

Связь замыканий и генераторов

```
8 <h1>Замыкание: нутро генератора</h1> <script type="text/javascript"> //
9     (()=&gt;{
10         function*s(){let i=0;while(1){yield ++i}}
11         let c1 = function(){ let n=0; return function (){return ++n} };
12         let c2 = function(){ let n=0; return function (){return ++n} }();
13         let c3 = s();
14
15         /*замыкание НЕ работает: вторая функция при каждом вызове создаётся заново*/
16         Out.log( c1()() );           //1
17         Out.log( c1()() );           //1
18         Out.log( c1()() );           //1
19         /*замыкание работает: вторая функция создана один раз и получила доступ к своему замыканию!!!*/ Out.br();
20         Out.log( c2() );              //1
21         Out.log( c2() );              //2
22         Out.log( c2() );              //3
23
24         Out.log( c3.next().value );    //1
25         Out.log( c3.next().value );    //2
26         Out.log( c3.next().value );    //3
27
28         /* создавая экземпляр генератора, мы фактически создаём такую вот вторую функцию */
29     })();</pre></div><div data-bbox="47 549 461 590" data-label="Section-Header"><h2>Замыкание: нутро генератора</h2></div><div data-bbox="47 643 59 860" data-label="Text"><pre>1
1
1

1
2
3

1
2
3</pre></div><div data-bbox="29 932 173 951" data-label="Page-Footer"><p>(c)2013 Gossoudarev. - <a href="#">About</a></p></div>
```