HTML5 新特性 WebSocket

1.WebSocket

1.1 什么是WebSocket、Socket.io

WebSocket是一种网络通信协议。 其最大特点是：服务器可以主动向客户端推送消息，当然客户端也可以主动向服务器发送消息，是真正平等的双向对话，属于服务器推送技术的一种。

WebSocket协议是2008年诞生，2011年成为国际标准。目前所有浏览器都支持WebSocket。

Socket.io是一个为浏览器与服务器之间提供实时、双向和基于事件的通信软件库。Socket.io把数据传输抽离成实时通信库，内部对WebSocket进行了封装，抹平了一些细节和平台的兼容性。

socket.io包括：

- Node.js服务器
- 浏览器的JavaScript客户端

1.2WebSocket的优点

- 数据格式比较轻量，通信高效
- 可以发送文本，也可以发送二进制数据
- 没有同源限制，客户端可以与任何服务器通信
- 协议标识符是ws（如果加密，则为wss）

之所以使用websocket是因为HTTP有一个缺点：通信只能由客户端发送

1.3 安装socket.io

1.3.1服务器端

```
npm install --save socket.io
```

1.3.2 JavaScript客户端

- 下载socket.io的JavaScript的客户端脚本文件

  下载地址为：https://cdn.jsdelivr.net/npm/socket.io-client@2/dist/socket.io.js

- 通过<script>标签引入该脚本文件

2.使用socket.io

2.1 服务器端

• 创建 Node HTTP Server

```
const app = require("http").createServer();
const server = require('socket.io')(app);
app.listen(5000);
```

- `connect/connection`事件

`connect/connection`事件在客户端连接到服务器时触发，其语法结构是：

```
server.on('connect',(client)=>{
    ...
});
```

- `emit()`方法

`emit()`方法用于实现服务器向客户端广播事件，其语法结构是：

```
server.emit('事件名称'[,数据])
```

> 事件名称为自定义的事件名称

2.2 客户端

- `JavaScript`客户端

```
<script src="scripts/socket.io.js"></script>
//连接到服务器
<script>
var client = io('ws://127.0.0.1:5000');
</script>
```

> 当通过脚本文件引入`socket.io.js`后将公开名称为`io`函数

- `on()`方法

`on()`方法用于实现根据指定的事件来注册一个函数（侦听服务器的事件，然后执行相关的任务），其语法结构是：

```
client.on('事件名称',callback)
```

> 事件名称是服务器广播的事件名称

- `emit()`方法

`emit()`方法用于实现客户端向服务器端广播事件，其语法结构是：

```
client.emit('事件名称'[,数据])
```

> 事件名称为自定义的事件名称
>
> `encodeURI`,把字符串作为 URI 进行编码
>
> `decodeURI`,对 encodeURI() 函数编码过的 URI 进行解码