Opzioni da linea di comando

Il modulo optparse

- Python mette a disposizione il modulo optparse per la definizione ed il parsing delle opzioni da riga di comando

```
from optparse import OptionParser
```

- La configurazione delle opzioni è gestita attraverso l'oggetto OptionParser()

```
parser = OptionParser()
```

Aggiunta di opzioni

- Le opzioni sono inserite tramite il metodo add_option() dell'oggetto OptionParser

```
parser.add_option("-f", "--file", dest="filename",
    help="write report to FILE", metavar="FILE")
parser.add_option("-q", "--quiet",
    action="store_false", dest="verbose", default=True,
    help="don't print status messages to stdout")
```

- Parametri:
  - “-f”: opzione breve
  - “--file”: opzione lunga
  - “dest=filename”: il nome della variabile in cui sarà salvato il valore dell'opzione
  - default=value: valore di default
  - help=”...”: messaggio di help

Aggiunta di opzioni

- Le opzioni sono inserite tramite il metodo add_option() dell'oggetto OptionParser

```
parser.add_option("-f", "--file", dest="filename",
    help="write report to FILE", metavar="FILE")
parser.add_option("-q", "--quiet",
    action="store_false", dest="verbose", default=True,
    help="don't print status messages to stdout")
```

- Parametri:
  - action=”...”: specifica come gestire il parametro (flag, valore variabile, valore di un array)
    - “store”: memorizzazione semplice
    - “store_true/store_false”: memorizza un flag true/false
    - “append”: inserimento valore in array

Aggiunta di opzioni

- Una volta impostate le opzioni, si invoca la funzione di parsing parse_args(), che restituisce una tupla (options, args)
  - options: dizionario contenente le variabili con i valori
  - args: lista contenente i parametri fissi

```
(options, args) = parser.parse_args()
value = options.value
```