

-JSON/XML

```
func (r *Response) WriteJson(content interface{}) error
func (r *Response) WriteJsonExit(content interface{}) error
func (r *Response) WriteJsonP(content interface{}) error
func (r *Response) WriteJsonPExit(content interface{}) error
func (r *Response) WriteXml(content interface{}, rootTag ...string) error
func (r *Response) WriteXmlExit(content interface{}, rootTag ...string) error
```

Content Menu

- [JSON](#)
- [JSONP](#)
- [XML](#)

ResponseJSON/XML

1. WriteJson* JSONstringmapstructContent-Typeapplication/json
2. WriteXml* XMLstringmapstructContent-Typeapplication/xml



JSONJSONP

JSON

```
package main

import (
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/net/ghttp"
)

func main() {
    s := g.Server()
    s.Group("/", func(group *ghttp.RouterGroup) {
        group.ALL("/json", func(r *ghttp.Request) {
            r.Response.WriteJson(g.Map{
                "id": 1,
                "name": "john",
            })
        })
    })
    s.SetPort(8199)
    s.Run()
}
```

curl

```
$ curl -i http://127.0.0.1:8199/json
HTTP/1.1 200 OK
Content-Type: application/json
Server: GF HTTP Server
Date: Sun, 05 Jan 2020 02:49:31 GMT
Content-Length: 22

{"id":1,"name":"john"}
```

JSONP

JSONPQuerycallback

```

package main

import (
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/net/ghttp"
)

func main() {
    s := g.Server()
    s.Group("/", func(group *ghttp.RouterGroup) {
        group.ALL("/jsonp", func(r *ghttp.Request) {
            r.Response.WriteJsonP(g.Map{
                "id": 1,
                "name": "john",
            })
        })
    })
    s.SetPort(8199)
    s.Run()
}

```

curl

```

$ curl -i "http://127.0.0.1:8199/jsonp?callback=MyCallback"
HTTP/1.1 200 OK
Server: GF HTTP Server
Date: Sun, 05 Jan 2020 02:50:42 GMT
Content-Length: 34
Content-Type: text/plain; charset=utf-8

MyCallback({"id":1,"name":"john"})

```

XML

```

package main

import (
    "github.com/gogf/gf/v2/frame/g"
    "github.com/gogf/gf/v2/net/ghttp"
)

func main() {
    s := g.Server()
    s.Group("/", func(group *ghttp.RouterGroup) {
        group.ALL("/xml", func(r *ghttp.Request) {
            r.Response.Write(`<?xml version="1.0" encoding="UTF-8"?>`)
            r.Response.WriteXml(g.Map{
                "id": 1,
                "name": "john",
            })
        })
    })
    s.SetPort(8199)
    s.Run()
}

```

curl

```
$ curl -i http://127.0.0.1:8199/xml
HTTP/1.1 200 OK
Content-Type: application/xml
Server: GF HTTP Server
Date: Sun, 05 Jan 2020 03:00:55 GMT
Content-Length: 76
```

```
<?xml version="1.0" encoding="UTF-8"?><doc><id>1</id><name>john</name><
/doc>
```