

WebSocket

gfwebsocketechogfwebsocketHTML5<https://github.com/gogf/gf/tree/master/geg/net/ghttp/server/websocket>

HTML5

H5

```
<!DOCTYPE html>
<html>
<head>
  <title>gf websocket echo server</title>
  <link rel="stylesheet" href="//cdn.bootcss.com/bootstrap/3.3.5/css
/bootstrap.min.css">
  <script src="//cdn.bootcss.com/jquery/1.11.3/jquery.min.js"></script>
</head>
<body>
<div class="container">
  <div class="list-group" id="divShow"></div>
  <div>
    <div><input class="form-control" id="txtContent" autofocus rows="6"
placeholder=""></div>
    <div><button class="btn btn-default" id="btnSend" style="margin-top:
15px"> </button></div>
  </div>
</div>
</body>
</html>

<script type="application/javascript">
  //
  function showInfo(content) {
    $("<div class=\"list-group-item list-group-item-info\">" + content +
"</div>").appendTo("#divShow")
  }
  //
  function showWaring(content) {
    $("<div class=\"list-group-item list-group-item-warning\">" + content
+ "</div>").appendTo("#divShow")
  }
  //
  function showSuccess(content) {
    $("<div class=\"list-group-item list-group-item-success\">" + content
+ "</div>").appendTo("#divShow")
  }
  //
  function showError(content) {
    $("<div class=\"list-group-item list-group-item-danger\">" + content
+ "</div>").appendTo("#divShow")
  }
  }

  $(function () {
    var url = "ws://127.0.0.1:8199/ws";
    var ws = new WebSocket(url);
    try {
      // ws
      ws.onopen = function () {
        showInfo("WebSocket Server [" + url + "] ");
      };
      // ws
      ws.onclose = function () {
        if (ws) {
          ws.close();
          ws = null;
        }
        showError("WebSocket Server [" + url + "] ");
      };
      // ws
      ws.onerror = function () {
        if (ws) {
          ws.close();
          ws = null;
        }
        showError("WebSocket Server [" + url + "] ");
      };
      // ws
      ws.onmessage = function (result) {
        showWaring(" > " + result.data);
      };
    }
  });
}
```

Content Menu

- [HTML5](#)
- [WebSocket](#)
- [HTTPSWebSocket](#)
- [Websocket](#)
- [WebSocket](#)

```

    } catch (e) {
        alert(e.message);
    }

    //
    $("#btnSend").on("click", function () {
        if (ws == null) {
            showError("WebSocket Server [" + url + "] F5!");
            return;
        }
        var content = $.trim($("#txtContent").val()).replace("/[\n]/g",
""");
        if (content.length <= 0) {
            alert("!");
            return;
        }
        $("#txtContent").val("")
        showSuccess(content);
        ws.send(content);
    });

    //
    $("#txtContent").on("keydown", function (event) {
        if (event.keyCode == 13) {
            $("#btnSend").trigger("click");
        }
    });
})

</script>

ws://127.0.0.1:8199/ws

```

- websocket
- websocket
- websocket

WebSocket

package main

```

import (
    "github.com/gogf/gf/frame/g"
    "github.com/gogf/gf/net/ghttp"
    "github.com/gogf/gf/os/gfile"
    "github.com/gogf/gf/os/glog"
)

func main() {
    s := g.Server()
    s.BindHandler("/ws", func(r *ghttp.Request) {
        ws, err := r.WebSocket()
        if err != nil {
            glog.Error(err)
            r.Exit()
        }
        for {
            msgType, msg, err := ws.ReadMessage()
            if err != nil {
                return
            }
            if err = ws.WriteMessage(msgType, msg); err != nil {
                return
            }
        }
    })
    s.SetServerRoot(gfile.MainPkgPath())
    s.SetPort(8199)
    s.Run()
}

3

```

1. **WebSocket** websockethttpghttp.Request.WebSocketr.WebSocket()websocketWebSock
etwebsocketwebsocketerror
2. **ReadMessage & WriteMessage** websocket(ReadMessage & WriteMessage)msgTypemsgT
ypemsgType

HTTPSWebSocket

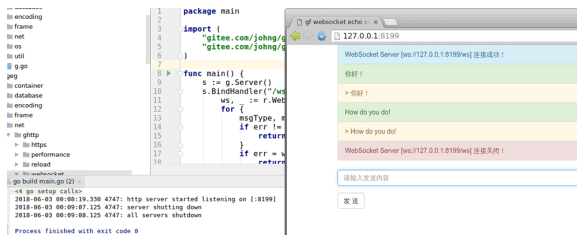
HTTPSWebSocketWebServerHTTPSWebSocket wss:// HTML5WebSocketwss://127.0.0.1:8199/wss

```
package main
```

```
import (
    "github.com/gogf/gf/frame/g"
    "github.com/gogf/gf/net/ghttp"
    "github.com/gogf/gf/os/gfile"
    "github.com/gogf/gf/os/glog"
)

func main() {
    s := g.Server()
    s.BindHandler("/wss", func(r *ghttp.Request) {
        ws, err := r.WebSocket()
        if err != nil {
            glog.Error(err)
            r.Exit()
        }
        for {
            msgType, msg, err := ws.ReadMessage()
            if err != nil {
                return
            }
            if err = ws.WriteMessage(msgType, msg); err != nil {
                return
            }
        }
    })
    s.SetServerRoot(gfile.MainPkgPath())
    s.EnableHTTPS("../https/server.crt", "../https/server.key")
    s.SetPort(8199)
    s.Run()
}
```

main.go <http://127.0.0.1:8199/websocket>



Websocket

gfwebsocket(origin)websocket

1. origin: r.WebSocket()origin(r.Exit() 1. websocket:

WebSocket

gfWebSocket<https://github.com/gogf/gf-demos>