app.use([path, ]hook)

app.get(path[, hook1, hook2, hook3, ...], handler)

任何一个钩子，只要调用了 res.end()（或者 res.send()、res.render() 之类最终会调用 res.end() 的方法）后，就将内容返回给用户，中断后续的钩子；如果调用 next() 则把请求传递给下一个钩子，一直传递到最后的 handler。

app.use(**function** (req, res, next) {

// I'm hook 1, I won't do anything.

next()

})

app.use(**function** (req, res, next) {

// I'm hook 2, I'll break the chain

res.send('Oops! Broke at hook 2')

})

app.get('/', **function** (req, res) {

// Handler for path /

// We responsed the user at hook 2,

// so requests would never reach this handler

res.send('You\'ll never see this')

})

一个请求过来, 被 .use 写好的各个函数依次处理,  
更准确说, 是从第一个开始处理, 到一个做一点操作, 也可能截断流程直接返回..  
而 app.get 等等 HTTP 请求的操作, 统一在 app.routes 着一个模块里边  
<http://expressjs.com/api.html#app.routes>  
所以, 先调用应该是 app.use, 而 app.get 应该是属于被前者调用才对..

我的水平只能解释到这里了..

app.use 加载用于处理http請求的middleware（中间件），当一个请求来的时候，会依次被这些 middlewares处理。



