

## Lösung zu Aufgabe 1 (Fibonacci Kaninchen)

```
1 # author: Johann-Mattis List
2 # date: 12.05.2011
3 # file: fibonacci.py
4
5
6
7 def fibonacci(number):
8     """
9     Dies ist die klassische Lösung, die sich fast überall im Internet finden
10    lässt. Dummerweise habe ich vergessen, die Quelle zu dokumentieren und
11    bin
12    jetzt zu faul, das zu tun.
13    """
14    # Dies ist die Bedingung für das Terminieren der Funktion. Wenn der Wert
15    1
16    # oder der Wert 0 eingegeben wird, wird der gleiche Wert zurückgegeben
17    und
18    # die Rekursion gestoppt.
19    if number == 0 or number == 1:
20        return number
21    # Dies ist die Rekursion. Da die Fibonaccizahlen immer aus den zwei
22    # Vorgängern berechnet werden, welche wiederum aus ihren zwei
23    # Vorgängern berechnet werden, muss die Funktion zwei mal
24    # aufgerufen
25    # werden.
26    else:
27        return fibonacci(number-1) + fibonacci(number-2)
```

## Lösung zu Aufgabe 2 (Fibonacci Kaninchen)

```
1 <html>
2 <head>
3   <title>Fibonacci's Kanickel</title>
4   <meta http-equiv="content-type" content="text/html; charset=utf-8">
5   <link rel="stylesheet" type="text/css" href="<+>" />
6 </head>
7 <body>
8   <label>How many generations do you want to breed your rabbits?</label>
9   <input type="number" value="5" max="10" min="1" id="kanickel"/>
10  <br>
11  <button onclick="fiboStart()">GO!</button>
12  <div id="kanickelfarm"></div>
13  <script>
14  function fiboStart() {
15
16    /* get the number of generations */
```

```
17  var ngen = document.getElementById('kanickel').value;
18
19  /* teste, ob ngen kleiner als 2 ist, dann lassen wir das ganze */
20  if (ngen < 2) {
21      return ngen;
22  }
23
24  /* wenn nicht, dann machen wir einen schönen array fertig */
25  var a = [0, 1];
26
27  /* wir bereiten unsere schönen kaninchenbilder vor */
28  var images = '';
29
30  /* jetzt ist es einfach, wir iterieren durch den array und expan-
31     dieren ihn mit seinen eigenen werten */
32  for (var i=1; i<ngen; i++) {
33      a.push(a[a.length-1] + a[a.length-2]);
34
35      /* jetzt fügen wir bilder hinzu */
36      var image = '';
37      for (var j=0; j<a[a.length-1]; j++) {
38          image += '';
39      }
40      images += "<p style=\"border:2px solid lightgray\">"+image+"</p>";
41  }
42
43  /* get the kanickelfarm */
44  var kf = document.getElementById('kanickelfarm');
45  kf.innerHTML = images;
46
47  }
48  </script>
49  </body>
50  </html>
```

Die Applikation kann unter <http://lingulist.de/pyjs/demos/fibonacci.html> getestet werden.