

Git Intro Hausaufgabe

Ziel: Diese Hausaufgabe soll euch helfen, die in der Klasse besprochenen Git-Konzepte zu vertiefen und anzuwenden. Ihr werdet ein kleines Python-Projekt erstellen und verwalten, um euer Verständnis von Git zu festigen.

Aufgabe 1: Git Repository initialisieren

1. Erstellt einen neuen Ordner auf eurem Computer für euer Python-Projekt, zum Beispiel `python-git-uebung`.
2. Öffnet ein Terminal und navigiert zu diesem Ordner.
3. Initialisiert ein neues Git-Repository mit dem Befehl `git init`.

Aufgabe 2: Einfaches Python-Skript erstellen und commiten

1. Im Ordner `python-git-uebung`, erstellt eine neue Python-Datei, zum Beispiel `hello.py`.
2. Schreibt ein einfaches Python-Programm, das "Hallo, Git!" in die Konsole ausgibt:

```
print("Hallo, Git!")
```

3. Speichert die Datei und fügt sie mit `git add hello.py` zum Repository hinzu.
4. Commitet eure Änderungen mit einem aussagekräftigen Kommentar, zum Beispiel `git commit -m "Erstes Python-Skript hinzugefügt"`.

Aufgabe 3: Branch für eine neue Funktion erstellen

1. Erstellt einen neuen Branch, um eine zusätzliche Funktion in eurem Python-Skript zu entwickeln. Verwendet zum Beispiel `git checkout -b neue-funktion`.
2. Ändert `hello.py`, um den Nutzer nach seinem Namen zu fragen und eine personalisierte Begrüßung auszugeben:

```
name = input("Wie heißt du? ")
print(f"Hallo, {name}, willkommen bei Git!")
```

3. Speichert die Änderungen, fügt sie hinzu (`git add -A`) und commitet sie mit einem passenden Kommentar, z.B. `git commit -m "Füge personalisierte Begrüßung hinzu"` .

Aufgabe 4: Änderungen in den Hauptbranch mergen

1. Wechselt zurück zum Hauptbranch mit `git checkout master` .
2. Führt einen Merge durch mit `git merge neue-funktion` , um die neuen Funktionen in den Hauptbranch zu integrieren.

Aufgabe 5: Dokumentation hinzufügen

1. Im Hauptbranch (`master`), erstellt eine neue Datei `README.md` .
2. Schreibt eine kurze Dokumentation für euer Projekt, die beschreibt, was das Skript tut und wie es ausgeführt wird.
3. Fügt die `README.md` Datei zum Repository hinzu, commitet und pusht sie.

Abgabe

Ladet euer fertiges Projekt in ein GitHub-Repository hoch und sendet den Link zu eurem Repository an euren Kursleiter. Stellt sicher, dass euer Repository öffentlich ist, damit euer Kursleiter darauf zugreifen kann.

Befehle, die dafür hilfreich sein können:

```
git remote add origin REMOTE-URL
```

```
git remote -v
```

```
git push -u origin master
```

Viel Erfolg!