

学习 Koa 之前的一些准备工作 Es6 的一些常用语法以及 Promise async await 的使用方法

主讲教师：（大地）

合作网站：<http://www.itying.com>

中文文档：<http://www.itying.com/koa>

目录

一、 Es6 常见语法的使用。	1
二、 Koa 异步处理 Async 、 Await 和 Promise 的使用(重点*)	2

一、 Es6 常见语法的使用。

1. let const
2. 箭头函数
3. 对象、属性的简写
4. 模板字符串
5. Promise

二、Koa 异步处理 Async 、Await 和 Promise 的使用(重点*)

async 是“异步”的简写，而 await 可以认为是 async wait 的简写。所以应该很好理解 async 用于申明一个 function 是异步的，而 await 用于等待一个异步方法执行完成。

简单理解：

async 是让方法变成异步。

await 是等待异步方法执行完成。

详细说明：

async 是让方法变成异步，在终端里用 node 执行这段代码，你会发现输出了 Promise { 'Hello async' }，这时候会发现它返回的是 Promise。

```
async function testAsync(){
  return 'Hello async';
}
const result = testAsync();
console.log(result);
```

```
PS E:\code\BXShop> node async.js
Promise { 'Hello async' }
```

await 在等待 async 方法执行完毕，其实 await 等待的只是一个表达式，这个表达式在官方文档里说的是 Promise 对象，但是它也可以接受普通值。 **注意：**await 必须在 async 方法中才可以使用因为 await 访问本身就会造成程序停止堵塞，所以必须在异步方法中才可以使用。

```
function getData(){
  return 'zhangsan';
}
async function testAsync(){
  return 'Hello async';
}
```

```
async function test(){
  const v1=await getData();
  const v2=await testAsync();
  console.log(v1,v2);
}
test();
```

async/await 同时使用

async 会将其后的函数（函数表达式或 Lambda）的返回值封装成一个 Promise 对象，而 await 会等待这个 Promise 完成，并将其 resolve 的结果返回出来。

```
function findData() {
  return new Promise(resolve => {
    setTimeout(() => resolve("long_time_value"), 1000);
  });
}

async function test() {
  const v = await findData();
  console.log(v);
}

test();
```