

Seminar 2

1. Funktionen

Eine Funktion in Python wird durch eine `def` Anweisung definiert. Die allgemeine Idee besteht darin, unsere Programme in einzelne Funktionen zu unterteilen, die einen bestimmten Zweck haben und mit ihrer Interaktion das vorliegende Problem lösen. Werte werden mithilfe von Parametern zwischen Funktionen ausgetauscht.

Die allgemeine Syntax sieht folgendermaßen aus:

```
def <name> (<Parameter>):  
    <Anweisungen>
```

Die Parameterliste besteht aus keinem oder mehreren Parametern.

```
def square (x): #x - formaler Parameter  
    return x*x  
  
def add_to_list (target, element):  
    target.append(element)  
  
def main ():  
    print (square (4)) #Aufruf der Funktion. 4 ist der tatsächliche Parameter  
  
    l = [] #l ist eine leere Liste  
    add_to_list(l, square(2))  
    print (l) #output: [4]
```

2. Übungen

1. Schreiben Sie eine Funktion, die ein String aufnimmt und daraus ein Dictionary erstellt. Der Schlüssel ist ein Buchstabe und der Wert ist die Anzahl seiner Erscheinungen in dem String.

Beispiel: school
{ 's': 1, 'c': 1, 'h': 1, 'o': 2, 'l': 1 }

2. Schreiben Sie eine Funktion, die ein String aufnimmt und in ein HTML-Tag einschließt. Das Tag wird als Parameter angegeben.

Beispiel:
`add_tags ('i', 'Python') => '<i> Python </i>'`

3. Schreiben Sie eine Funktion, die prüft, ob ein übergebener String palindrom ist.



4. Schreiben Sie eine Funktion mit folgender Deklaration von
`def str_find (target, source);`

Die Funktion soll den Index des ersten Auftreten von `source` in `target` (und sonst -1) zurückgeben.

Beispiel: `str_find("testing","ing") => 5`

5. Schreiben Sie eine Funktion, die einen Text aufnimmt und mit einer Caesar-Chiffre verschlüsselt. Wikipedia:
https://en.wikipedia.org/wiki/Caesar_cipher