# Websocket

## Server

package main  
  
import (  
 "fmt"  
 "log"  
 "net/http"  
  
 "github.com/gorilla/websocket"  
)  
  
var upgrader = websocket.Upgrader{  
 ReadBufferSize: 1024,  
 WriteBufferSize: 1024,  
}  
  
func handleWebSocket(w http.ResponseWriter, r \*http.Request) {  
 *// 升级HTTP连接为WebSocket协议* conn, err := upgrader.Upgrade(w, r, nil)  
 if err != nil {  
 log.Println("WebSocket upgrade failed:", err)  
 return  
 }  
 defer conn.Close()  
  
 *// 不断读取来自客户端的消息并回复* for {  
 *// 读取消息* \_, msg, err := conn.ReadMessage()  
 if err != nil {  
 log.Println("Error reading message:", err)  
 break  
 }  
 fmt.Printf("Received message: %s\n", msg)  
  
 *// 回复消息* err = conn.WriteMessage(websocket.TextMessage, []byte("Server received: "+string(msg)))  
 if err != nil {  
 log.Println("Error writing message:", err)  
 break  
 }  
 }  
}  
  
func main() {  
 http.HandleFunc("/websocket", handleWebSocket)  
  
 *// 启动Web服务器并监听指定端口* err := http.ListenAndServe(":8080", nil)  
 if err != nil {  
 log.Fatal("Server start failed:", err)  
 }  
}

## Client

package main  
  
import (  
 "fmt"  
 "log"  
 "time"  
  
 "github.com/gorilla/websocket"  
)  
  
func main() {  
 *// 建立WebSocket连接* conn, \_, err := websocket.DefaultDialer.Dial("ws://localhost:8080/websocket", nil)  
 if err != nil {  
 log.Fatal("WebSocket connection failed:", err)  
 }  
 defer conn.Close()  
  
 *// 启动协程来接收消息* go receiveMessages(conn)  
  
 *// 发送消息给服务器* go func() {  
 var i int32  
 for {  
 time.Sleep(3 \* time.Second)  
 i++  
 sendMessage(conn, fmt.Sprintf("%d你好啊 hahah", i))  
 }  
 }()  
  
 *// 保持主程序不退出* select {}  
}  
  
func receiveMessages(conn \*websocket.Conn) {  
 for {  
 *// 读取消息* \_, msg, err := conn.ReadMessage()  
 if err != nil {  
 log.Println("Error reading message:", err)  
 break  
 }  
 fmt.Printf("Received message from server: %s\n", msg)  
 }  
}  
  
func sendMessage(conn \*websocket.Conn, msg string) {  
 err := conn.WriteMessage(websocket.TextMessage, []byte(msg))  
 if err != nil {  
 log.Println("Error sending message:", err)  
 }  
 fmt.Printf("Sent message to server: %s\n", msg)  
}