

Git-Cheat-Sheet: Wichtigste Befehle im Überblick

Einrichtung

```
git config --global user.name "Max Mustermann"  
git config --global user.email "max@example.com"  
git config --global core.editor "nano"
```

Repository erstellen oder klonen

```
git init # Neues Repo anlegen  
git clone # Bestehendes Repo klonen
```

Status, Staging und Commit

```
git status # Status anzeigen  
git add # Datei zum Staging hinzufügen  
git add . # Alle Änderungen hinzufügen  
git commit -m "Nachricht" # Commit erstellen
```

Unterschiede anzeigen

```
git diff # Arbeitsverzeichnis vs. Index  
git diff --staged # Index vs. letzter Commit
```

Remote-Repos

```
git remote -v # Remotes anzeigen  
git remote add origin # Remote hinzufügen  
git push -u origin main # Push + Tracking setzen  
git pull origin main # Änderungen holen + mergen
```

Branches

```
git branch # Branches anzeigen  
git switch # Branch wechseln  
git switch -c # Neuen Branch erstellen  
git merge # Branch mergen  
git branch -d # Branch löschen
```

Änderungen rückgängig machen

```
git restore # Datei zurücksetzen  
git restore --staged # Aus Staging entfernen  
git reset --hard # ⚠️ Alles zurücksetzen  
git revert # Commit rückgängig machen
```

Sonstiges

```
git log --oneline --graph # Kompakte Historie  
git stash # Arbeit zwischenspeichern  
git stash apply # Wiederherstellen  
git tag v1.0.0 # Version markieren  
git cherry-pick # Commit übernehmen  
git reflog # HEAD-Historie anzeigen
```

🌟 Aliase (optional)

```
git config --global alias.s "status -sb"
```

```
git config --global alias.lg "log --oneline --graph --all"
```

Hinweis: Alle Befehle gelten für die Kommandozeile (Bash, PowerShell, Terminal).

📄 Git-Cheat-Sheet herunterladen:

Für eine druckbare Version dieses Spickzettels kannst du das PDF direkt aufrufen:

👉 [Git-Befehle Cheat Sheet als PDF herunterladen](#)