



GitLab

GitLab CI/CD

A Complete DevOps Platform Delivered as a Single Application



Manage



Plan



Create



Verify



Package



Secure



Release



Configure



Monitor



Defend

Single Conversation
Single Data Store
Single Permission Model
Single Interface
Governance & Security
Team Collaboration
Lifecycle Analytics



Release Stage CI/CD

Lab Assignment Part 3

Scenario: Create a `.gitlab-ci.yml` file to visualize an automated CI process.

Key Tasks to Complete

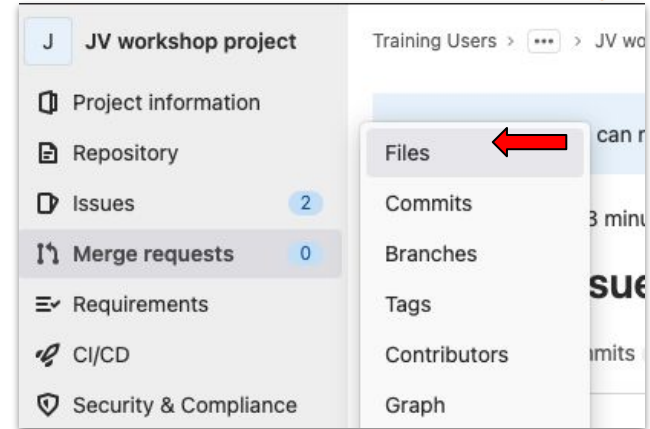
1. Create a `.gitlab-ci.yml` File
2. View the CI pipeline
3. Update the CI pipeline

Access repository files



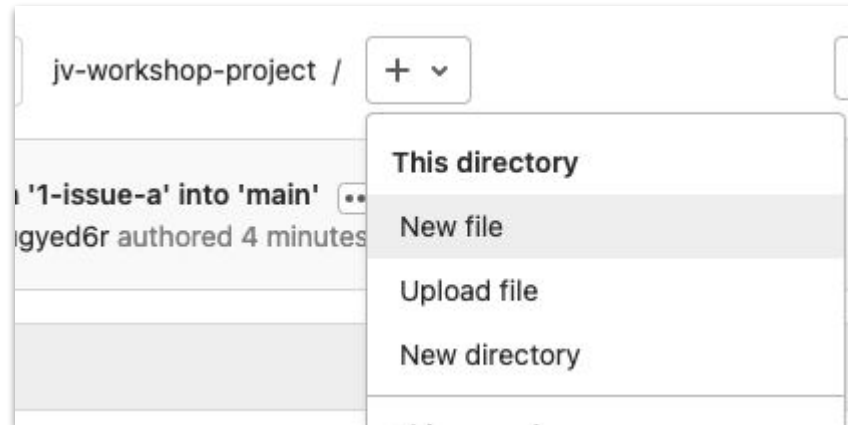
In the MR view:

1. Select **Repository**→**Files**.



In the repository file view:

2. Select the '+' → 'New file'

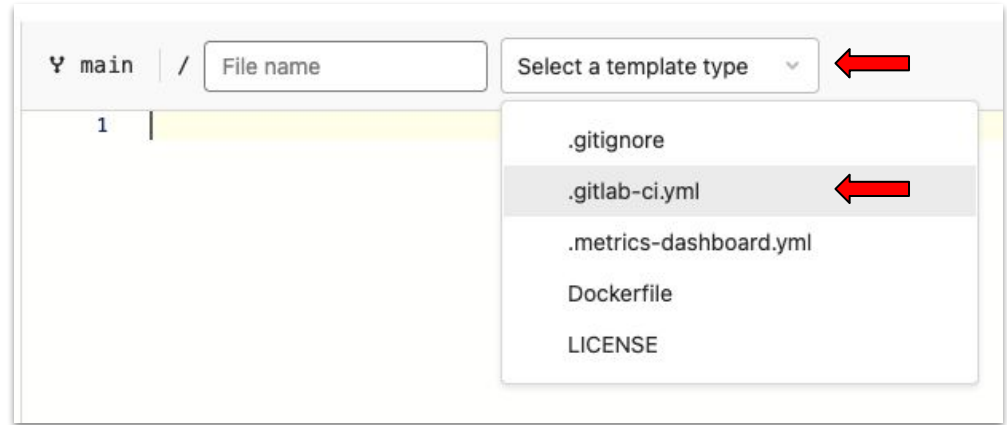


Select file template type



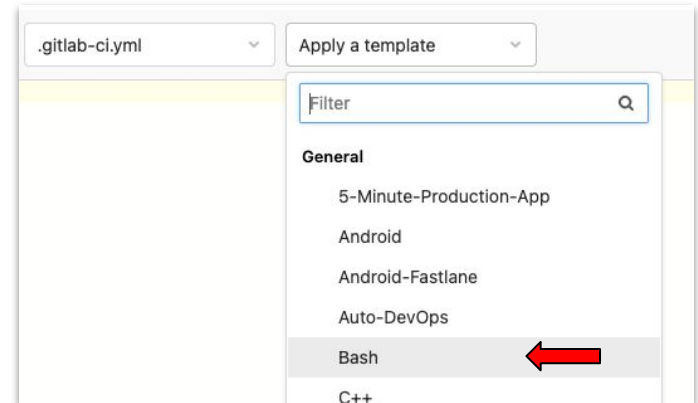
In the New file view:

1. Select the '**Select a template type**' drop-down→
'**.gitlab-ci.yml**' option.



In the New file view:

2. Select the '**Apply a template**' drop-down→ '**Bash**' option.



File title and contents populated with template



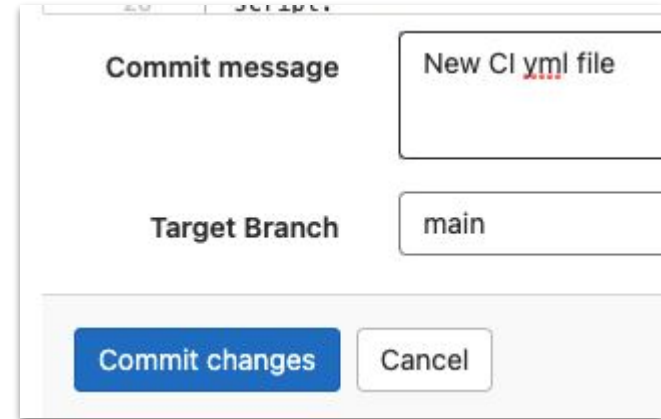
```
Y main / .gitlab-ci.yml .gitlab-ci.yml Apply a template
1 # This file is a template, and might need editing before it works on your project.
2 # To contribute improvements to CI/CD templates, please follow the Development guide at:
3 # https://docs.gitlab.com/ee/development/cicd/templates.html
4 # This specific template is located at:
5 # https://gitlab.com/gitlab-org/gitlab/-/blob/master/lib/gitlab/ci/templates/Bash.gitlab-ci.yml
6
7 # See https://docs.gitlab.com/ee/ci/yaml/README.html for all available options
8
9 # you can delete this line if you're not using Docker
10 image: busybox:latest
11
12 before_script:
13   - echo "Before script section"
14   - echo "For example you might run an update here or install a build dependency"
15   - echo "Or perhaps you might print out some debugging details"
16
17 after_script:
18   - echo "After script section"
19   - echo "For example you might do some cleanup here"
20
21 build:
22   stage: build
23   script:
24     - echo "Do your build here"
25
26 test1:
27   stage: test
28   script:
```

Add commit message



In the New file view:

1. Enter <commit message text> into the 'Commit message' pane.
2. Select the 'Commit changes' button.

A screenshot of a commit dialog box. It has two input fields: 'Commit message' with the text 'New CI yml file' and 'Target Branch' with the text 'main'. At the bottom are two buttons: 'Commit changes' (blue) and 'Cancel' (white). A red arrow points from the left towards the 'Commit changes' button, and another red arrow points from the right towards the 'Commit message' field.

Components of this .gitlab-ci.yml



In the file view, confirm the following components.

1. Leverages Docker **image**.
2. **'before_script'** block.
3. **'after_script'** block.
4. Pipeline elements of **'stage'** and **'script'**.



.gitlab-ci.yml 1.15 KB

EditWeb IDE

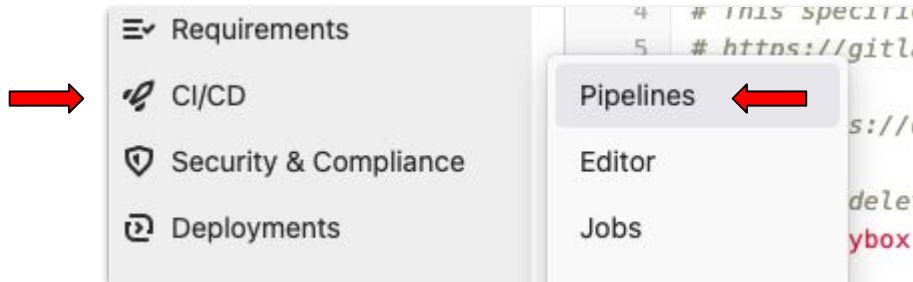
```
1  # This file is a template, and might need editing before it works on
2  # To contribute improvements to CI/CD templates, please follow the D
3  # https://docs.gitlab.com/ee/development/cicd/templates.html
4  # This specific template is located at:
5  # https://gitlab.com/gitlab-org/gitlab/-/blob/master/lib/gitlab/ci/t
6
7  # See https://docs.gitlab.com/ee/ci/yaml/README.html for all availab
8
9  # you can delete this line if you're not using Docker
10 image: busybox:latest
11
12 before_script:
13   - echo "Before script section"
14   - echo "For example you might run an update here or install a buil
15   - echo "Or perhaps you might print out some debugging details"
16
17 after_script:
18   - echo "After script section"
19   - echo "For example you might do some cleanup here"
20
21 build1:
22   stage: build
23   script:
24     - echo "Do your build here"
25
```


Access repository pipelines



In the file view:






1. Select **CI/CD**→**Pipelines**.




In the Pipelines view:

2. Confirm the pipeline elements below.

- Pipeline status.
- Pipeline ID.
- Pipeline triggerer.
- Pipeline target.
- Pipeline stages and status.
- Pipeline time.

Status	Pipeline	Triggerer	Commit	Stages	Duration
 passed	#132290 latest		main ~ 1fb67868 New CI yaml file	  	00:00:22 1 minute ago



Access pipeline details

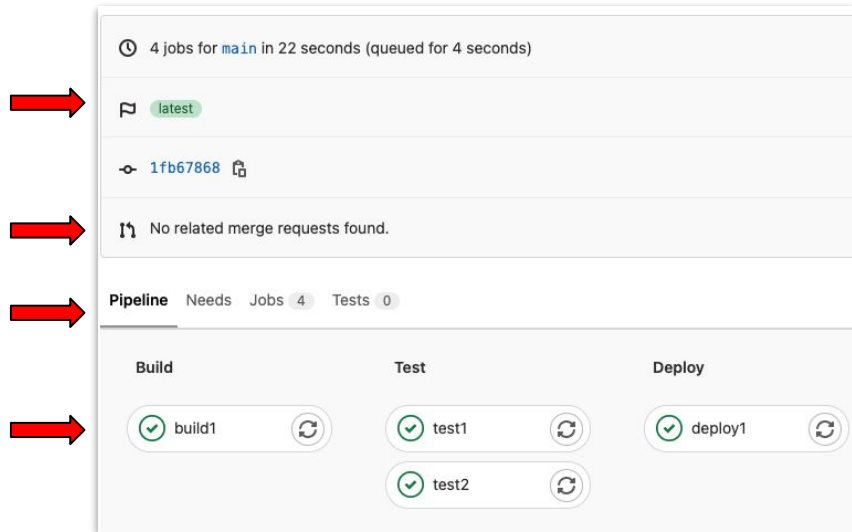
In the Pipelines view:

1. Select the '<pipeline status>' button, or
2. Select the '<pipeline ID>' hyperlink.

In the pipeline view:

3. Confirm the additional pipeline elements.

- 'latest' flag,
- Related merge requests.
- Additional tabs for context.
- Expanded pipeline view.

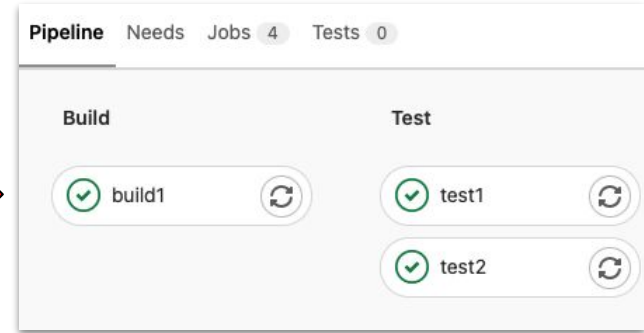


Access job details



In the pipeline view:

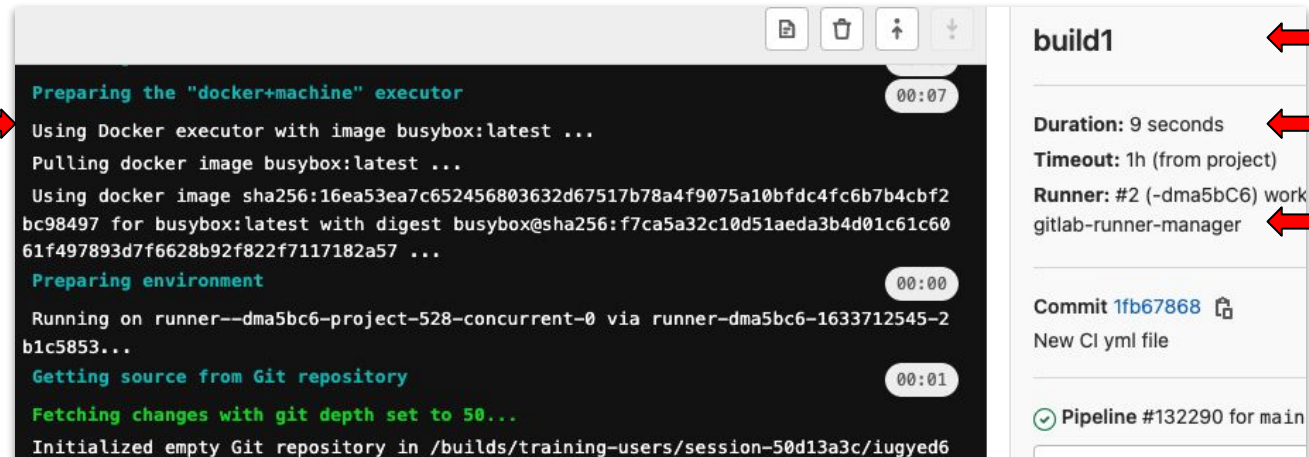
1. Select '<individual job>' button.



In the job view:

2. Confirm the following elements.

- Job name.
- Job duration and runner name.
- Job artifacts (if avail.).
- Pipeline stage drop-down.
- List of pipeline jobs.
- Terminal output.

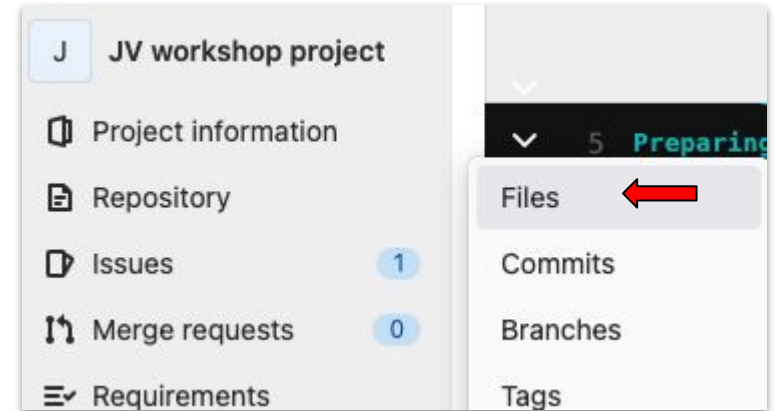


Access repository files



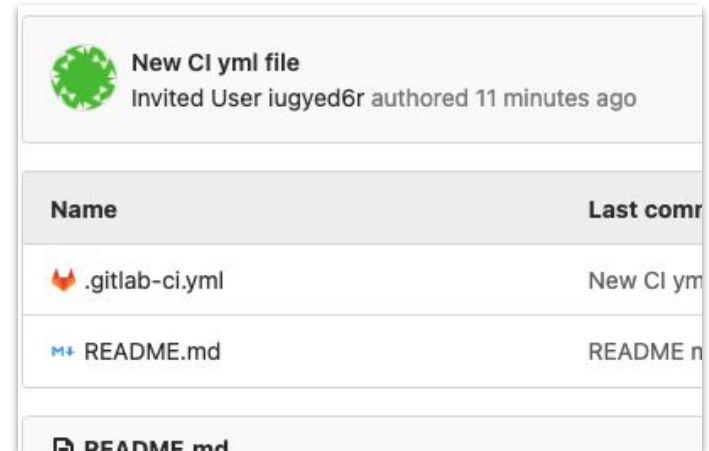
In the job view:

1. Select **Repository**→**Files**.



In the Repository files list view:

2. Select **'`.gitlab-ci.yml`'** file name.



Open .gitlab-ci.yml file to edits



In the file content view:

1. Select the **Edit** button.



```
1 # This file is a template, and might need editing before it works on
2 # To contribute improvements to CI/CD templates, please follow the De
3 # https://docs.gitlab.com/ee/development/cicd/templates.html
4 # This specific template is located at:
5 # https://gitlab.com/gitlab-org/gitlab/-/blob/master/lib/gitlab/ci/te
6
7 # See https://docs.gitlab.com/ee/ci/yaml/README.html for all availabl
```

In the file edit view:

2. Copy the contents of the **'test1:'** element.



```
20
21 ∨ build1:
22   stage: build
23   script:
24     - echo "Do your build here"
25
26 ∨ test1:
27   stage: test
28   script:
29     - echo "Do a test here"
30     - echo "For example run a test suite"
31
32 ∨ test2:
33   stage: test
34   script:
35     - echo "Do another parallel test here"
```

Modify the .gitlab-ci.yml file - part 1



In the file edit view:

1. Scroll to the bottom of the file.
2. Insert a space after the last row in the file.
3. **Paste** the copied contents to the end of the `.gitlab-ci.yml` file.



```
37
38  deploy1:
39    stage: deploy
40    script:
41      - echo "Do your deploy here"
42
43  test1:
44    stage: test
45    script:
46      - echo "Do a test here"
47      - echo "For example run a test suite"
```

In the file edit view:

4. Modify the pasted contents to reflect:

```
test3:
  stage: test
  script:
    - cat README.md
```



```
37
38  deploy1:
39    stage: deploy
40    script:
41      - echo "Do your deploy here"
42
43  test3:
44    stage: test
45    script:
46      - cat README.md
47
```

Add commit message for .gitlab-ci.yml file



In the file edit view:

1. Enter **<commit message text>** into the **'Commit message'** pane.
2. Select the **'Commit changes'** button.

The screenshot shows the GitLab file editor interface. At the top, a code editor displays the contents of the `.gitlab-ci.yml` file, with line numbers 43 to 47. The code defines a `test3` stage with a `script` block containing `cat README.md`. Below the code editor, there is a form for committing changes. The **Commit message** field contains the text "Created test3 job for test stage in `gitlab-ci.yml`". The **Target Branch** is set to `main`. At the bottom of the form, there are two buttons: **Commit changes** (in blue) and **Cancel** (in white). A red arrow points from the first step of the instructions to the commit message field, and another red arrow points from the second step to the **Commit changes** button.

In the file view:

3. Verify the following components.
 - Commit status message.
 - Configuration lint.

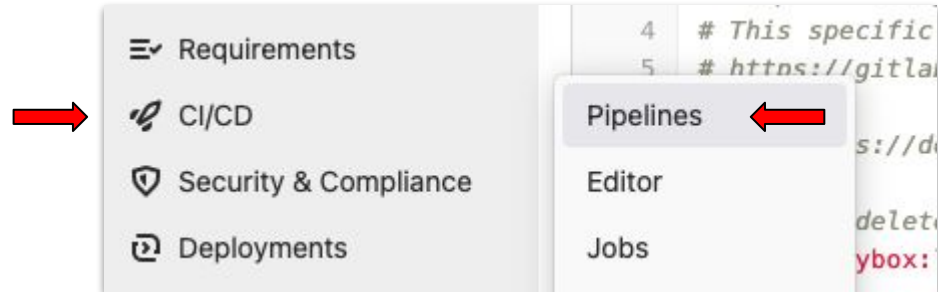
The screenshot shows the GitLab file view after a successful commit. A blue banner at the top displays a success message: "Your changes have been successfully committed." Below this, the file path is shown as `main` / `ju-workshop-project` / `.gitlab-ci.yml`. The commit status is displayed with a green circular icon and the message "Created test3 job for test stage in `gitlab-ci.yml`". Below the status, it says "Invited User iugyed6r authored just now". At the bottom, there is a green checkmark icon followed by the text "This GitLab CI configuration is valid. [Learn more](#)". A red arrow points from the third step of the instructions to the success banner, and another red arrow points from the bottom of the instructions to the configuration lint message.

Access repository pipelines



In the file view:















1. Select **CI/CD**→**Pipelines**.







In the Pipelines view:

2. Confirm updates to the following pipeline elements.

- Pipeline status.
- Pipeline ID.
- Pipeline triggerer.
- Pipeline target.
- Pipeline stages and status.
- Pipeline time.

Status	Pipeline	Triggerer	Commit	Stages	Duration
	#132292 latest		 main → b1b2318b Created test3 job for ...	  	⌚ 00:00:23 🕒 2 minutes ago
	#132291 yml invalid error		 main → 27184cff Created test3 job to t...		🕒 5 minutes ago
	#132290		 main → 1fb67868 New CI yml file	  	⌚ 00:00:22 🕒 22 minutes ago



Access pipeline details



In the Pipelines view:

1. Select the '<pipeline status>' button, or
2. Select the '<pipeline ID>' hyperlink. →

Status	Pipeline	Triggerer	Commit	Stages	Duration
passed	#132292 latest		main → b1b2318b Created test3 job for ...	✓ ✓ ✓	00:00:23 2 minutes ago

In the pipeline view:

3. Confirm the following pipeline elements.
 - Commit message listed.
 - Additional job added to “Test” stage.



Created test3 job for test stage in gitlab-ci.yml

5 jobs for main in 23 seconds (queued for 4 seconds)

latest

b1b2318b

No related merge requests found.

Pipeline Needs Jobs 5 Tests 0

Build	Test	Deploy
✓ build1	✓ test1	✓ deploy1
	✓ test2	
	✓ test3	

Access job details



In the pipeline view:

1. Select '<individual job>' button.

Example:

1. Select the '**test3**' button.

Created test3 job for test stage in gitlab-ci.yml

🕒 5 jobs for `main` in 23 seconds (queued for 4 seconds)

🚩 latest

🔗 b1b2318b 📄

🔍 No related merge requests found.

Pipeline Needs Jobs 5 Tests 0

Build	Test	Deploy
✓ build1 🔄	✓ test1 🔄	✓ deploy1 🔄
	✓ test2 🔄	
	✓ test3 🔄	

Job details

In the job view:

1. Confirm the following elements.

- Job name reflects added job.
- Job duration and runner name.
- Job artifacts (if avail.).
- Pipeline stage drop-down. includes added job.
- Terminal output reflects “script” for ‘test3’ job.

A screenshot of the GitLab CI job view for a job named 'test3'. The interface is divided into two main sections. The left section is a terminal window showing the execution steps and output. The right section is a summary panel. The terminal output shows steps for preparing the Docker executor, preparing the environment, getting source from the Git repository, and executing the 'step_script' stage. The summary panel shows the job name 'test3', duration '8 seconds', timeout '1h', runner 'workshop-gitlab-runner-manager', commit 'b1b2318b', and the pipeline 'Pipeline #132292 for main'. The job is listed as 'test' in the pipeline stage dropdown. The job status is 'test3' with a 'Retry' button. Red arrows point to the job name 'test3', the 'Retry' button, the job name in the pipeline stage dropdown, and the job name in the pipeline stage dropdown.

test3 Retry

Duration: 8 seconds
Timeout: 1h (from project)
Runner: #2 (-dma5bc6) workshop-gitlab-runner-manager

Commit [b1b2318b](#)
Created test3 job for test stage in gitlab-ci.yml

✓ Pipeline #132292 for main
test

✓ test1
✓ test2
→ ✓ test3

```
5 5 Preparing the "docker+machine" executor 00:05
6 Using Docker executor with image busybox:latest ...
7 Pulling docker image busybox:latest ...
8 Using docker image sha256:16ea53ea7c652456803632d67517b78a4f9075a10bfdc4fc6b7b4cbf2
  bc98497 for busybox:latest with digest busybox@sha256:f7ca5a32c10d51aeda3b4d01c61c60
  61f497893d7f6628b92f822f7117182a57 ...
10 10 Preparing environment 00:01
11 Running on runner--dma5bc6-project-528-concurrent-0 via runner-dma5bc6-1633744046-0
  8037a5c...
13 13 Getting source from Git repository 00:00
14 Fetching changes with git depth set to 50...
15 Initialized empty Git repository in /builds/training-users/session-50d13a3c/iugyed6
  r/jv-workshop-prep/jv-workshop-project/.git/
16 Created fresh repository.
17 Checking out b1b2318b as main...
18 Skipping Git submodules setup
20 20 Executing "step_script" stage of the job script 00:01
21 Using docker image sha256:16ea53ea7c652456803632d67517b78a4f9075a10bfdc4fc6b7b4cbf2
  bc98497 for busybox:latest with digest busybox@sha256:f7ca5a32c10d51aeda3b4d01c61c60
  61f497893d7f6628b92f822f7117182a57 ...
22 $ echo "Before script section"
23 Before script section
24 $ echo "For example you might run an update here or install a build dependency"
```



- CI/CD example projects: <https://docs.gitlab.com/ee/ci/examples/#cicd-examples>
- CI/CD templates: <https://docs.gitlab.com/ee/ci/examples/#cicd-templates>



GitLab

Everyone can contribute