

Installare Git

GitHub offre un client desktop con interfaccia grafica per svolgere le principali funzioni sui repository. È inoltre presente una versione auto-aggiornante di Git da linea di comando, per gli scenari più avanzati.

GitHub per Windows

windows.github.com

GitHub per Mac

mac.github.com

Le distribuzioni di Git per Linux e sistemi POSIX sono disponibili sul sito ufficiale di Git SCM.

Git per tutte le piattaforme

git-scm.com

Effettuare modifiche

Rivedi i cambiamenti al codice e prepara una commit

```
$ git status
```

Elenca tutti i file nuovi o modificati

```
$ git diff
```

Mostra le differenze non ancora nell'area di staging

```
$ git add [file]
```

Crea uno snapshot del file in preparazione al versioning

```
$ git diff --staged
```

Mostra le differenze tra staging e ultima modifica

```
$ git reset [file]
```

Rimuovi un file dall'area di staging, ma mantieni le modifiche

```
$ git commit -m"[descriptive message]"
```

Salva gli snapshot dei file in maniera permanente nello storico

Configurazione globale

Configurazione dell'utente valida per tutti i repository

```
$ git config --global user.name "[name]"
```

Imposta il nome che vuoi mostrare sulle tue commit

```
$ git config --global user.email "[email address]"
```

Imposta l'email che vuoi mostrare sulle tue commit

Creare repository

Crea un nuovo repository o clonane uno esistente da un URL

```
$ git init [project-name]
```

Crea un nuovo repository locale con il nome specificato

```
$ git clone [url]
```

Scarica un progetto esistente e il suo storico di cambiamenti

Refactoring dei nomi di file

Ricerca e rimuovi file dallo storico

Modifiche di gruppo

Aggrega una serie di commit all'interno di un branch

```
$ git branch
```

Elenca tutti i branch nel repository corrente

```
$ git branch [branch-name]
```

Crea un nuovo branch

```
$ git switch -c [branch-name]
```

Passa al branch specificato e aggiorna la directory corrente

```
$ git merge [branch-name]
```

Unisci lo storico del branch specificato con quello corrente

```
$ git branch -d [branch-name]
```

Elimina il branch specificato

Rivedere lo storico

Esplora l'evoluzione dei file del progetto

```
$ git rm [file]
```

Rimuovi un file dalla directory e prepara l'eliminazione definitiva

```
$ git rm --cached [file]
```

Elimina il file dallo storico di versione ma mantieni il file locale

```
$ git mv [file-original] [file-renamed]
```

Modifica il nome del file in preparazione a una commit

Escludere file dallo storico

Escludi file e percorsi temporanei

```
*.log  
build/  
temp-*
```

Un file di testo chiamato `.gitignore` previene il versioning accidentale di file o directory secondo un pattern specificato.

```
$ git ls-files --others --ignored --exclude-standard
```

Elenca tutti i file ignorati in questo progetto

```
$ git log
```

Elenca lo storico di versione per il branch corrente

```
$ git log --follow [file]
```

Elenca lo storico di versione per il file specificato, incluse rinominazioni

```
$ git diff [first-branch]...[second-branch]
```

Mostra la differenza tra due branch

```
$ git show [commit]
```

Mostra i metadati e i cambiamenti della commit specificata

Annullare commit

Elimina errori e altera lo storico dei cambiamenti

```
$ git reset [commit]
```

Annulla tutte le commit effettuate dopo `[commit]`, preservando i cambiamenti locali

```
$ git reset --hard [commit]
```

Elimina tutto lo storico e i cambiamenti fino alla commit specificata

Salvare frammenti

Archivia e ripristina cambiamenti incompleti

```
$ git stash
```

Archivia temporaneamente tutti i file modificati

```
$ git stash pop
```

Ripristina tutti i file modificati recentemente

```
$ git stash list
```

Elenca i set di cambiamenti archiviati

```
$ git stash drop
```

Elimina il set di cambiamenti più recente

Sincronizzare i cambiamenti

Collegati a un URL remoto e ottieni lo storico dei cambiamenti

```
$ git fetch [remote]
```

Scarica lo storico dei cambiamenti dal repository remoto

```
$ git merge [remote]/[branch]
```

Unisci il branch remoto con quello locale

```
$ git push [remote] [branch]
```

Carica tutti i cambiamenti dal branch locale su GitHub

```
$ git pull
```

Scarica lo storico e unisci i cambiamenti

Made with ❤️ by s and friends



COMMUNITY

© 2022 GitHub, Inc. Jekyll & Primer.