async/await使用同步的方式来书写异步代码，将异步调用的难度降低到接近于0，未来必将大放异彩。然而在当下，由于标准化的缓存步伐，async/await尚在ES7的草案中。为了尝先，特试用了下面两种方式：

* 使用社区提供的asyncawait封装
* 使用ES7草案

## 使用社区提供的asyncawait模块

### Git地址

[git@github.com:yortus/asyncawait.git](mailto:git@github.com:yortus/asyncawait.git)

### 使用方法:

1. 安装node模块
   1. npm install [asyncawait@1.0.3](mailto:asyncawait@1.0.3) –save
2. 创建示例类AsyncService.js

**var *async*** = *require*(**'asyncawait/async'**);  
**var *await*** = *require*(**'asyncawait/await'**);  
**var *sleep*** = ***async***(  
 **function** *sleep*(timeout) {  
 **return new *Promise***(**function** (resolve, reject) {  
 setTimeout(**function** () {  
 resolve();  
 }, timeout);  
 });  
 }  
);  
  
(***async***(  
 **function** () {  
 **console**.log(**'Do some thing, '** + **new** Date());  
 ***await***(***sleep***(3000));  
 **console**.log(**'Do other things, '** + **new** Date());  
 }  
))();

1. 运行AsyncService.js
   1. node AsyncService.js
   2. 运行结果：

Do some thing, Wed Jun 15 2016 11:09:05 GMT+0800 (中国标准时间)

Do other things, Wed Jun 15 2016 11:09:08 GMT+0800 (中国标准时间)

### 注意事项

1. asyncawait模块内部引用bluebird模块.
2. 无须编译为Es5，可直接运行.

## 使用ES7草案

### 使用方法:

1. 安装node模块，需要的一系列模块如下：
   1. **babel-cli**
   2. **babel-preset-es2015"**
   3. **babel-preset-react"**:
   4. **babel-preset-stage-3**
   5. **babel-polyfill**
2. 创建示例类 AsyncAwaitService.js

**async function** *sleep*(timeout) {  
 **return new *Promise***((resolve, reject) => {  
 setTimeout(**function** () {  
 resolve();  
 }, timeout);  
 });  
}  
  
(**async function** () {  
 **console**.log(**'Do some thing, '** + **new** Date());  
 **await** *sleep*(3000);  
 **console**.log(**'Do other things, '** + **new** Date());  
})();

1. 编译AsyncAwaitService.js
   1. 配置babel
      1. 在package.json中加入babel节点，内容如下：

**"babel"**: {  
 **"presets"**: [  
 **"es2015"**,  
 **"react"**,  
 **"stage-3"** ],  
 **"plugins"**: []  
}

* 1. 编译

babel AsyncAwaitService.js --out-file AsyncAwaitService\_es5.js

或者

babel AsyncAwaitService.js -o AsyncAwaitService\_es5.js

* 1. 标记编译后的代码

在AsyncAwaitService\_es5.js脚本头部加入以下代码：

*require*(**'babel-polyfill'**)

1. 运行AsyncAwaitService\_es5.js
   1. node AsyncAwaitService\_es5.js
   2. 运行结果：

Do some thing, Wed Jun 15 2016 11:54:13 GMT+0800 (中国标准时间)

Do other things, Wed Jun 15 2016 11:54:16 GMT+0800 (中国标准时间)

### 注意事项

1. async/await通过babel编译为Es5，方可直接运行.
2. babel编译相关内容可参考阮一峰博客 http://www.ruanyifeng.com/blog/2016/01/babel.html