

# **COMANDOS GITHUB**

**SABE QUAIS SÃO?**

**UM GUIA RÁPIDO DOS MAIS  
UTILIZADOS NOS PROJETOS**

---

# **INICIALIZAÇÕES**

---

CONFIGURAR USUÁRIO DO GITHUB AO SISTEMA:

`GIT CONFIG GLOBAL USER.NAME "[NOME]"`

- - -

CLONAR REPOSITÓRIO EXISTENTE:

`GIT CLONE [ENDEREÇO]`

- - -

CRIAR NOVO REPOSITÓRIO LOCAL:

`GIT INIT`

---

---

# **ACÕES DIÁRIAS**

---

PARA DEIXAR TUDO EM DIA:

`GIT PULL`

`GIT ADD .`

`GIT COMMIT -m "MENSAGEM"`

`GIT PUSH -u ORIGIN MASTER`

---

# MODIFICAÇÕES

---

VERIFICAR MUDANÇAS DO DIRETÓRIO DE TRABALHO:

`GIT STATUS`

- - -

COMPARAR ALTERAÇÕES ENTRE 2 BRANCHS:

`GIT DIFF [BRANCH1] [BRANCH2]`

- - -

ADICIONAR ARQUIVO AO PRÓXIMO COMMIT:

`GIT ADD [ARQUIVO]`

- - -

REALIZAR COMMIT COM MENSAGEM:

`GIT COMMIT -m "MENSAGEM"`

- - -

MARCAR O COMMIT ATUAL:

`GIT TAG [NOME]`

- - -

MODIFICAR BRANCH ATUAL:

`GIT SWITCH -c [PRÓXIMO] [ANTERIOR]`

- - -

ENVIAR ARQUIVOS ADICIONADOS AO REPOSITÓRIO REMOTO:

`GIT REMOTE ADD ORIGIN [URL]`

---

# HISTÓRICO

---

LISTAR CONFIGURAÇÕES DO REPOSITÓRIO:

`GIT REMOTE -V`

- - -

MOSTRAR INFORMAÇÕES DO REPOSITÓRIO:

`GIT REMOTE SHOW [NOME]`

- - -

VERIFICAR COMMITS REALIZADOS (Q PARA SAIR):

`GIT LOG`

- - -

VERIFICAR COMMITS DE UM ARQUIVO (Q PARA SAIR):

`GIT LOG [ARQUIVO]`

- - -

QUEM/QUANDO MUDOU UM ARQUIVO (Q PARA SAIR):

`GIT BLAME [ARQUIVO]`

---

# **DESFAZER MODIFICAÇÕES**

---

**ELIMINAR ARQUIVO:**

**GIT RM [ARQUIVO]**

- - -

**DESCARTAR MUDANÇAS LOCAIS:**

**GIT RESET --HARD HEAD**

- - -

**RETORNAR PONTO DE COMMIT (SEM DESCARTAR):**

**GIT RESET [COMMIT]**

- - -

**RETORNAR PONTO DE COMMIT (MANTER MUDANÇAS ATUAIS):**

**GIT RESET --KEEP [COMMIT]**

- - -

**RETORNAR PONTO DE COMMIT (DESCARTAR):**

**GIT RESET --HARD [COMMIT]**

- - -

**DESCARTAR MUDANÇAS DE UM ARQUIVO:**

**GIT CHECKOUT HEAD [ARQUIVO]**

- - -

**REVERTER COMMIT EFETUADO:**

**GIT REVERT [COMMIT]**

---

# **BRANCHES E TAGS**

---

LISTAR BRANCHES EXISTENTES:

**GIT BRANCH -AV**

- - -

MODIFICAR CABEÇALHO DA BRANCH:

**GIT CHECKOUT [BRANCH]**

- - -

CRIAR NOVA BRANCH COM BASE NA CORRENTE:

**GIT BRANCH [NOVABRANCH]**

- - -

CRIAR NOVA TRILHA:

**GIT CHECKOUT --TRACK [REMOTE/BRANCH]**

- - -

REMOVER BRACH LOCAL:

**GIT BRANCH -D [BRANCH]**

- - -

ELIMINAR BRACH REMOTA:

**GIT BRANCH -DR [REMOTE/BRANCH]**

---

# **PUBLICAÇÕES**

---

**BAIXAR MUDANÇAS (NÃO INTEGRA O CABEÇALHO):**

**GIT FETCH [REPOSITÓRIO]**

- - -

**BAIXAR MUDANÇAS (INTEGRA O CABEÇALHO):**

**GIT PULL [REPOSITÓRIO] [BRANCH]**

- - -

**PUBLICAR MUDANÇAS:**

**GIT PUSH -U ORIGIN [BRANCH]**

- - -

**PUBLICAR TAGS:**

**GIT PUSH --TAGS**

---

# **INTEGRAR ALTERAÇÕES DA RAMIFICAÇÃO**

---

**REALIZAR MESCLAGEM:**

**GIT MERGE [BRANCH]**

- - -

**REALIZAR REBASE:**

**GIT REBASE [BRANCH]**

- - -

**INTERROMPER REBASE:**

**GIT ABORT [BRANCH]**

- - -

**RESOLVER CONFLITOS PELA FERRAMENTA:**

**GIT MERGETOOL**