

Python - Übungsblatt 2

Gerald Senarclens de Grancy

March 9, 2007

Übung 1 - Built-in Typen

Starte einen Python Interpreter (entweder Idle oder in der Kommandozeile) und experimentiere mit folgenden Eingaben und versuche, die Resultate zu verstehen und erklären.

Strings (immutable)

```
“Hello” + “World”
S = “informatics”; S
“Bio” + S
S * 5
S[3], S[0:3], S[:3], S[:0]
S[:-2], S[:]
“%slogy is as hard as %s” % (“Bio”, S)
S[3] = ‘a’
help(str)
```

Hinweis: Strings sind immutable und erlauben daher keine Zuweisung von Werten in bestehende Tuples!

Tuples (immutable)

```
(3, ‘x’)
(3, ‘x’,)
T = (3, ‘x’,); T
T[0], T[0:]
T[0] = 2
help(()) bzw. help(tuple)
```

Hinweis: Tuples sind immutable und erlauben daher keine Zuweisung von Werten in bestehende Tuples!

Lists (mutable)

Listen sind im Gegensatz zu Tuples mutable. Man kann daher bestehende Listen verändern.

```
L = [1,2,3]; L
L = [1,2,3,]; L
L = [1,2,3] + [4,5,6]; L
L, L[:]
L[:0], L[-1], L[-3], L[-2:]
L[1:3], L[3:1]
(['a', 'b', 'c'] + ['d', 'e', 'f'])[2:4]
[L[3], L[4]]
L[-100:100]
L.reverse(); L
L.index(4)
L.sort(); L
L.index(4)
['', [], (), {}, 0, None]
L[1] = []; L
L[1:3] = []; L
help([]) bzw. help(list)
```

Dictionaries (mutable)

```
{'a':1, 'b':2, 'c':3}
{'a':1, 'b':2, 'c':3}['b']
D = {'alpha':1, 'beta':2, 'gamma':3}; D
D['beta'] = 5; D
D['alpha'] + D['gamma']
D['delta'] = 4; D
D.keys()
D.values()
D.has_key('alpha')
help(dict) bzw. help()
```

Übung 2 - Fehlermeldungen

Die folgenden Eingaben führen im Python Interpreter zu Fehlermeldungen. Versuche diese Fehlermeldungen zu verstehen.

```
T[0] = 2
{'a':1, 'b':2, 'c':3}[1]
S[3] = 'a'
L[24]
L.has_key()
D.has_key()
D.sort()
L[1:3]=0
L + S
```

Übung 3 - Dateien

Erstelle eine neue Textdatei in Scite und schreibe darin ca. zwei bis drei Zeilen Text (Eingabetaste am Ende jeder Zeile drücken). Versuche danach, diese Datei im Python Interpreter auszulesen:

```
file = open('filename', 'r')
contents = file.read()
file.close()
print contents
```

```
file = open('filename', 'r')
file.read(10)
file.close()
```

```
file = open('filename', 'r')
file.readline()
file.readline()
file.close()
```

```
file = open('filename', 'r')
contents = file.readlines()
file.close()
print contents
```

Jetzt versuche, den Inhalt dieser Datei zu überschreiben. Benutze dazu folgende Methoden:

```
file = open('filename', 'w')
file.write(string)
file.writelines(stringlist)
file.close()
```

Anschließend versuche, statt die bestehende Datei zu überschreiben, diese zu editieren oder an sie anzufügen. Wie könnte das gehen? Unter Umständen nützliche sind

```
file.closed, file.mode, file.name
```