

# Exercício (instruções): Comandos básicos do Git

---

## Objetivos e resultados

Neste exercício, o senhor se familiarizará com alguns comandos básicos do Git. Ao final do exercício, o senhor será capaz de:

- Configurar uma pasta como um repositório Git
- Executar operações básicas do Git em seu repositório Git

### Comandos básicos do Git

- Em um local conveniente em seu computador, crie uma pasta chamada **git-test**.
- Abra essa pasta git-test em seu editor favorito.
- Adicione um arquivo chamado *index.html* a essa pasta e adicione o seguinte código HTML a esse arquivo:

```
<!DOCTYPE html>
<html>
  <head></head>

  <body>
    <h1>This is a Header</h1>
  </body>
</html>
```

- Inicialização da pasta como um repositório Git
  - Vá para a pasta git-test em sua janela/terminal cmd e digite o seguinte no prompt para inicializar a pasta como um repositório Git:

```
git init
```

### Verificação do status do repositório Git

- Digite o seguinte no prompt para verificar o status do seu repositório Git:

```
git status
```

- Adicionar arquivos à área de preparação
  - Para adicionar arquivos à área de preparação do seu repositório Git, digite:

```
git add .
```

- Committing to the Git repository (Confirmar no repositório Git)
  - Para confirmar a área de preparação atual no repositório Git, digite:

```
git commit -m "first commit"
```

- Verificando o registro de commits do Git
  - Para verificar o registro dos commits no repositório Git, digite:

```
git log --oneline
```

- Agora, modifique o arquivo *index.html* da seguinte forma:

```
<!DOCTYPE html>
<html>
  <head></head>

  <body>
    <h1>This is a Header</h1>
    <p>This is a paragraph</p>
  </body>
</html>
```

- Adicione uma subpasta chamada **templates** à sua pasta **git-test** e, em seguida, adicione um arquivo chamado *test.html* à pasta *templates*. Em seguida, defina o conteúdo desse arquivo como sendo o mesmo do arquivo *index.html* acima.
- Em seguida, verifique o status e adicione todos os arquivos à área de preparação.
- Em seguida, faça o segundo commit em seu repositório.
- Agora, modifique o arquivo *index.html* da seguinte forma:

```
<!DOCTYPE html>
<html>
  <head></head>

  <body>
    <h1>This is a Header</h1>
    <p>This is a paragraph</p>
    <p>This is a second paragraph</p>
  </body>
</html>
```

- Agora, adicione o arquivo `index.html` modificado à área de preparação e faça uma terceira confirmação.

### Verificação de um arquivo de um commit anterior

- Para verificar o `index.html` do segundo commit, encontre o número do segundo commit usando o log do git e digite o seguinte no prompt:

```
git checkout <second commit's number> index.html
```

### Redefinir o repositório Git

- Para descartar o efeito da operação anterior e restaurar o *index.html* ao seu estado no final do terceiro commit, digite:

```
git reset HEAD index.html
```

- Em seguida, digite o seguinte no prompt:

```
git checkout -- index.html
```

- Também pode usar o **git reset** para redefinir a área de preparação para o último commit sem interferir no diretório de trabalho.

## Conclusões

Ao final deste exercício, vocês devem ter aprendido alguns comandos básicos do Git. Faça o exercício dado em sala para compreender totalmente como usar o Git.