because the package 'cowsay' is not installed on our Jenkins master.



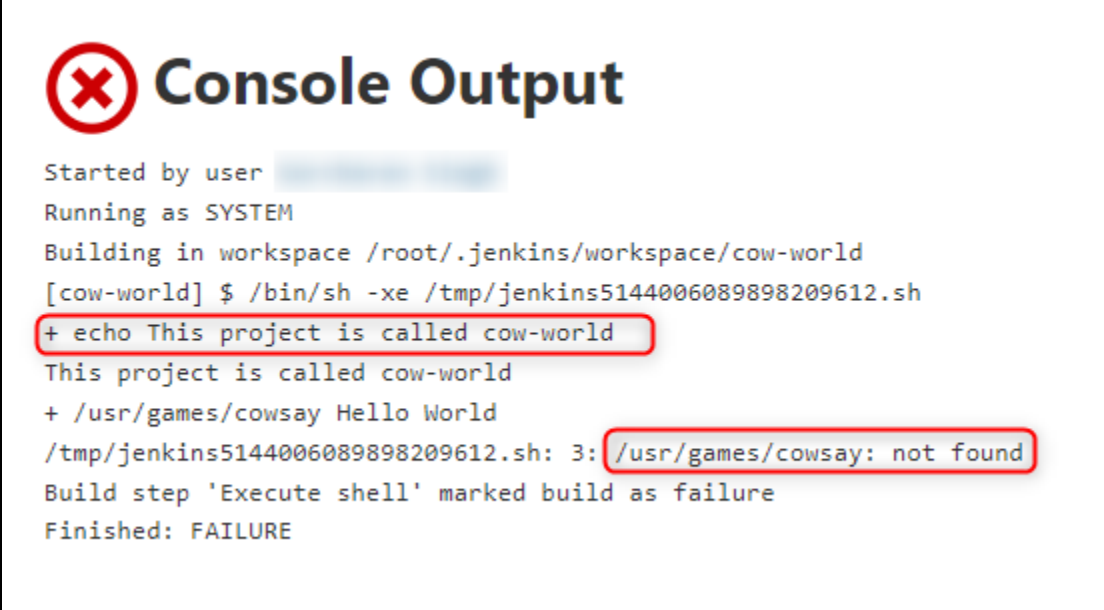
Note this fails.

12. Under the '**Build History**' on the left, look for the red ball next to the **#1**, indicating that **Build #1** was not successful.



13. Click the red cross under '**Build history**', to view and read the errors.
14. First, see that your echo command worked. It wrote to the console output "This project is called cow-world", because it interpolated your \$JOB_NAME environment variable.

15. Next, look for the error that reads: "cowsay: not found"



The image shows a Jenkins 'Console Output' window. At the top, there is a red circle with a white 'X' icon followed by the text 'Console Output'. Below this, the output text is as follows: 'Started by user [redacted]', 'Running as SYSTEM', 'Building in workspace /root/.jenkins/workspace/cow-world', '[cow-world] \$ /bin/sh -xe /tmp/jenkins5144006089898209612.sh', '+ echo This project is called cow-world' (this line is highlighted with a red box), 'This project is called cow-world', '+ /usr/games/cowsay Hello World', '/tmp/jenkins5144006089898209612.sh: 3: /usr/games/cowsay: not found' (this line is highlighted with a red box), 'Build step \'Execute shell\' marked build as failure', and 'Finished: FAILURE'.

```
Started by user [redacted]
Running as SYSTEM
Building in workspace /root/.jenkins/workspace/cow-world
[cow-world] $ /bin/sh -xe /tmp/jenkins5144006089898209612.sh
+ echo This project is called cow-world
This project is called cow-world
+ /usr/games/cowsay Hello World
/tmp/jenkins5144006089898209612.sh: 3: /usr/games/cowsay: not found
Build step 'Execute shell' marked build as failure
Finished: FAILURE
```

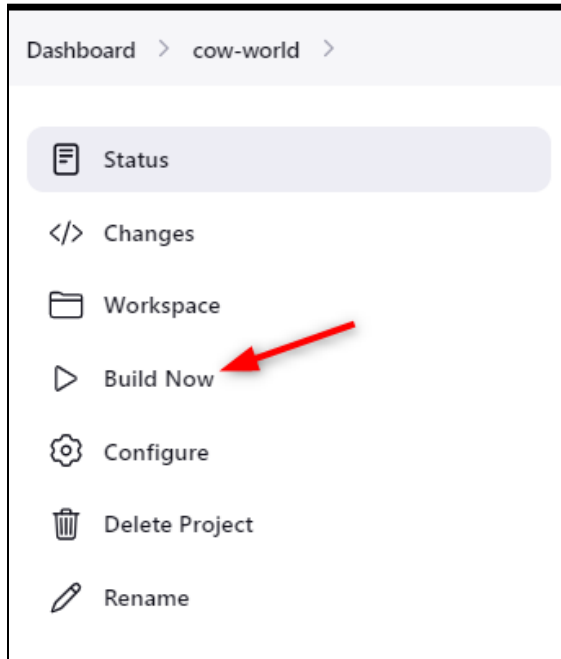
It seems cowsay is not installed on this machine, so Jenkins can not use it.

16. To resolve this, we will install cowsay on Jenkins, so from the Jenkins Master VM in a terminal window, run this command:

\$ sudo apt-get install -y cowsay

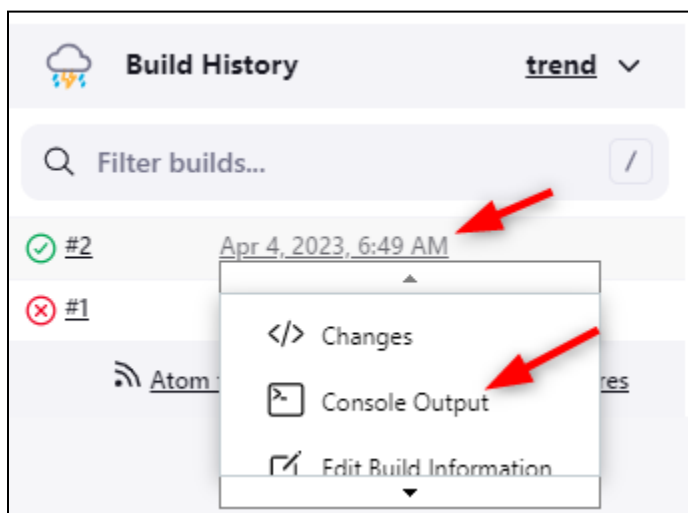
17. Re-run the build

18. Back in the Jenkins interface, click on your “**Project**” and re-run the build with '**Build Now**'



19. Under the '**Build History**' on the left, look for the green tick next to the #2, indicating that Build #2 was successful. If you have a red cross next to #2, duck for cover and watch out for rogue cows!! In the case of a failed build, click 'Configure', fix the errors and try again.

20. Click the **dropdown** next to the #2 and click '**Console output**'. This is another way to get to the console output (you could also just click the Green Tick).



Console Output

```
Started by user [redacted]
Running as SYSTEM
Building in workspace /root/.jenkins/workspace/cow-world
[cow-world] $ /bin/sh -xe /tmp/jenkins2974205707000049598.sh
+ echo This project is called cow-world
This project is called cow-world
+ /usr/games/cowsay Hello World

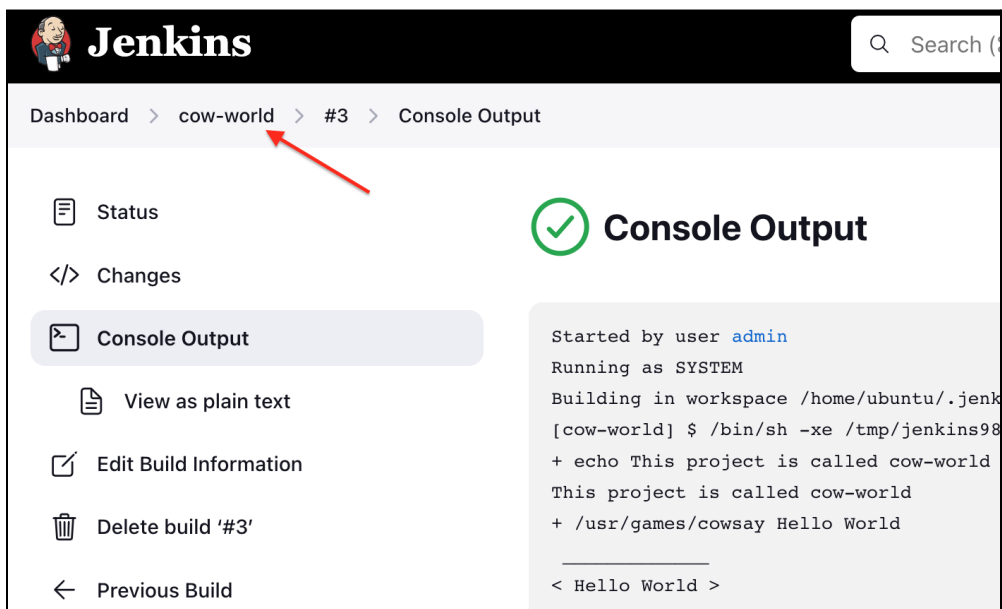
  _____
< Hello World >
  -----
      \   ^__^
       \  (oo)\_______
          (__)\       )\/\
              ||----w |
              ||     ||

Finished: SUCCESS
```

You should have a cow

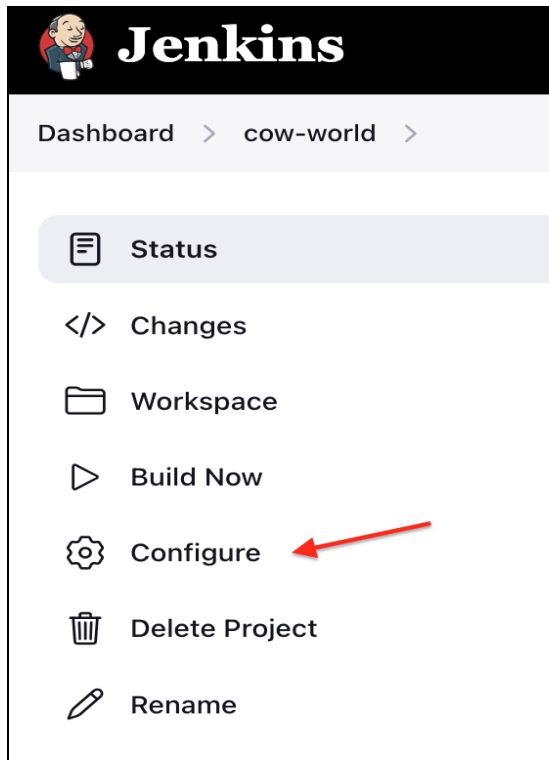
Add Parameterization

21. Click on your project 'cow-world'.



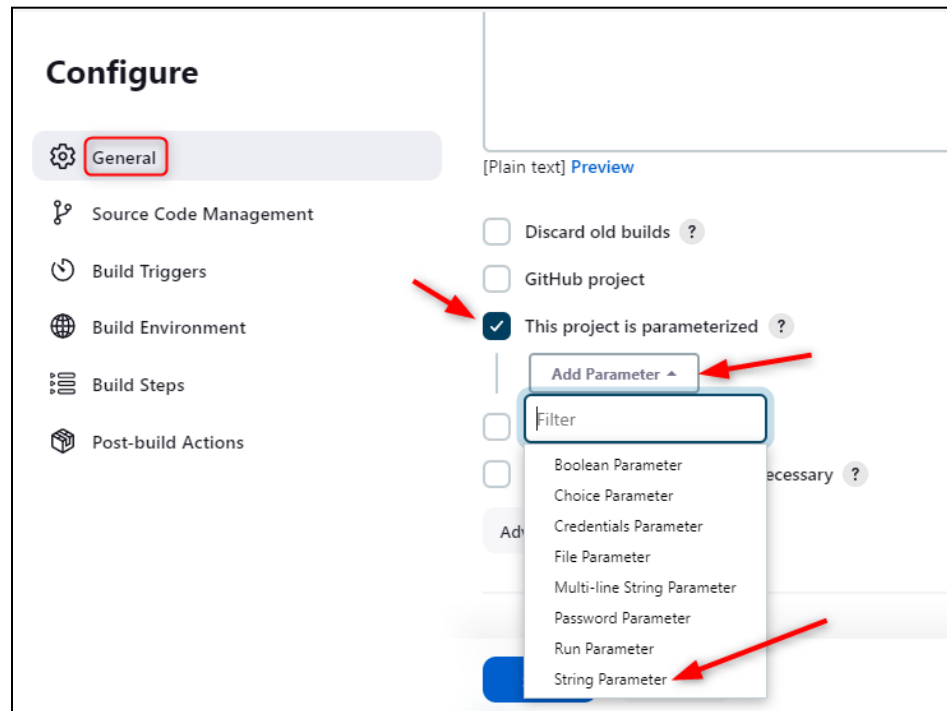
The screenshot shows the Jenkins web interface. At the top, the Jenkins logo and a search bar are visible. Below the header, a breadcrumb navigation path reads: Dashboard > cow-world > #3 > Console Output. A red arrow points to the 'cow-world' link. On the left side, there is a sidebar with several links: Status, Changes, Console Output (which is highlighted with a light blue background), View as plain text, Edit Build Information, Delete build '#3', and Previous Build. The main content area on the right is titled 'Console Output' with a green checkmark icon. It displays the same terminal output as shown in the first image, including the 'Hello World' message and the 'Finished: SUCCESS' status.

22. Click '**Configure**' on the left nav bar



23. Under the General section, check the box labeled '**This project is parameterized**'

a. Click 'Add parameter'



b. Choose 'String Parameter'

i. Name: cow_param

ii. Default Value: \$JOB_NAME

iii. Description: What the cow says

24. Scroll down to the Execute shell field and utilize the new parameter.
Change your code to:

```
/usr/games/cowsay "Hello $cow_param"
```


Build Steps

≡ Execute shell ?

Command

See [the list of available environment variables](#)

```
echo "This project is called $JOB_NAME"  
/usr/games/cowsay "Hello $cow_param"
```

25. Save the project configuration and then click **Build with Parameters**.
26. Leave the default value for cow_param of \$JOB_NAME and click 'Build'.

Project cow-world

This build requires parameters:

cow_param

What the cow says

\$JOB_NAME

Build

27. Look at the console output for your most recent build. Notice what the cow says now. She is saying "Hello" followed by the name of your project "cow-world". This is because \$JOB_NAME resolved to the name of the project, which is "cow-world"

Change the parameter value

28. Click "Back to Project" and run the build again:

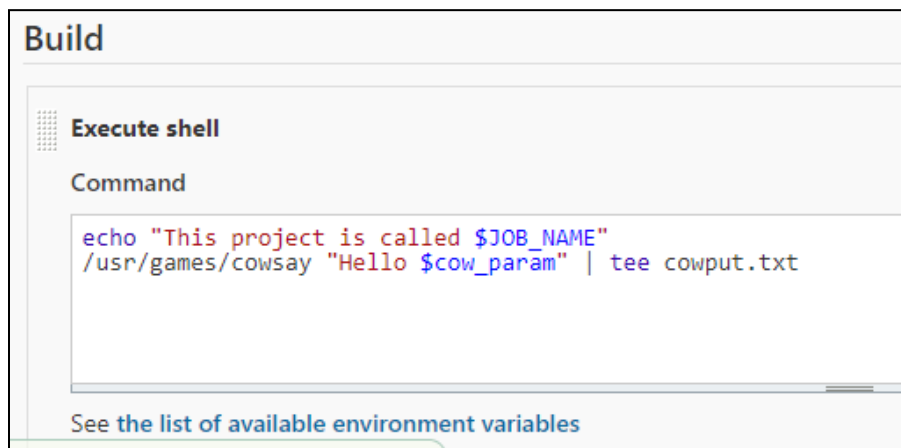
- a. Click 'Build with Parameters'
- b. Change the parameter \$JOB_NAME in the Build window to any value you want
- c. Click 'Build'
- d. See what the cow says in the console output for your latest build

Write to a file to the workspace

29. Go back to the configuration of your project

30. Change the Execute Shell command to match the following two lines:

```
echo "This project is called $JOB_NAME"  
/usr/games/cowsay "Hello $cow_param" | tee cowput.txt
```



31. This command will send the output to the console, and also create a file in the workspace, named cowput.txt, with the cow output

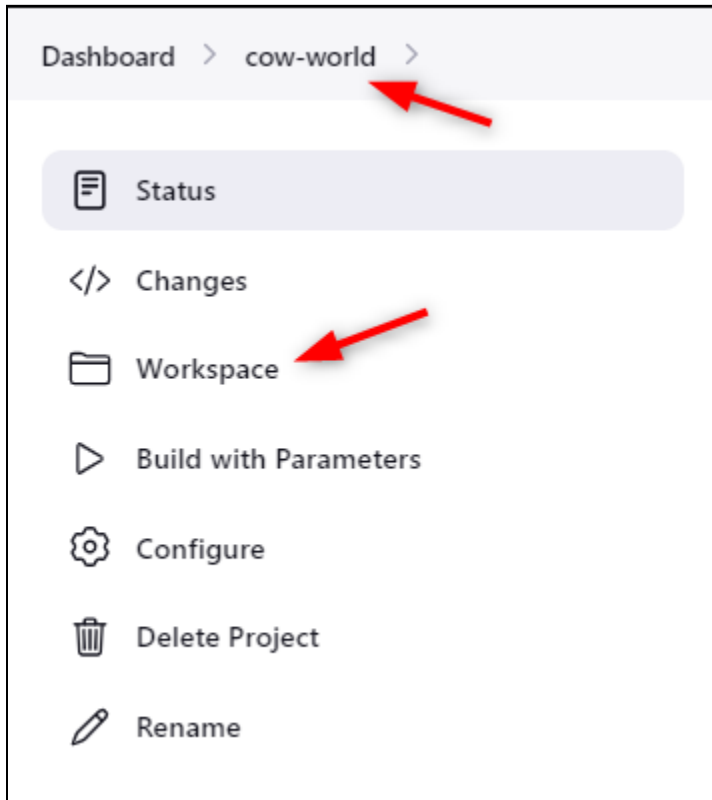
32. Save & Build

33. Check your console output for two items:

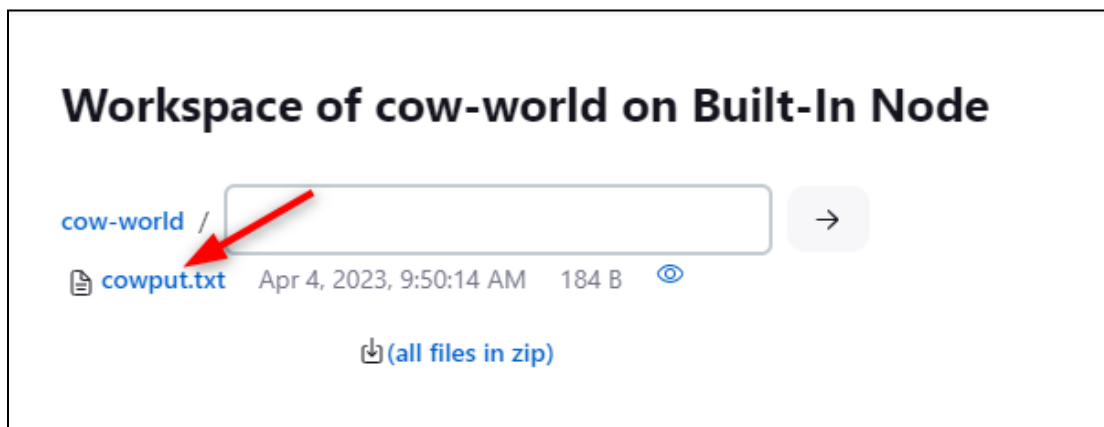
- a. check for what you echoed to the console

b. check for your cow

34. Click '**Back to Project**', and then click "**Workspace**" in the middle of the page



35. Click **cowput.txt** and see your cow



36. Click the **browser back button** to return to the Jenkins interface (don't close the browser tab, or you close the Jenkins interface)

37. Return to the main dashboard (click the "Jenkins" logo at the top left of the browser window)

Notify your instructor that you are done with the lab

END OF LAB