

# GABRIELE ROMANATO

## Le variabili delle pipeline di GitLab

Le **variabili delle pipeline** di GitLab sono un elemento chiave per rendere le pipeline CI/CD (Continuous Integration/Continuous Deployment) più flessibili, sicure e personalizzabili. Consentono agli sviluppatori di parametrizzare configurazioni, gestire credenziali sensibili e creare pipeline più generiche e riutilizzabili. In questo articolo esploreremo in dettaglio cosa sono, come funzionano, i tipi di variabili disponibili e alcune best practice per usarle.

Le variabili delle pipeline sono valori dinamici che possono essere definiti e utilizzati all'interno dei file di configurazione `.gitlab-ci.yml`. Queste variabili possono essere usate per sostituire stringhe statiche, fornire valori sensibili come token API, configurare comportamenti delle pipeline o passare informazioni tra i vari job.

Nel file `.gitlab-ci.yml`:

```
variables:
  APP_ENV: production
  APP_VERSION: 1.0.0

build_job:
  script:
    - echo "Deploying version $APP_VERSION in $APP_ENV
environment"
```

Quando la pipeline viene eseguita, il comando restituirà:

```
Deploying version 1.0.0 in production environment
```

GitLab offre diversi tipi di variabili, ciascuna con specifici utilizzi e metodi di configurazione.

## Variabili predefinite

Sono variabili integrate in GitLab e disponibili automaticamente per tutte le pipeline. Contengono informazioni utili sul progetto, sull'ambiente e sul job.

Esempi:

- `CI_PROJECT_NAME`: Nome del progetto.
- `CI_COMMIT_REF_NAME`: Nome del branch o del tag.
- `CI_PIPELINE_ID`: ID univoco della pipeline.

## Variabili definite dall'utente

Queste variabili possono essere dichiarate direttamente nel file `.gitlab-ci.yml` o a livello di progetto, gruppo o pipeline.

Configurazione a livello di progetto:

1. Vai su **Settings > CI/CD > Variables**.
2. Crea una nuova variabile specificando:
  - **Key**: Nome della variabile.
  - **Value**: Valore assegnato.
  - **Scope**: Accesso limitato a determinate pipeline o job.

Le variabili definite in un gruppo sono ereditate da tutti i progetti al suo interno.

## Variabili di ambiente

Permettono di definire valori specifici per uno stage o job specifico.

```
deploy:
  variables:
    APP_ENV: production
  script:
    - echo "Environment: $APP_ENV"
```

## Variabili protette

Le variabili protette possono essere utilizzate solo da job associati a branch o tag protetti. Questa caratteristica è utile per gestire in modo sicuro credenziali o configurazioni sensibili.

Creazione:

- Vai su **Settings > CI/CD > Variables**.
- Seleziona l'opzione "Protected".

## Variabili mascherate

Le variabili mascherate nascondono il loro valore nei log di esecuzione delle pipeline, utile per proteggere informazioni sensibili come chiavi API.

Requisiti:

- Il valore deve contenere solo caratteri alfanumerici.
- Non deve superare i 512 caratteri.

## Variabili segrete (secret variables)

Le secret variables sono ideali per la gestione di dati sensibili come password o token. Sono configurabili tramite l'interfaccia di GitLab e non sono visibili nel repository o nei file `.gitlab-ci.yml`.

# Variabili passate da pipeline manuali

Le pipeline manuali consentono di passare variabili durante l'avvio.

```
test:
  script:
    - echo "Running in $MODE"
```

Quando si avvia manualmente una pipeline, è possibile specificare il valore di `MODE`.

## Priorità delle variabili

Quando sono definite variabili con lo stesso nome, GitLab assegna priorità in base al seguente ordine (dal più alto al più basso):

1. Variabili definite manualmente durante l'esecuzione della pipeline.
2. Variabili segrete configurate a livello di progetto o gruppo.
3. Variabili definite nel file `.gitlab-ci.yml`.
4. Variabili predefinite di GitLab.

## Buone pratiche

1. **Usa variabili protette e mascherate per credenziali sensibili:** Evita di inserire informazioni sensibili direttamente nel file `.gitlab-ci.yml`.
2. **Centralizza le variabili a livello di gruppo o progetto:** Per evitare ridondanza e facilitare la manutenzione.
3. **Documenta le variabili:** Mantieni una lista aggiornata delle variabili usate nel progetto, con la descrizione del loro scopo.

4. **Limita l'accesso alle variabili:** Usa la funzione di protezione per branch/tag sensibili e maschera i valori nei log.
5. **Rendi le pipeline riutilizzabili:** Sfrutta le variabili per creare pipeline generiche che possono essere adattate a diversi scenari.

## Conclusione

Le variabili delle pipeline di GitLab sono uno strumento potente per rendere i processi CI/CD flessibili, sicuri e scalabili. Sfruttarle correttamente richiede una buona conoscenza delle loro caratteristiche e una gestione accurata, specialmente quando si tratta di dati sensibili. Con le best practice illustrate, puoi migliorare l'efficienza e la sicurezza delle tue pipeline, semplificando al contempo la gestione del codice e delle configurazioni.