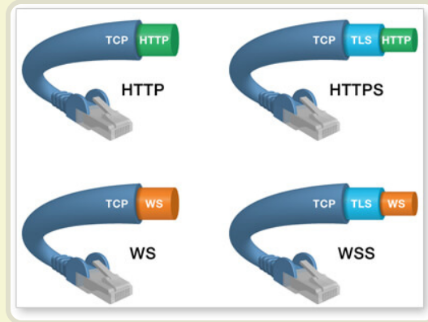
HTTP协议的特点决定，通信永远只能是客户端发起请求，服务端响应。HTTP协议做不到服务端主动向客户端推送消息。

如果服务器有连续的变化，客户端想要获知就非常麻烦。只能轮询：每隔一段时间，就发出一个询问。轮询的效率很低，非常浪费资源，必须不停的连接或者保持连接始终打开。WebSocket就是因此而生。



1. 建立在TCP协议之上
2. 与HTTP协议有良好的兼容性，默认端口号也是80和443，握手阶段也是采用HTTP协议，因此握手阶段不容易被屏蔽，能通过各种HTTP代理服务器。
3. 没有同源限制，客户端可以和任意的服务器通信。
4. 标识符为ws，安全版为wss
5. 构造函数

Var ws=new WebSocket(“ws://localhost:8080”);

新建WebSocket实例。执行上述语句，客户端就会与服务器开始建立连接。

2.webSocket.readyState

readyState有四个值，CONNECTING--0表示正在连接，OPEN--1表示连接已建立

CLOSING—2表示连接正在关闭，CLOSED—3表示连接已关闭活结连接失败

1. ws.onopen=function(){}

指定连接成功之后回调函数

1. ws.onclose  
    指定关闭连接之后的回调函数
2. ws.onmessage  
   指定收到服务器的数据后的回调函数,其事件对象的data表示服务器的数据,event.data
3. ws.onerror  
   用于指定报错时的回调函数
4. we.send()  
   用于向服务器发送信息