Spiegazione dei comandi su Git Bash utilizzati per salvare READMI su GitHub

* \*\*Comando:\*\* `cd "/c/Users/nesma.abd/Desktop/LINUX SERVER/backup\_home\_script"`

Accede alla cartella del progetto contenente i file da caricare su GitHub.

* \*\*Comando:\*\* `git init`

Inizializza un nuovo repository Git nella cartella. Se è già inizializzato, lo riattiva.

* \*\*Comando:\*\* `git add .`

Aggiunge tutti i file modificati o nuovi al prossimo commit (preparazione per il salvataggio).

* \*\*Comando:\*\* `git commit -m "messaggio"`

Crea un 'salvataggio' locale con i file aggiunti. Il messaggio descrive cosa è stato fatto.

* \*\*Comando:\*\* `git branch -M main`

Rinomina il ramo corrente in 'main', che è lo standard attuale su GitHub.

* \*\*Comando:\*\* `git remote add origin https://github.com/NesmaAbd/backup\_home\_script.git`

Collega il repository locale al repository remoto su GitHub.

* \*\*Comando:\*\* `git remote set-url origin https://github.com/NesmaAbd/backup\_home\_script.git`

Aggiorna l’URL del repository remoto se era stato inserito sbagliato in precedenza.

* \*\*Comando:\*\* `git push -u origin main`

Carica il contenuto del ramo 'main' sul repository remoto GitHub e imposta il tracking automatico.

* \*\*Comando:\*\* `git status`

Mostra lo stato attuale del repository: file modificati, nuovi, non tracciati, ecc.

* \*\*Comando:\*\* `git log`

Mostra la cronologia dei commit con autore, data e messaggio.

* \*\*Comando:\*\* `git commit -m 'messaggio'`

Nuovo commit per salvare modifiche successive, come aggiornamenti al README o altri file.