LU08.A01: Arbeiten mit Dateien und Verzeichnissen

Ziel: Verwenden Sie das `subprocess`-Modul in Python, um mehrere Bash-Befehle auszuführen und die Ergebnisse zu verarbeiten.

Aufgabe

Erstellen Sie ein Python-Skript, das folgende Schritte ausführt:

1. Erstellen eines neuen Verzeichnisses:

- 1. Das Skript soll ein neues Verzeichnis mit dem Namen `python_bash_test` im aktuellen Verzeichnis erstellen.
- 2. Überprüfen Sie, ob das Verzeichnis erfolgreich erstellt wurde.

2. Erstellen von Dateien im neuen Verzeichnis:

Erstellen Sie drei leere Dateien (`file1.txt`, `file2.txt`, `file3.txt`) im Verzeichnis
 `python bash_test`.

3. Auflisten der Dateien:

1. Listen Sie den Inhalt des Verzeichnisses `python_bash_test` mit dem Befehl `ls` auf und geben Sie die Ausgabe in der Konsole aus.

4. Dateigröße anzeigen:

1. Zeigen Sie die Größe der Dateien in diesem Verzeichnis mit dem Befehl `du -h` an.

5. Zusatz (optional):

1. Löschen Sie eine der Dateien (`file3.txt`) und geben Sie anschließend die aktualisierte Liste der Dateien aus.

Beispielcode für den Einstieg

Hier ist der Anfang des Skripts als Orientierungshilfe:

```python import subprocess

# 1. Verzeichnis erstellen def create\_directory(directory\_name):

```
result = subprocess.run(['mkdir', '-p', directory_name],
capture_output=True, text=True)
 if result.returncode == 0:
 print(f"Verzeichnis '{directory_name}' erfolgreich erstellt.")
```

```
else:
 print(f"Fehler beim Erstellen des Verzeichnisses: {result.stderr}")
```

# 2. Dateien erstellen def create\_files(directory\_name, file\_names):

```
for file name in file names:
 file_path = f"{directory_name}/{file_name}"
 result = subprocess.run(['touch', file path], capture output=True,
text=True)
 if result.returncode == 0:
 print(f"Datei '{file path}' erfolgreich erstellt.")
 print(f"Fehler beim Erstellen der Datei '{file path}':
{result.stderr}")
```

# Weiterer Code für Schritte 3-5 ```

#### Erwartete Ergebnisse

Beim Ausführen des Skripts sollte die Konsole folgendes ausgeben:

# 1. Nach Schritt 1:

```
Verzeichnis 'python_bash_test' erfolgreich erstellt.
```

# 2. Nach Schritt 2:

```
Datei 'python bash test/file1.txt' erfolgreich erstellt.
Datei 'python_bash_test/file2.txt' erfolgreich erstellt.
Datei 'python_bash_test/file3.txt' erfolgreich erstellt.
```

# 3. Nach Schritt 3 (Liste der Dateien):

```
Dateien im Verzeichnis 'python_bash_test':
file1.txt
file2.txt
file3.txt
```

# 4. Nach Schritt 4 (Dateigröße):

https://wiki.bzz.ch/ Printed on 2025/11/05 05:06

```
Größen der Dateien im Verzeichnis 'python_bash_test':

OB file1.txt

OB file2.txt

OB file3.txt
```

5. Nach Schritt 5 (Aktualisierte Liste nach Löschung):

```
Datei 'python_bash_test/file3.txt' wurde gelöscht.
Aktualisierte Dateien im Verzeichnis:
file1.txt
file2.txt
```

#### Bewertungskriterien

- **Korrektheit**: Das Skript führt alle Schritte ohne Fehler aus. - **Struktur**: Der Code ist modular und klar strukturiert (Verwendung von Funktionen). - **Fehlerbehandlung**: Rückgabecodes und Fehler werden überprüft und ordnungsgemäß verarbeitet.

From:

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/modul/m122/learningunits/lu08/aufgaben/dateien?rev=1733128230



