**Chapter 10. 채팅 서버 만들기**

* socket.io 사용하기
* 우선 socket.io 모듈과 CORS 모듈을 설치한다.
  + - * CORS가 필요한 이유 : IE등 브라우저는 Ajax를 사용해 데이터를 불러올 수 있도록 XHR(XMLHttpRequest)를 지원한다. 그러나 기본적으로 보안 이슈로 인해 해당 페이지 제공 서버 이외의 다른 서버 접속을 제한한다.
      * CORS 표준을 적용하면, ‘메인 웹 서버가 허용 시 다른 웹 서버에도 접속 가능하다’는 제약 예외가 처리됨.
      * 만일 웹 브라우저가 웹 소켓을 지원하지 않으면 XHR을 사용해야 하며, 이 경우 CORS 미들웨어 적용된 서버가 Access-Control-Allow-Origin 헤더를 포함해 응답을 보낸다. 이 경우에는 다른 웹 서버로도 데이터 요청이 가능해진다.
* [app.js]

socketio모듈, cors모듈 require후 cors를 app.use로 미들웨어 등록한다.

익스프레스 서버 시작 코드를 server 객체로 반환하고, 이를 기반으로 socket.io 서버를 시작한다(listen).

* + - * 클라이언트(socket.io 클라이언트)가 웹 소켓으로 익스프레스 서버에 요청을 보내면, 이를 socket.io 모듈을 기반으로 리턴이 가능해진다! socketio.listen( ) 메소드는 httpServer를 파라미터로 받기 때문에 익스프레스 서버가 실행된 server 변수를 파라미터로 전달한다.

io(소켓 서버 변수)는 sockets라는 객체가 클라이언트 접속/데이터 전송 이벤트를 발생할 수 있다. 이를 통해 connection 이벤트를 우선 처리한다. 요청 클라이언트의 IP, 포트 정보를 우선 설정해준다.

* [chat01.html]

서버 / IP 포트 입력 input box 두 개와 연결하기 버튼을 생성하고, <div> 태그를 추가해 id 속성값으로 result를 넣어둔다. 🡪 이 div 태그 내에 진행과정 로그 출력.

[연결하기] 버튼 클릭 시 socket.io서버에 연결하도록 작성. jQuery 파일 / socket.io 라이브러리 파일 다운로드 진행 후 script src로 연결

그리고

<script> 태그로, 서버 연결 함수를 정의한다.

우선 $( function( ) ) 를 사용해 초기화 함수를 지정한다.

단, 주의사항

script src로 socket.io를 로드할 때 <https://cdn.socket.io/socket.io-1.4.0.js> 를 사용해서 로드한다. 그냥 socket.io.js 로드하면 에러 발생함.

이 html파일 내에서는 클라이언트가 웹 소켓 서버

* 서버에 보낸 메시지 그대로 받기

socket.io 모듈은 메시지를 보내고 받을 때 이벤트 처리 방식을 사용함.  
알다시피 on(event, callback)은 이벤트 수신형으로, 수신 객체가 콜백에 전달되고

emit(event, object)는 이벤트 송신 형태로 여기서는 메시지를 송신함. 사용자 정의 이벤트.

* Echo기능 구현 : 클라에서 서버로 메시지 전송 시, 서버에서 메시지를 그대로 다시 돌려보내는 기능

[chat02.html]

Socket.on에서 connect 이벤트 내부에 socket.on으로 message이벤트 설정한다.

Message 파라미터로 받은 JSON 데이터를 stringfy 해주고, sender, recipient, command, type, data 속성을 설정해 준다.

그리고 클라 화면에서 보내는 사람 아이디, 받는사람 아이디, 메시지를 입력하는 기능을 추가한다.

그 다음 위의 전송버튼 클릭 이벤트를 정의한다. 이 처리는 sendButton에 click 이벤트를 바인딩하고, 서버로 보낼 output에 sender, recipient, data, type등을 stringify시켜준다. 그럼 전송버튼을 클릭하면 message 이벤트를 emit하고 파라미터로 output(JSON포맷 데이터 종류들 구분)output을 전달하면 된다.