LU10b - Komplexe Argumente

Das Modul `argparse` ermöglicht es, benannte Argumente, Pflichtargumente und optionale Argumente zu definieren und dabei die Benutzerfreundlichkeit zu erhöhen.

Das Modul argparse wird unter anderem beim Python-Webapplikationen eingesetzt, um die Eingabewerte in einem Formular zu prüfen. Bei systemnahen Scripts nutzen wir vor allem die Möglichkeiten Parameter mit Schlüsselwörtern (z.B. `-n 3`) und optionale Parameter zu definieren.

Wichtige Funktionen von argparse

- `add_argument()`: Definiert, welche Argumente akzeptiert werden, ob sie erforderlich sind oder ob sie Optionen (mit Präfix wie -v oder –verbose) sind.
- `parse_args()`: Parst die Argumente und gibt ein Objekt zurück, mit dem Sie auf die übergebenen Werte zugreifen können.
- `action=„store_true“`: Aktiviert eine Option ohne zusätzlichen Wert, z. B. für Schalter wie –verbose.
- `help`: Kann für die automatische Generierung von Hilfetexten für die Argumente verwendet werden.

Beispiel

```
import argparse

def main():

    # Create the ArgumentParser
    parser = argparse.ArgumentParser(description='Simple Example')

    # Add some arguments
    parser.add_argument('arg1', help='First argument')
    parser.add_argument('arg2', help='Second argument')
    parser.add_argument('-v', '--verbose', help='Shows more information
about the scripts', action='store_true')

    # Parse the argument
    args = parser.parse_args()

    # Print the arguments
    print(f'Argument 1: {args.arg1}')
    print(f'Argument 2: {args.arg2}')

    # Check the verbose switch
    if args.verbose:
```

```
        print('Verbose mode active!')

if __name__ == '__main__':
    main()
```

M122-LU10

Marcel Suter

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

Permanent link:
https://wiki.bzz.ch/modul/m122/learningunits/lu10/argparse

Last update: **2025/01/08 09:06**