

# Übungsblatt 1: Verfolgung von Git-Dokumenten und Ordnern

Ergänzungsfach Informatik

06.03.2024

## Ziel

Dieses Übungsblatt soll Ihnen helfen, Edits an Dokumenten und Ordnern in Git nachzuvollziehen.

## Instruktionen

1. **Vorbereitung:** Öffnen Sie das Terminal auf Ihrem Mac oder Windows-System.
2. **Übersicht im Terminal:** Verwenden Sie den Befehl `ls` (Mac) oder `dir` (Windows) im Terminal, um den Inhalt des Ordners anzuzeigen.
3. **Navigieren Sie zum gewünschten Ordner (Verzeichnis):**

```
cd /Pfad/zum/gewünschten/Ordner
```

4. **Git Repository initialisieren:** Das Initialisieren eines Git-Repositorys ermöglicht es uns, Git zur Versionskontrolle und Verwaltung unserer Dateien und Änderungen zu verwenden. Um ein Git-Repository zu initialisieren, führen wir den Befehl

```
git init
```

im gewünschten Verzeichnis aus. Dieser Befehl erstellt ein leeres Git-Repository im aktuellen Verzeichnis und ermöglicht es uns, Dateien zu verfolgen, Commits durchzuführen und mit einem Repository zu interagieren.

5. **Pfad des Git-Storages:** Überprüfen Sie den Pfad des versteckten `.git`-Ordners im Stammverzeichnis des Git-Repositorys. Versteckte Ordner können auf folgende Weise angezeigt werden:

- Mac: Verwenden Sie den Befehl `ls -a` im Terminal, um versteckte Ordner anzuzeigen.
- Windows: Verwenden Sie den Befehl `dir /a` im Befehlsfenster, um versteckte Ordner anzuzeigen.

#### 6. Um versteckte Ordner im Explorer anzuzeigen:

- Mac: Drücken Sie `Cmd + Shift + .` (Punkt) im Finder, um versteckte Dateien und Ordner anzuzeigen.
- Windows: Klicken Sie oben im Menü auf "Ansicht" und aktivieren Sie das Kontrollkästchen "Versteckte Elemente" im Abschnitt "Ein-/Ausblenden".

Nachdem Sie versteckte Dateien und Ordner sichtbar gemacht haben, sollten Sie den `.git`-Ordner im Verzeichnis des Git-Repositorys sehen können.

#### 7. Erstellen Sie ein leeres Textdokument:

Windows:

```
type nul > mein_text.txt
```

Mac:

```
touch mein_text.txt
```

#### 8. Text zum Dokument hinzufügen:

```
echo "Text" > mein_text.txt
```

#### 9. Ordner erstellen:

```
mkdir mein_ordner
```

#### 10. Datei löschen:

```
del mein_text.txt
```

#### 11. Ordner mit Inhalt löschen:

```
rmdir /s mein_ordner
```

## Git-Befehle

- **Git Status anzeigen:**

```
git status
```

- **Dateien zum Staging-Bereich hinzufügen:**

```
git add <Dateiname>
```

- **Commit-Änderungen mit einer Nachricht:**

```
git commit -m "Nachricht, die die Änderungen beschreibt"
```

Jetzt können Sie die oben genannten Schritte ausführen und die Git-Befehle verwenden, um Ihre Änderungen zu verfolgen und zu verwalten.