

Egg.js 控制器基类 BaseController、控制器兼容写法

主讲教师：（大地）

合作网站：www.itying.com （IT 营）

我的专栏：<https://www.itying.com/category-79-b0.html>

- 一、 Egg.js 控制器基类 BaseController..... 1
- 二、 Egg.js 控制器兼容写法（不推荐使用，只是为了兼容） 3

一、Egg.js 控制器基类 BaseController

按照类的方式编写 Controller，不仅可以让我们更好的对 Controller 层代码进行抽象（例如将一些统一的处理抽象成一些私有方法），还可以通过自定义 Controller 基类的方式封装应用中常用的方法。

```
const { Controller } = require('egg');

class BaseController extends Controller {

  get user() {

    return this.ctx.session.user;

  }

  success(data) {

    this.ctx.body = {

      success: true,
```

```
    data,  
  
    };  
  
}  
  
notFound(msg) {  
  
    msg = msg || 'not found';  
  
    this.ctx.throw(404, msg);  
  
}  
  
}  
  
module.exports = BaseController;
```

```
const Controller = require('../core/base_controller');  
  
class PostController extends Controller {  
  
    async list() {  
  
        const posts = await this.service.listByUser(this.user);  
  
        this.success(posts);  
  
    }  
  
}
```

二、Egg.js 控制器兼容写法（不推荐使用，只是为了兼容）

```
'use strict';

const Controller = require('egg').Controller;

class HomeController extends Controller {

  async index(ctx) {

    await ctx.render('home');

  }

}

module.exports = HomeController;
```