



《手写OS操作系统》小班二期招生，全程直播授课，大牛带你掌握硬核技术！

点此

慕课网首页

免费课

实战课

体系课

慕课教程

专栏

手记

企业服务



我的

从所有教程的词条中查询...

首页 &gt; 慕课教程 &gt; Go工程师体系课全新版 &gt; 6. 中间件和next函数

全部开发者教程



8. 品牌分类接口

第12周 商品微服务的gin层和oss图片服务

1. 商品列表接口

2. 新建商品

3. 删除，更新商品

4. 商品分类接口

5. 轮播图接口

6. 品牌分类

7. oss快速入门

8. 阿里云oss开发入门

9. 服务端签名直传

10. 内网穿透

11. 项目文档和资源



bobby · 更新于 2022-11-08

◀ 上一节 5. 表单验证 7. 设置静态文件... 下一节 ▶

## 1. 无中间件启动

&lt;&gt; 代码块

预览 复制

```
1  #使用
2  r := gin.New()
3  #替代
4  // 默认启动方式，包含 Logger、Recovery 中间件
5  r := gin.Default()
```

## 2. 使用中间件

&lt;&gt; 代码块

```
1  func main() {
2      // 创建一个不包含中间件的路由器
3      r := gin.New()
4
5      // 全局中间件
6      // 使用 Logger 中间件
7      r.Use(gin.Logger())
8
9      // 使用 Recovery 中间件
10     r.Use(gin.Recovery())
11
12     // 路由添加中间件，可以添加任意多个
13     r.GET("/benchmark", MyBenchLogger(), benchEndpoint)
14
15     // 路由组中添加中间件
16     // authorized := r.Group("/", AuthRequired())
17     // exactly the same as:
18     authorized := r.Group("/")
19     // per group middleware! in this case we use the custom created
20     // AuthRequired() middleware just in the "authorized" group.
21     authorized.Use(AuthRequired())
22     {
23         authorized.POST("/login", loginEndpoint)
24         authorized.POST("/submit", submitEndpoint)
25         authorized.POST("/read", readEndpoint)
26
27         // nested group
28         testing := authorized.Group("testing")
29         testing.GET("/analytics", analyticsEndpoint)
30     }
31
32     // listen and serve on 0.0.0.0:8080
```

意见反馈

收藏教程

标记书签



```
34     r.Run(":8080")
    }
```

### 3. 自定义组件

<> 代码块

```
1  func Logger() gin.HandlerFunc {
2      return func(c *gin.Context) {
3          t := time.Now()
4
5          // Set example variable
6          c.Set("example", "12345")
7
8          // before request
9
10         c.Next()
11
12         // after request
13         latency := time.Since(t)
14         log.Print(latency)
15
16         // access the status we are sending
17         status := c.Writer.Status()
18         log.Println(status)
19     }
20 }
21
22 func main() {
23     r := gin.New()
24     r.Use(Logger())
25
26     r.GET("/test", func(c *gin.Context) {
27         example := c.MustGet("example").(string)
28
29         // it would print: "12345"
30         log.Println(example)
31     })
32
33     // Listen and serve on 0.0.0.0:8080
34     r.Run(":8080")
35 }
```

5. 表单验证 ◀ 上一节      下一节 ▶ 7. 设置静态文件路径和html文件

 我要提出意见反馈