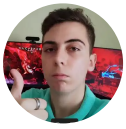

Victor Navarro
08/03/2024 14:41

Todos os Comandos Git: Uma Referência Completa 🏆

#Git

O Git é um sistema de controle de versão distribuído que permite aos desenvolvedores gerenciar e acompanhar mudanças no código-fonte de um projeto. Este artigo aborda uma lista completa dos comandos Git, divididos em categorias para facilitar a compreensão e o uso.

Configuração

Configurando o Git

- git config: Configura as opções de configuração para o Git.
- git config --global user.name "Seu Nome": Define o nome do usuário para todos os repositórios.

- git config --global user.email "seuemail@exemplo.com": Define o email do usuário para todos os repositórios.
- git config --list: Lista todas as configurações.

Configurando o Repositório

- git init: Inicializa um novo repositório Git.
- git clone: Clona um repositório existente.
- git clone <url>: Clona o repositório da URL especificada.

Trabalhando com Repositórios

Verificando o Status

- git status: Mostra o status do repositório, incluindo quais arquivos foram modificados.

Adicionando Arquivos

- git add: Adiciona arquivos ao índice.
- git add .: Adiciona todos os arquivos modificados.
- git add <arquivo>: Adiciona um arquivo específico.

Removendo Arquivos

- git rm: Remove arquivos do índice e do diretório de trabalho.
- git rm --cached <arquivo>: Remove o arquivo do índice, mas mantém o arquivo no diretório de trabalho.

Commitando Mudanças

- git commit: Grava as mudanças no repositório.
- git commit -m "Mensagem do commit": Commita as mudanças com uma mensagem.

Visualizando o Histórico

- git log: Mostra o histórico de commits.
- git log --oneline: Mostra o histórico de commits em uma linha.

Desfazendo Mudanças

- git reset: Redefine o índice e o diretório de trabalho para o estado de um commit específico.
- git reset --hard <commit>: Redefine o índice e o diretório de trabalho para o estado do commit especificado.

Branches e Tags

- git branch: Lista, cria ou exclui branches.
- git branch <nome>: Cria uma nova branch.
- git branch -d <nome>: Exclui a branch especificada.
- git checkout: Muda para a branch especificada.
- git checkout -b <nome>: Cria uma nova branch e muda para ela.
- git tag: Cria, lista, exclui ou verifica tags.
- git tag <nome>: Cria uma tag para o commit atual.

Trabalhando com Remotos

Adicionando Remotos

- git remote: Gerencia os remotos conectados ao repositório.
- git remote add <nome> <url>: Adiciona um novo remote.

Fetching e Pulling

- git fetch: Busca as mudanças do remote, mas não as mescla.
- git pull: Busca as mudanças do remote e as mescla.

Pushing

- git push: Envia as mudanças para o remote.
- git push <nome> <branch>: Envia as mudanças para o branch especificado no remote.

Rebasing

- git rebase: Reaplica commits em cima de outro branch.
- git rebase <branch>: Reaplica os commits do branch atual em cima do branch especificado.

Merging

- git merge: Combina as mudanças de outro branch.
- git merge <branch>: Combina as mudanças do branch especificado.

Stashing

- git stash: Salva temporariamente as mudanças não commitadas.
- git stash pop: Aplica as mudanças salvas e remove-as do stash.

Submodules

- git submodule: Gerencia submódulos.
- git submodule add <url> <caminho>: Adiciona um submódulo.
- git submodule update: Atualiza os submódulos.

Ferramentas e Configurações Avançadas

Git Hooks

- git hooks: Scripts executados automaticamente antes ou depois de eventos específicos do Git.

Git LFS

- git lfs: Gerencia arquivos grandes.
- git lfs install: Instala o Git LFS.
- git lfs track: Adiciona arquivos para serem rastreados pelo Git LFS.

Git Bisect

- git bisect: Encontra o commit que introduziu um bug.
- git bisect start: Inicia a busca pelo commit problemático.
- git bisect bad: Marca o commit atual como ruim.

- git bisect good <commit>: Marca o commit especificado como bom.

Git Subtree

- git subtree: Gerencia submódulos como submódulos de subárvore.

Git Subrepo

- git subrepo: Gerencia submódulos como submódulos de subrepositório.

Conclusão

O Git é uma ferramenta poderosa e versátil para o controle de versão. Este artigo forneceu uma visão geral dos comandos Git mais comuns, mas o Git tem muitos outros comandos e opções avançadas. Para uma compreensão mais profunda, consulte a documentação oficial do Git e recursos de aprendizado adicionais.

Comentários (2)

Rosângela Severo - 08/03/2024 18:39

Parabéns! Excelente artigo.

Matheus Paiva - 08/03/2024 15:35

Ótimo material Victor, parabéns!!!

Leia a seguir

Superando Desafios: A Jornada dos Novos Alunos no Campo do Desenvolvimento de Software

Leonardo Capra - 21 de Agosto

#GitHub #Git

Commits Semânticos - Seu github, seu código (e sua vida) muito melhores com ele

Jonas Campos - 21 de Agosto

#GitHub #Git #Java

Colaborando em Equipe com Git e Github

Lucas Santos - 14 de Agosto

#GitHub #Git

Mais informações

Central de Ajuda

Termos de Uso

Políticas de Privacidade

Canal de Contato LGPD

ACADEMIA PME EDUCACAO E CONSULTORIA EM NEGOCIOS LTDA. CNPJ: 26.965.884/0001-02