

# CICD

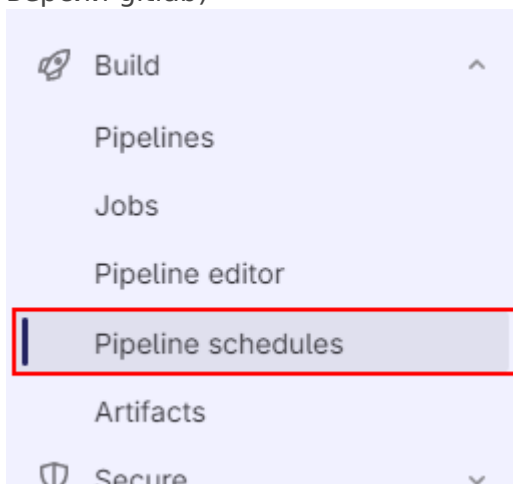
Информация по непрерывной интеграции (continuous integration) и непрерывному развертыванию (continuous delivery или continuous deployment) программного обеспечения в процессе разработки

- [Как реализовать запуск задач в .gitlab-ci.yml с различной периодичностью выполнения](#)

# Как реализовать запуск задач в .gitlab-ci.yml с различной периодичностью выполнения

## 1. Создать 2 планировщика (sheduler)

1. Меню: **Build - Pipeline Schedules** или **CI/CD - Schedules** (в зависимости от версии gitlab)



## 2. Нажать на кнопку "New schedule"



## 3. Указать в разделе переменных переменную для одного расписания

# Edit Pipeline Schedule

## Description

Every 60 min

## Interval Pattern

- ☐ Every day (at 2:00am)  
☐ Every week (Thursday at 2:00am)  
☐ Every month (Day 5 at 2:00am)  
☒ Custom ([Learn more.](#))

\* / 60 9-07 \* \* 1-5

## Cron Timezone

[UTC+3] Moscow

## Target branch or tag

main

## Variables

Variable

interval

long

X

Variable

Input variable key

Input variable

Hide value

## Activated

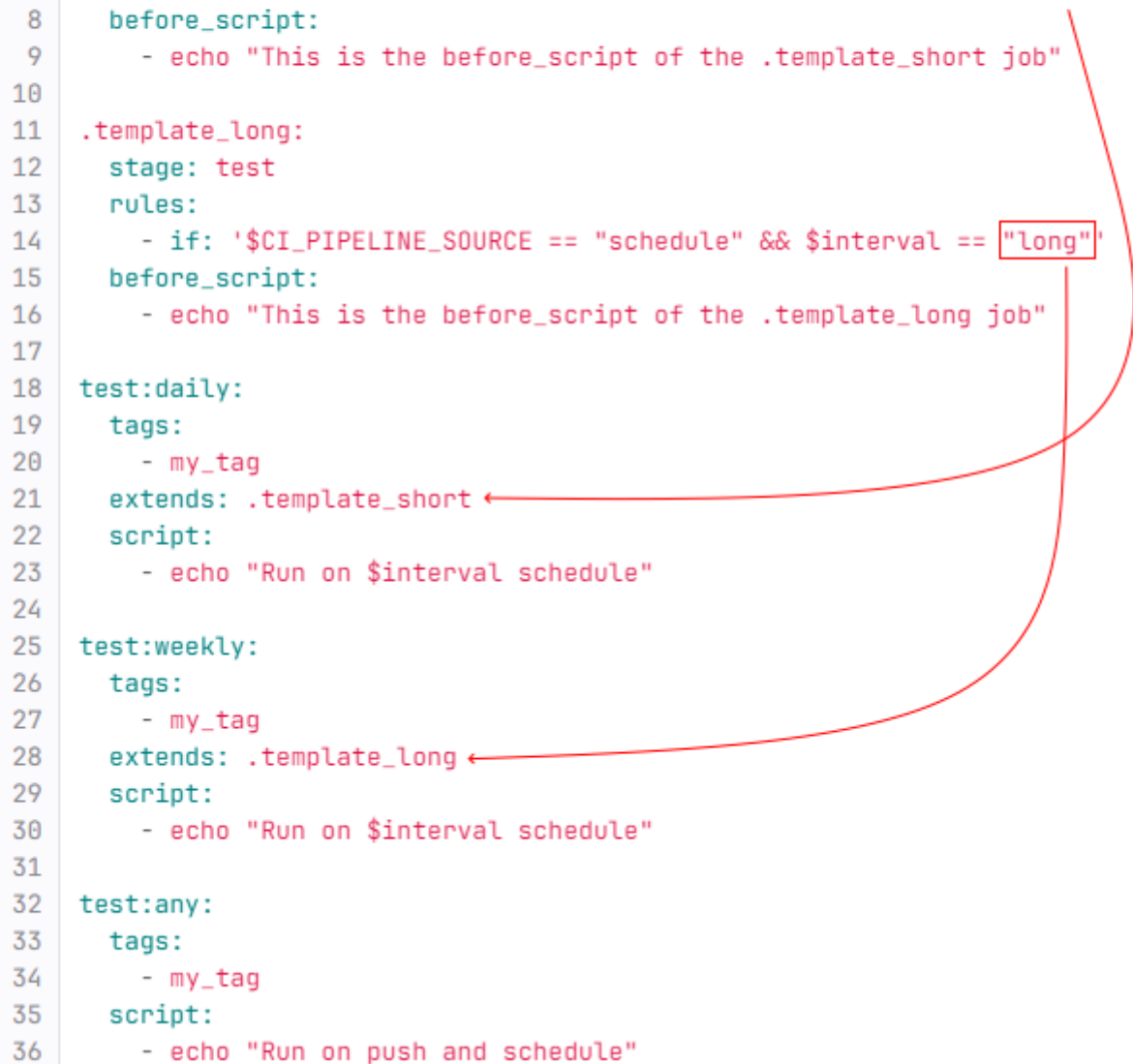
☒ Active

Save pipeline schedule

Cancel

4. Аналогично добавить второй планировщик и указать для него свою переменную
2. В файле .gitlab-ci.yml добавить правила выполнения для каждой переменной

```
1 stages:
2   - test
3
4 .template_short:
5   stage: test
6   rules:
7     - if: '$CI_PIPELINE_SOURCE == "schedule" && $interval == "short"'
8   before_script:
9     - echo "This is the before_script of the .template_short job"
10
11 .template_long:
12   stage: test
13   rules:
14     - if: '$CI_PIPELINE_SOURCE == "schedule" && $interval == "long"'
15   before_script:
16     - echo "This is the before_script of the .template_long job"
17
18 test:daily:
19   tags:
20     - my_tag
21   extends: .template_short
22   script:
23     - echo "Run on $interval schedule"
24
25 test:weekly:
26   tags:
27     - my_tag
28   extends: .template_long
29   script:
30     - echo "Run on $interval schedule"
31
32 test:any:
33   tags:
34     - my_tag
35   script:
36     - echo "Run on push and schedule"
```



### 3. Пример файла .gitlab-ci.yml

```
stages:
  - test

.template_short:
  stage: test
  rules:
    - if: '$CI_PIPELINE_SOURCE == "schedule" && $interval == "short"'
```

before\_script:

- echo "This is the before\_script of the .template\_short job"

.template\_long:

stage: test

rules:

- if: '\$CI\_PIPELINE\_SOURCE == "schedule" && \$interval == "long"'

before\_script:

- echo "This is the before\_script of the .template\_long job"

test:daily:

tags:

- my\_tag

extends: .template\_short

script:

- echo "Run on \$interval schedule"

test:weekly:

tags:

- my\_tag

extends: .template\_long

script:

- echo "Run on \$interval schedule"

test:any:

tags:

- my\_tag

script:

- echo "Run on push and schedule"