



Faça o que eu fiz na aula

Agora que sabemos o que é um pipeline, vamos criar um arquivo na raiz de nosso projeto chamado `.gitlab-ci.yml`. É neste arquivo que configuramos todo o processo de CI/CD.

Então, vamos criar no arquivo `gitlab-ci` a primeira tarefa da pipeline. Um detalhe que temos que nos atentar é que os arquivos com extensão `.yml` são organizados com indentação de 2 espaços, ou seja, caso não respeitemos essa regra, o processo inteiro para.

```
job1:
  script:
    - echo "hello world"
```

Já vamos versioná-lo em nosso projeto. Para isso vamos dar um "commit" e um "push" desse arquivo. Abra um terminal e navegue até a pasta do seu projeto e em seguida digite:

```
git add --all
```

Para adicionar todos os arquivos modificados no projeto:

```
git commit -m "adicionando pipeline"
```

Para registrar uma mensagem e criar um hash de commit, marcando sua alteração

E por fim vamos dar um Push:

```
git push gitlab master
```

Após o push do arquivo `gitlab-ci.yml`, ao entrarmos novamente no menu CI/CD do gitlab, podemos verificar que o pipeline foi criado, e que já temos um "job" iniciado automaticamente, que possivelmente irá funcionar com sucesso. Agora, por que esse pipeline foi executado?

Bom, como criamos o arquivo `gitlab-ci.yml`, na raiz do projeto, toda vez em que realizarmos qualquer alteração e versionamos com gitlab, automaticamente o pipeline será executado.