

Python-Projekt: Temporäre Dateien löschen mit Python

Lernziele

- Dateien in einem Ordner suchen
- Dateiendungen prüfen (Filter auf .tmp und .log)
- Dateien mit Python löschen
- Fehlerbehandlung einbauen
- Nützliches System-Tool entwickeln

Schritt 1 – „Hallo Cleanup“

Theorie:

Bevor wir anfangen, lassen wir das Programm starten.

Beispielcode:

```
print("Cleanup-Programm gestartet!")
```

Übung:

1. Schreibe ein Programm, das beim Start „Cleanup gestartet“ und am Ende „Cleanup beendet“ ausgibt.
2. Ergänze dein Programm so, dass dein Name und das aktuelle Datum angezeigt werden.

Schritt 2 – Dateien im Ordner auflisten

Theorie:

Mit `os.walk` können wir alle Dateien in einem Verzeichnis und seinen Unterordnern durchlaufen.

Beispielcode:

```
import os

target_dir = "/home/user/temp" # oder
"C:\\Users\\Rico\\AppData\\Local\\Temp"

for root, dirs, files in os.walk(target_dir):
    for file in files:
        print(file)
```

Übung:

1. Liste alle Dateien in deinem Temp-Ordner auf.
2. Ändere den Pfad so, dass du einen eigenen Test-Ordner mit Dateien verwendest.

Schritt 3 – Dateiendungen prüfen

Theorie:

Wir wollen nur bestimmte Dateitypen löschen – hier .tmp und .log Dateien.

Beispielcode:

```
if file.endswith(".tmp") or file.endswith(".log"):  
    print(f'{file} wird gelöscht.')
```

Übung:

1. Ergänze deinen Code so, dass nur .tmp und .log angezeigt werden.
2. Erzeuge eigene Testdateien (z. B. test1.tmp, test2.log, behalten.txt) und teste dein Programm.

Schritt 4 – Dateien löschen

Theorie:

Mit `os.remove()` können Dateien gelöscht werden.

Beispielcode:

```
os.remove(os.path.join(root, file))  
print(f'{file} gelöscht.')
```

Übung:

1. Ergänze deinen Code so, dass gefundene .tmp- und .log-Dateien tatsächlich gelöscht werden.
2. Überprüfe im Explorer, ob die Dateien wirklich verschwunden sind.

Schritt 5 – Fehler abfangen

Theorie:

Falls eine Datei gerade gesperrt ist, soll das Programm nicht abstürzen.

Beispielcode:

```
try:  
    os.remove(os.path.join(root, file))  
    print(f"{file} gelöscht.")  
except Exception as e:  
    print(f"Fehler beim Löschen von {file}: {e}")
```

Übung:

1. Provoziere einen Fehler (z. B. öffne eine .log-Datei in einem Editor und lasse das Programm laufen).
2. Stelle sicher, dass das Programm nicht abstürzt, sondern eine Fehlermeldung ausgibt.

Erweiterungsideen

- Schreibe ein Logfile (cleanup.log), in dem alle gelöschten Dateien protokolliert werden.
- Ergänze eine Eingabe (input()), damit der Benutzer den Zielordner selbst eingeben kann.
- Baue ein Menü mit zwei Optionen:
 1. „Cleanup starten“
 2. „Log anzeigen“