

Lernzettel: Python-Modul **rich** für farbige und schöne Ausgaben

1 Einführung

Das Python-Modul **rich** hilft dir, Konsolenausgaben bunt, übersichtlich und professionell zu gestalten. Es ist ideal für farbige Texte, Tabellen, Fortschrittsbalken und mehr. Dieser Lernzettel erklärt, wie du **rich** installierst und benutzt, um Ausgaben farbig zu formatieren, ohne Internetzugang.

2 Installation

Um **rich** zu nutzen, musst du es einmal installieren (wenn du offline bist, muss das auf einem Gerät mit Internet erfolgen, bevor du den Lernzettel nutzt):

```
1 pip install rich
```

Danach kannst du **rich** offline verwenden, solange Python und die Bibliothek installiert sind.

3 Grundlagen: Farbige Ausgaben

Das Herz von **rich** ist die **Console**-Klasse. Sie steuert, wie Text in der Konsole angezeigt wird. Du kannst Farben, Stile (fett, kursiv) und mehr hinzufügen.

```
1 from rich.console import Console
2 console = Console()
3 console.print("Hallo!", style="bold red")
```

Erklärung:

- `Console()`: Erstellt ein Objekt, um Ausgaben zu steuern.
- `console.print()`: Gibt Text aus, ähnlich wie `print()`.
- `style="bold red"`: Macht den Text fett und rot.

4 Farben und Stile

Du kannst Texte in verschiedenen Farben und Stilen ausgeben. Farben wie `red`, `green`, `blue`, `yellow` und Stile wie `bold`, `italic`, `underline` sind verfügbar.

```
1 console.print("Grün und kursiv", style="italic green")
2 console.print("Hintergrund gelb", style="on yellow")
```

Erklärung:

- Farben: red, green, blue, cyan, magenta, etc.
- Stile: bold, italic, underline, blink.
- Hintergrund: Füge on vor die Farbe, z. B. on blue.

Tipp: Kombiniere sie, z. B. style="bold italic on red".

5 Text mit Farben in einer Zeile

Du kannst verschiedene Teile eines Textes unterschiedlich formatieren, indem du [style] ... [/style] verwendest.

```
1 console.print("Das ist [bold red]wichtig[/bold red]!")
```

Erklärung: Der Text zwischen [bold red] und [/bold red] wird fett und rot angezeigt.

6 Tabellen erstellen

rich kann schöne Tabellen ausgeben, die farbig formatiert werden können.

```
1 from rich.table import Table
2 table = Table(title="Meine Tabelle")
3 table.add_column("Name", style="cyan")
4 table.add_row("Alice", style="green")
```

Erklärung:

- Table(): Erstellt eine Tabelle.
- add_column(): Fügt eine Spalte hinzu, z. B. mit Titel und Stil.

7 Fortschrittsbalken

Für langlaufende Aufgaben kannst du Fortschrittsbalken anzeigen.

```
1 from rich.progress import Progress
2 with Progress() as progress:
3     task = progress.add_task("[green]Arbeiten...", total=100)
4     progress.update(task, advance=10)
```

Erklärung:

- Progress(): Startet einen Fortschrittsbalken.
- add_task(): Erstellt eine Aufgabe mit Gesamtfortschritt. update(): Aktualisiert den Fortschritt (z. B. + 10%).

8 Fehlerbehandlung

Wenn etwas nicht funktioniert:

- **Modul nicht gefunden:** Stelle sicher, dass `rich` installiert ist (`pip install rich`).
- **Falsche Farbe/Stil:** Überprüfe die Schreibweise, z. B. `red` statt `rot`.
- **Konsole sieht komisch aus:** Manche Terminals (z. B. alte Windows CMD) unterstützen keine Farben gut. Verwende moderne Terminals wie PowerShell oder VS Code.

9 Tipps für den Offline-Gebrauch

- Speichere diesen Lernzettel als PDF oder drucke ihn aus.
- Teste die Beispiele in einer Python-Konsole (z. B. IDLE oder VS Code).
- Häufige Farben: `black`, `red`, `green`, `yellow`, `blue`, `magenta`, `cyan`, `white`.
- Häufige Stile: `bold`, `italic`, `underline`, `blink`, `reverse`.