

Imperatív programozás

12. Előadás

Argumentumok kezelése

Szkriptek készítése

Parancssori argumentumok kezelése

az argparse modul segítségével

- új argumentumkezelő létrehozása

`argparse.ArgumentParser()`

```
import argparse
```

```
parser = argparse.ArgumentParser(description='Argumentumkezelő példa.',  
                                epilog='Példa program.', prefix_chars='-+')
```

- lehetséges paraméterek beállítása

```
parser.add_argument()
```

- argumentumok eltárolása

- akciók végrehajtása

```
parser.add_argument('--verbose', dest='verb', action='store_const',  
                    const=True, default=False, help='Setting verbose mode.')
```

```
parser.add_argument('+i', dest='params', action='append_const',  
                    const="42", help='Adding 42.')
```

```
parser.add_argument('+s', dest='params2', action='append',  
                    help='Adding string arguments.')
```

```
parser.add_argument('str_args', type=str, nargs='*', metavar='Str',  
                    help='Arguments of the program.')
```

- paraméterek elérése

```
args = parser.parse_args()
```

```
args.verb      # verbose mode
```

```
args.params    # storing 42 values
```

```
args.params2   # storing string parameters
```

```
args.str_args  # additional string arguments
```

Teljes példa

argparse1.py

./argparse1.py

./argparse1.py szoveg1

./argparse1.py --v

./argparse1.py --v szoveg1 szoveg2

./argparse1.py -h

./argparse1.py -t

./argparse1.py --v +i +i +s egy +s kettő három négy

Teljes szkript példa

Egyszerű grep program

[gr.py](#)

Operációs rendszer szintű parancsok

- os modul
 - környezeti változók kezelése
 - fájlok, könyvtárak listázása, létrehozása, törlése
 - parancsok végrehajtása
 - teljes könyvtárstruktúra bejárása
- os.path modul
 - útvonal-kifejezések kezelése

példaprogram: [os1.py](#)

Teljes szkript példa

Egyszerű locate program

[locate.py](#)