博客系统

功能：主页、注册、登录、发表文章、为文章点赞

1.注册 127.0.0.1:9090/sign post请求

```go
func sign(c *gin.Context) {
    var u user
    u.Username = c.Query("username")
    u.Password = c.Query("password")
    if queryRow(u) {
        c.JSON(http.StatusForbidden, gin.H{
            "status": http.StatusForbidden,
            "message": "已存在该用户名！",
        })
    } else {
        insertRow(u)
        c.JSON(http.StatusOK, gin.H{
            "status": http.StatusOK,
            "message": "注册成功！",
        })
    }
}
```

输入username, password(Query) 在user表中查询

若用户名不存在 则注册成功 同时将用户名和密码 写入数据库user表

若用户名存在 提示：用户名已存在

2.登录 127.0.0.1:9090/login post请求

```go
func login(c *gin.Context) {
    var u user
    u.Username = c.Query("username")
    u.Password = c.Query("password")
    if queryRow(u) {
        cookie := &http.Cookie{
            Name: "username",
            Value: u.Username,
            MaxAge: 1000,
            Path: "/",
            HttpOnly: true,
        }
        http.SetCookie(c.Writer, cookie)
        c.JSON(http.StatusOK, gin.H{
            "status": http.StatusOK,
            "message": "登录成功！",
        })
    } else {
        c.JSON(http.StatusForbidden, gin.H{
            "status": http.StatusForbidden,
            "message": "用户名与密码有误！",
        })
    }
}
```

输入username, password(Query) 在user表中查询
若查询失败　提示：用户名或密码错误

| POST | ∨ | 127.0.0.1:9090/login?username=zy&password=12 | | | Send | ∨ |

Params ● | Authorization | Headers (8) | Body | Pre-request Script | Tests | Settings | Cookies

	KEY	VALUE	DESCRIPTION	●●●	Bulk Edit
☑	username	zy			
☑	password	12			
	Key	Value	Description		

Body | Cookies (1) | Headers (3) | Test Results | Status: 403 Forbidden Time: 8 ms Size: 184 B | Save Response ∨

Pretty | Raw | Preview | Visualize | JSON ∨

```
1  {
2      "message": "用户名与密码有误！",
3      "status": 403
4  }
```

若查询成功　提示：登录成功

| POST | ∨ | 127.0.0.1:9090/login?username=zy&password=123 | | | Send | ∨ |

Params ● | Authorization | Headers (8) | Body | Pre-request Script | Tests | Settings | Cookies

	KEY	VALUE	DESCRIPTION	●●●	Bulk Edit
☑	username	zy			
☑	password	123			
	Key	Value	Description		

Body | Cookies (1) | Headers (4) | Test Results | Status: 200 OK Time: 21 ms Size: 222 B | Save Response ∨

Pretty | Raw | Preview | Visualize | JSON ∨

```
1  {
2      "message": "登录成功！",
3      "status": 200
4  }
```
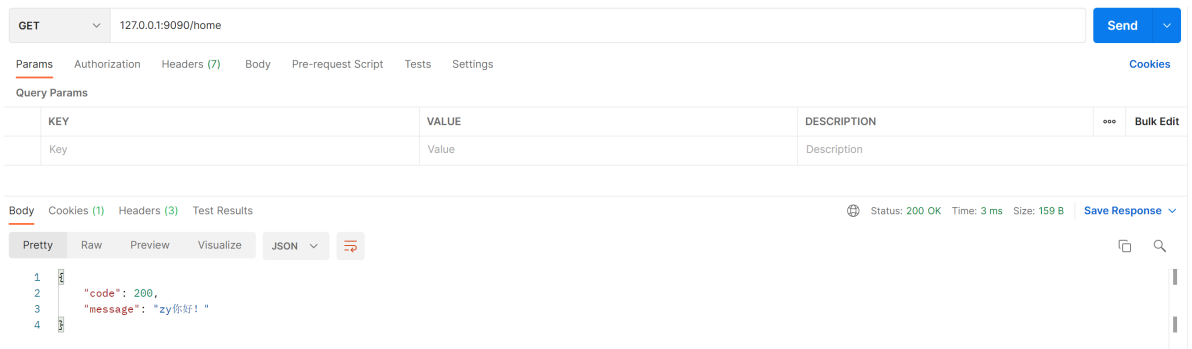
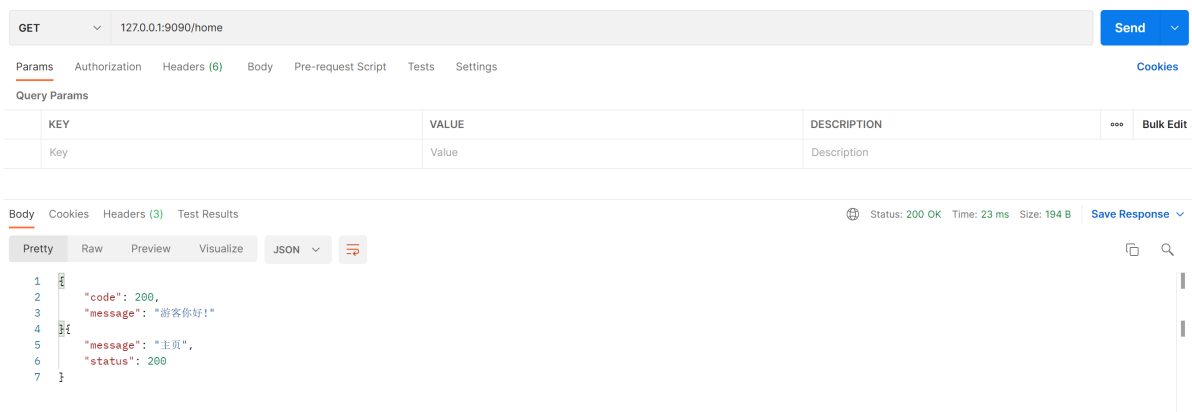## 3.主页 127.0.0.1:9090/home get请求

```go
func mid() gin.HandlerFunc {
    return func(c *gin.Context) {
        cookie, err := c.Request.Cookie("username")
        if err != nil {
            c.JSON(http.StatusOK, gin.H{
                "code": 200,
                "message": "游客你好!",
            })
            c.Next()
            return
        }
        c.JSON(http.StatusOK, gin.H{
            "code": 200,
            "message": cookie.Value + "你好! ",
        })
        c.Abort()
    }
}
```

读取cookie数据

若有   提示: username你好!

GET ∨	127.0.0.1:9090/home			Send ∨
Params	Authorization	Headers (7)	Body	Pre-request Script	Tests	Settings		Cookies

Query Params

KEY	VALUE	DESCRIPTION	∘∘∘	Bulk Edit
Key	Value	Description		

Body Cookies (1) Headers (3) Test Results   Status: 200 OK Time: 3 ms Size: 159 B   Save Response ∨

Pretty Raw Preview Visualize JSON ∨

```
1  {
2      "code": 200,
3      "message": "zy你好! "
4  }
```

若无   提示: 游客你好!

GET ∨	127.0.0.1:9090/home			Send ∨
Params	Authorization	Headers (6)	Body	Pre-request Script	Tests	Settings		Cookies

Query Params

KEY	VALUE	DESCRIPTION	∘∘∘	Bulk Edit
Key	Value	Description		

Body Cookies Headers (3) Test Results   Status: 200 OK Time: 23 ms Size: 194 B   Save Response ∨

Pretty Raw Preview Visualize JSON ∨

```
1  {
2      "code": 200,
3      "message": "游客你好!"
4  }{
5      "message": "主页",
6      "status": 200
7  }
```

4.发表文章 127.0.0.1:9090/post post请求

```go
func postBlog(c *gin.Context) {
    username, err := c.Cookie("username")
    if err != nil{
        c.JSON(http.StatusForbidden, gin.H{
            "status": http.StatusForbidden,
            "message": "请先登录! ",
        })
        return
    }
    title := c.Query("title")
```

```go
        content := c.Query("content")
        if content == "" || title == ""{
            c.JSON(http.StatusForbidden, gin.H{
                "status": http.StatusForbidden,
                "message": "发表失败",
                "reason": "title or content 为空",
            })
            return
        }
        sqlStr := "insert into blog(username, title, content) values (?,?,?)"
        _ , err = db.Exec(sqlStr, username, title, content)
        if err != nil{
            c.JSON(http.StatusBadRequest, gin.H{
                "status": http.StatusBadRequest,
                "message": "发表失败！ ",
                "err": err,
            })
            return
        }
        c.JSON(http.StatusOK, gin.H{
            "status": http.StatusOK,
            "message": "发表成功！ ",
            "username": username,
            "title": title,
            "content": content,
        })
    }
```

读取cookie数据

若无　　　　提示：请先登录！ 并返回

输入title, content(Query)

若title, content为空　　提示：title or content 为空 并返回

title, content都不为空,则将cookie中读取的value, title, content写入数据库blog表中

```
POST    ∨    127.0.0.1:9090/post?title=hello go&content=good golang!                    Send ∨
```

Params ●   Authorization   Headers (8)   Body   Pre-request Script   Tests   Settings   Cookies

	KEY	VALUE	DESCRIPTION		Bulk Edit
☑	title	hello go			
☑	content	good golang!			
	Key	Value	Description		

Body   Cookies (1)   Headers (3)   Test Results   ⊕ Status: 200 OK   Time: 16 ms   Size: 226 B   Save Response ∨

Pretty   Raw   Preview   Visualize   JSON ∨

```
1
2      "content": "good golang!",
3      "message": "发表成功！",
4      "status": 200,
5      "title": "hello go",
6      "username": "zy"
7
```

5.为文章点赞 127.0.0.1:9090/favor put请求

```go
func likeOther(c *gin.Context) {
    u, erro := c.Cookie("username")
    if erro != nil {
        c.JSON(http.StatusForbidden, gin.H{
            "status": http.StatusForbidden,
            "message": "请先登录！",
        })
        return
    }
    username := c.Query("username")
    title := c.Query("title")

    if u == username{
        c.JSON(http.StatusOK, gin.H{
            "status": http.StatusOK,
            "message": "你不能给自己点赞！",
        })
        return
    }

    sqlStr := "update blog set favor=favor+1 where username=? and title = ?"
    ret, err := db.Exec(sqlStr, username, title)
    if err != nil {
        c.JSON(http.StatusInternalServerError, gin.H{
            "status": http.StatusInternalServerError,
            "message": "点赞失败！",
            "err": err,
        })
        return
    }
    n, err1 := ret.RowsAffected()
    if err1 != nil {
        c.JSON(http.StatusOK, gin.H{
            "status": http.StatusOK,
            "message": "点赞失败！",
        })
        return
    }
    if n == 0 {
        c.JSON(http.StatusForbidden, gin.H{
            "status": http.StatusForbidden,
            "message": "点赞失败！",
```
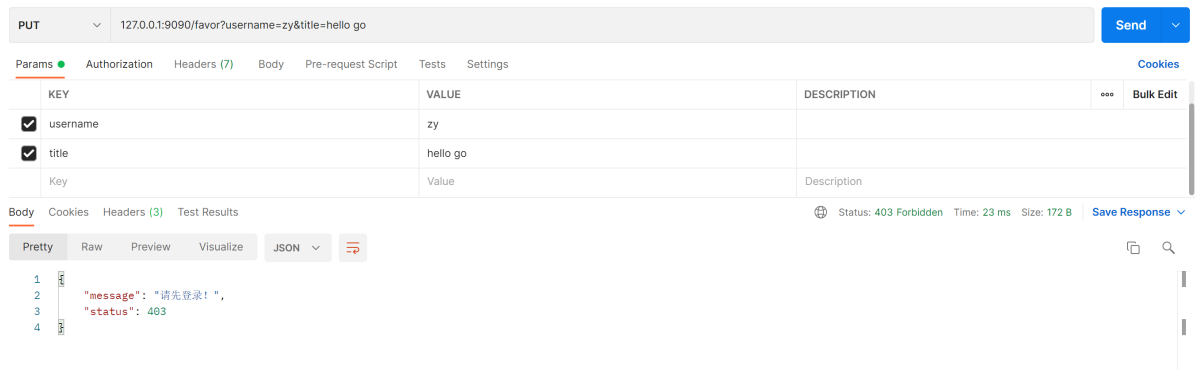
```
            "reason": "文章不存在!",
        })
        return
    }
    c.JSON(http.StatusOK, gin.H{
        "message": "给" + username + "点赞成功!",
        "title": title,
    })
}
```
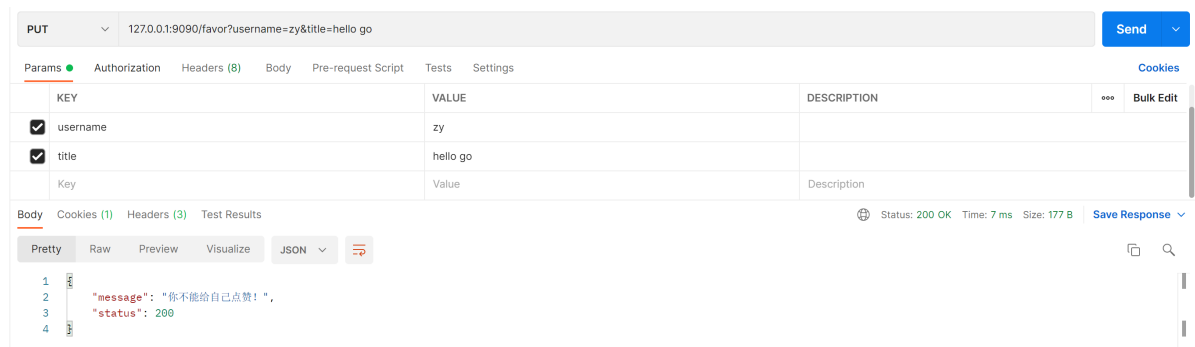
读取cookie数据

若无　　　　提示：请先登录！


```
{
    "message": "请先登录! ",
    "status": 403
}
```

输入username, title(Query)

若username与cookie中value相同　提示：你不能给自己点赞！　并返回


```
{
    "message": "你不能给自己点赞! ",
    "status": 200
}
```

在数据库blog表中查询username, title

若查询失败　　　　提示：点赞失败！文章不存在！　并返回


```
{
    "message": "点赞失败! ",
    "reason": "文章不存在!",
    "status": 403
}
```

若查询成功 将blog表中对应的favor++　　提示：给username点赞成功

PUT | 127.0.0.1:9090/favor?username=zyy&title=hello | Send

Params ● | Authorization | Headers (8) | Body | Pre-request Script | Tests | Settings | Cookies

	KEY	VALUE	DESCRIPTION	●●● Bulk Edit
☑	username	zyy		✕
☑	title	hello		
	Key	Value	Description	

Body | Cookies (1) | Headers (3) | Test Results | Status: 200 OK | Time: 19 ms | Size: 174 B | Save Response ∨

Pretty | Raw | Preview | Visualize | JSON ∨

```
1  {
2      "message": "给zyy点赞成功！",
3      "title": "hello"
4  }
```

数据库表

user表

名	类型	长度	小数点	不是 null	虚拟	键	注释
username	varchar	20	0	☑	☐	🔑1	
password	varchar	20	0	☑	☐		

blog表

名	类型	长度	小数点	不是 null	虚拟	键	注释
username	varchar	20	0	☑	☐		
title	varchar	200	0	☑	☐		
content	varchar	10000	0	☑	☐		
favor	int	11	0	☐	☐		