

HTTPClient-

HTTPHTTP MethodCloseHTTP

GET

```
if r, err := g.Client().Get(ctx, "https://goframe.org"); err != nil {
    panic(err)
}
defer r.Close()
fmt.Println(r.ReadAllString())
```

GET

```
if r, err := g.Client().Get(ctx, "https://goframe.org/cover.png"); err != nil {
    panic(err)
}
defer r.Close()
gfile.PutBytes("/Users/john/Temp/cover.png", r.ReadAll())
```

GETHeader

POST

```
if r, err := g.Client().Post(ctx, "http://127.0.0.1:8199/form",
"name=john&age=18"); err != nil {
    panic(err)
}
defer r.Close()
fmt.Println(r.ReadAllString())
```

&gurl.Encode

POSTmap

```
if r, err := g.Client().Post(
    ctx,
    "http://127.0.0.1:8199/form",
    g.Map{
        "submit"    : "1",
        "callback"  : "http://127.0.0.1/callback?url=http://baidu.com",
    }
); err != nil {
    panic(err)
}
defer r.Close()
fmt.Println(r.ReadAllString())
```

&mapstruct

POSTJSON

Content Menu

- - GET
 - GET
 - POST
 - POSTmap
 - POSTJSON
 - DELETE
- *Bytes*Content
 - GET
 - POST
- *Var

```

if r, err := g.Client().Post(
    ctx,
    "http://user.svc/v1/user/create",
    `{"passport":"john","password":"123456","password-confirm":"123456"}`,
); err != nil {
    panic(err)
}
defer r.Close()
fmt.Println(r.ReadAllString())

```

ghhttpJSONPostContentTypeContent-Typeapplication/json

DELETE

```

if r, err := g.Client().Delete(ctx, "http://user.svc/v1/user/delete/1",
"10000"); err != nil {
    panic(err)
}
defer r.Close()
fmt.Println(r.ReadAllString())

```

*Bytes*Content

BytesContent*Bytes[]byte*Contentstring

GET

```

// content[]bytes
content := g.Client().GetBytes(ctx, "https://goframe.org")

```

```

// contentstring
content := g.Client().GetContent(ctx, "https://goframe.org")

```

POST

```

// content[]bytes
content := g.Client().PostBytes(
    ctx,
    "http://user.svc/v1/user/create",
    `{"passport":"john","password":"123456","password-confirm":"123456"}`,
)

```

```

// contentstring
content := g.Client().PostContent(
    ctx,
    "http://user.svc/v1/user/create",
    `{"passport":"john","password":"123456","password-confirm":"123456"}`,
)

```

*Var

VarHTTPg.VarJSON/XMLg.Varg.Var

```
type User struct {  
    Id    int  
    Name string  
}
```

```
// Struct  
var user *User  
g.Client().GetVar(ctx, url).Scan(&user)
```

```
// Struct  
var users []*User  
g.Client().GetVar(ctx, url).Scan(&users)
```