

```
1 import Zombie from 'zombie';
2
3 const URL = 'https://kodaktor.ru/g/testing_f38ac';
4
5 const page = new Zombie();
6
7 const goURL = new Promise(      page.visit(URL, err => (err) ? rej(err) : res())      |      );
8
9 goURL(URL+'?test=ilya@gmail.com')
10 .then ( ()=>console.log(page.document.querySelector('h3#response').textContent)      )
11 .catch( err=>console.error(err));
12
```

```
$ ./node_modules/babel-cli/bin/babel-node.js zomb1.js
/Users/eliasgoss/app06-09-2017/zomb1.js:13
var goURL = new Promise(page.visit(URL, function (err) {
                        ^
TypeError: Promise resolver null is not a function
    at Promise (<anonymous>)
    at Object.<anonymous> (/Users/eliasgoss/app06-09-2017/zomb1.js:7:15)
```

```
1 import Zombie from 'zombie';
2
3 const URL = 'https://kodaktor.ru/g/testing_f38ac';
4
5 const page = new Zombie();
6
7 const goURL = new Promise( (res,rej)=>page.visit(URL, err => (err) ? rej(err) : res()) );
8
9 goURL(URL+'?test=ilya@gmail.com')
10 .then( ()=>console.log(page.document.querySelector('h3#response').textContent) )
11 .catch( err=>console.error(err));
12
```

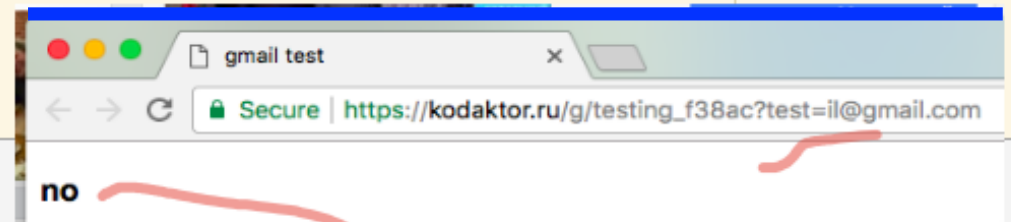
```
$ ./node_modules/babel-cli/bin/babel-node.js zomb1.js
/Users/eliasgoss/app06-09-2017/zomb1.js:19
goURL(URL + '?test=ilya@gmail.com').then(function () {
^
TypeError: goURL is not a function
    at Object.<anonymous> (/Users/eliasgoss/app06-09-2017/zomb1.js:9:1)
```

```
1 import Zombie from 'zombie';
2
3 const URL = 'https://kodaktor.ru/g/testing_f38ac';
4
5 const page = new Zombie();
6
7 const goURL = URL=>new Promise( (res,rej)=>page.visit(URL, err => (err) ? rej(err) : res()) );
8
9 goURL(URL+'?test=ilya@gmail.com')
10 .then ( ()=>console.log(page.document.querySelector('h3#response').textContent) )
11 .catch( err=>console.error(err));
12
```



```
$ ./node_modules/babel-cli/bin/babel-node.js zomb1.js
yes
→ eliasgoss@ ~/app06-09-2017
$
```

```
1 import Zombie from 'zombie';
2 const URL = 'https://kodaktor.ru/g/testing_f38ac';
3 const page = new Zombie();
4 const g=URL=>new Promise((rs,rj)=>page.visit(URL,e=>(e)?rj(e):rs()));
5 const vals = [
6   {val: "?test=ilya@gmail.com", expected:"yes"},
7   {val: "?test=il@gmail.com", expected:"no"},
8   {val: "?test=ilya@gmail.com", expected:"yes"}
9 ];
10 vals.forEach(o=>
11   g(URL+o.val)
12   .then (()=>console.log(URL+o.val, page.document.querySelector('h3#response').textContent))
13   .catch(e=>console.error(e))
14 );
15
```



```
$ ./node_modules/babel-cli/bin/babel-node.js zombprom4.js
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=il@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
```

почему всюду yes?

```
1 import Zombie from 'zombie';
2 const URL = 'https://kodaktor.ru/g/testing_f38ac';
3 const page = new Zombie();
4 const g=URL=>new Promise((rs,rj)=>page.visit(URL,e=>(e)?rj(e):rs()));
5 const vals = [
6   {val: "?test=ilya@gmail.com", expected:"yes"},
7   {val: "?test=il@gmail.com", expected:"no"},
8   {val: "?test=ilya@gmail.com", expected:"yes"}
9 ];
10 for (const o of vals) {
11   g(URL+o.val)
12   .then (()=>console.log(URL+o.val, page.document.querySelector('h3#response').textContent))
13   .catch(e=>console.error(e))
14 }
15
```

императивный вариант  
то же самое

```
$ ./node_modules/babel-cli/bin/babel-node.js zombprom5.js
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=il@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
```

```
1 import Zombie from 'zombie';
2 const URL = 'https://kodaktor.ru/g/testing_f38ac';
3 const page = new Zombie();
4 const g=URL=>new Promise((rs,rj)=>page.visit(URL,e=>(e)?rj(e):rs()));
5 const vals = [
6   {val: "?test=ilya@gmail.com", expected:"yes"},
7   {val: "?test=il@gmail.com", expected:"no"},
8   {val: "?test=ilya@gmail.com", expected:"yes"}
9 ];
10 for (const o of vals) {
11   (async()=>{
12     await g(URL+o.val);
13     console.log(URL+o.val, page.document.querySelector('h3#response').textContent)
14   })();
15 }
```

введение async/await  
не помогло...  
почему?

```
$ ./node_modules/babel-cli/bin/babel-node.js zombprom6.js
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=il@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
```

```
1 import Zombie from 'zombie';
2 const URL = 'https://kodaktor.ru/g/testing_f38ac';
3 const page = new Zombie();
4 const g=URL=>new Promise((rs,rj)=>page.visit(URL,e=>(e)?rj(e):rs()));
5 const vals = [
6   {val: "?test=ilya@gmail.com", expected:"yes"},
7   {val: "?test=il@gmail.com", expected:"no"},
8   {val: "?test=ilya@gmail.com", expected:"yes"}
9 ];
10 (async()=>{
11   for (const o of vals) {
12     await g(URL+o.val);
13     console.log(URL+o.val, page.document.querySelector('h3#response').textContent);
14   }
15 })();
```

а в предыдущем примере  
были три асинхронные  
и брался результат  
работы последней

всё дело в том, что  
цикл целиком в одной  
асинхронной лямбде

ура!

```
$ ./node_modules/babel-cli/bin/babel-node.js zombprom7.js
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
https://kodaktor.ru/g/testing_f38ac?test=il@gmail.com no
https://kodaktor.ru/g/testing_f38ac?test=ilya@gmail.com yes
```