WEBSOCKET

创建服务器

依赖nodejs-websocket

服务器代码

```
let ws=require("nodejs-websocket")

let server=ws.createServer(conn=>{
    //1
    console.log("new connection")
    conn.on("text",str=>{
        console.log("received"+str)
        conn.sendText(str.toUpperCase()+"!!!")
    })
    conn.on("close",(code,reason)=>{
        console.log("connection closed")
    })
    conn.on("error",(err)=>{
        console.log("x")
    })
}).listen(8001)
```

简单的页面

html

```
<h1>test</h1>
<input type="text" id="sendTxt">
<button id="sendBtn">发送</button>
<div id="recv"></div>
```

js

```
<script>
    const socket=new WebSocket("ws://localhost:8001/")
    socket.onopen=()=>{
        console.log("WS OPEN")
        document.getElementById("recv").innerHTML="connected"
    }
    socket.onclose=()=>{
        console.log("WS CLOSE")
    }
    socket.onmessage=(e)=>{
        console.log("onmessage")
        document.getElementById("recv").innerHTML=e.data
    }
    document.getElementById("sendBtn").onclick=()=>{
        console.log("sendBtn")
        let txt=document.getElementById("sendTxt").value
        socket.send(txt)
    }
</script>
```

接口

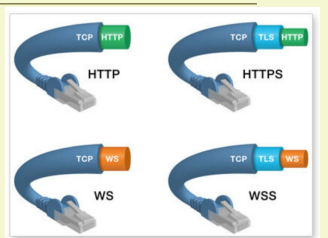
用于连接WebSocket服务器的主要接口，之后可以在这个来凝结上发送和接收数据 — WebSocket

连接关闭时，WebSocket对象发送的事件 — CloseEvent

当从服务器获取到消息时WebSocket对象触发的事件 — MessageEvent

简介

WebSocket最大的特点就是客户端和服务器是平等的，不在是从前只有客户端可以发起通信



一个比较http和ws的图

为什么需要WEBSOCKET

1.传统的http协议是单向，即只能由客户端向服务器发送请求，以达到获取信息的目的

2.想象一个场景，微信的公众号推送，如果使用http协议，那么我们的微信端就必须无时无刻向微信服务器轮询。显然这样做是非常浪费资源的

3.那么，现在就需要一个可以由服务器向客户端发送请求的协议，所以WebSocket诞生了！

4.还有一种典型场景：聊天室

LONG POLLING（轮询）

是另一种流行的通信方法，客户端向服务器请求信息，并在设定的时间段内打开一个连接。