|  |
| --- |
|  |
| **채팅 웹 사이트 구현** |
|  |

|  |  |  |
| --- | --- | --- |
|  | 학 번 | 201821898 |
|  | **이 름** | **유 경 석** |

목 차

[1 홈페이지 화면 4](#_Toc90498711)

[2 소스코드 7](#_Toc90498712)

[2.1 회원가입, 로그인 기능 7](#_Toc90498713)

[2.2 채팅 기능 8](#_Toc90498714)

[2.3 소켓(Socket) 실시간 방(room)통신 10](#_Toc90498715)

[2.4 소켓 통신 시 XSS스크립트 방지 11](#_Toc90498716)

[2.5 Jquery 기능 12](#_Toc90498717)

그림 목차

[그림 1‑1 로그인 페이지 4](#_Toc90498719)

[그림 1‑2 회원가입 페이지 4](#_Toc90498720)

[그림 1‑3 연락처 페이지 5](#_Toc90498721)

[그림 1‑4 연락처 추가 페이지 5](#_Toc90498722)

[그림 1‑5 가입되지 않은 사용자와의 채팅 시 5](#_Toc90498723)

[그림 1‑6 채팅 페이지 6](#_Toc90498724)

[그림 2‑1 로그인 기능 코드 7](#_Toc90498725)

[그림 2‑2 회원가입 기능 코드 7](#_Toc90498726)

[그림 2‑3 채팅 코드(메인) 8](#_Toc90498727)

[그림 2‑4 채팅 코드(방 코드 접근) 8](#_Toc90498728)

[그림 2‑5 채팅 코드(연락처를 통한 채팅) 9](#_Toc90498729)

[그림 2‑6 채팅 코드(실시간 통신) 10](#_Toc90498730)

[그림 2‑7 XSS스크립트 방지 코드 11](#_Toc90498731)

[그림 2‑8 처리 결과 11](#_Toc90498732)

[그림 2‑9 스크롤 자동 하강 12](#_Toc90498733)

[그림 2‑10 전달받은 실시간 메시지 출력 12](#_Toc90498734)

# 홈페이지 화면

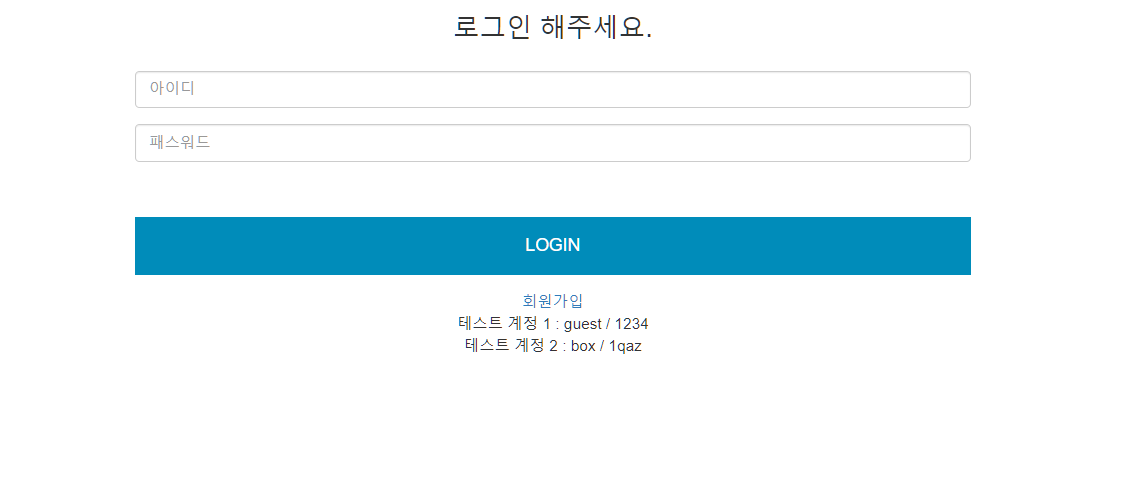


그림 1‑1 로그인 페이지



그림 1‑2 회원가입 페이지

소스코드는 2.1에 있다. 회원가입 시 중복된 아이디나 닉네임을 사용하면 회원가입이 되지 않는다. 또한 특정 폼이 채워지지 않았을 경우에도 회원가입 되지 않는다.

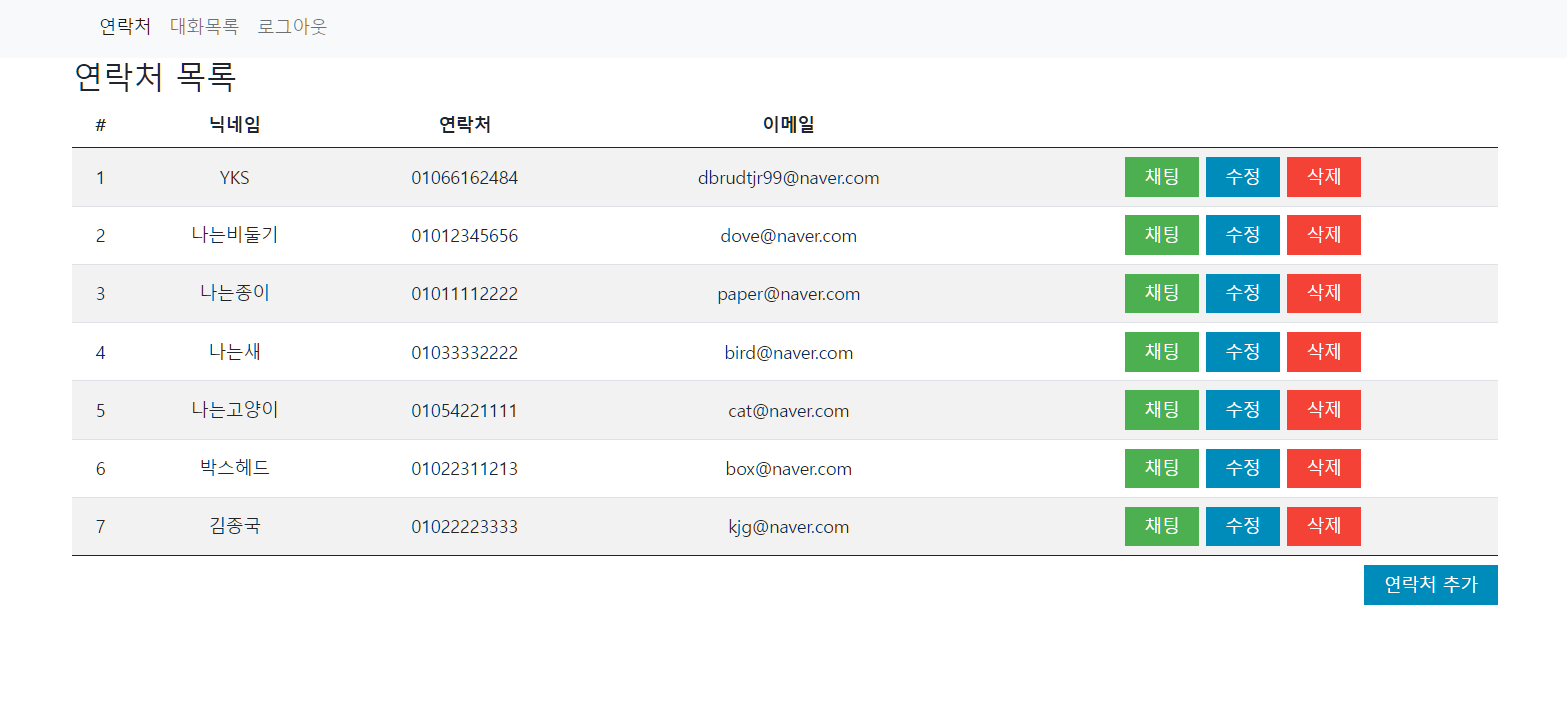


그림 1‑3 연락처 페이지



그림 1‑4 연락처 추가 페이지

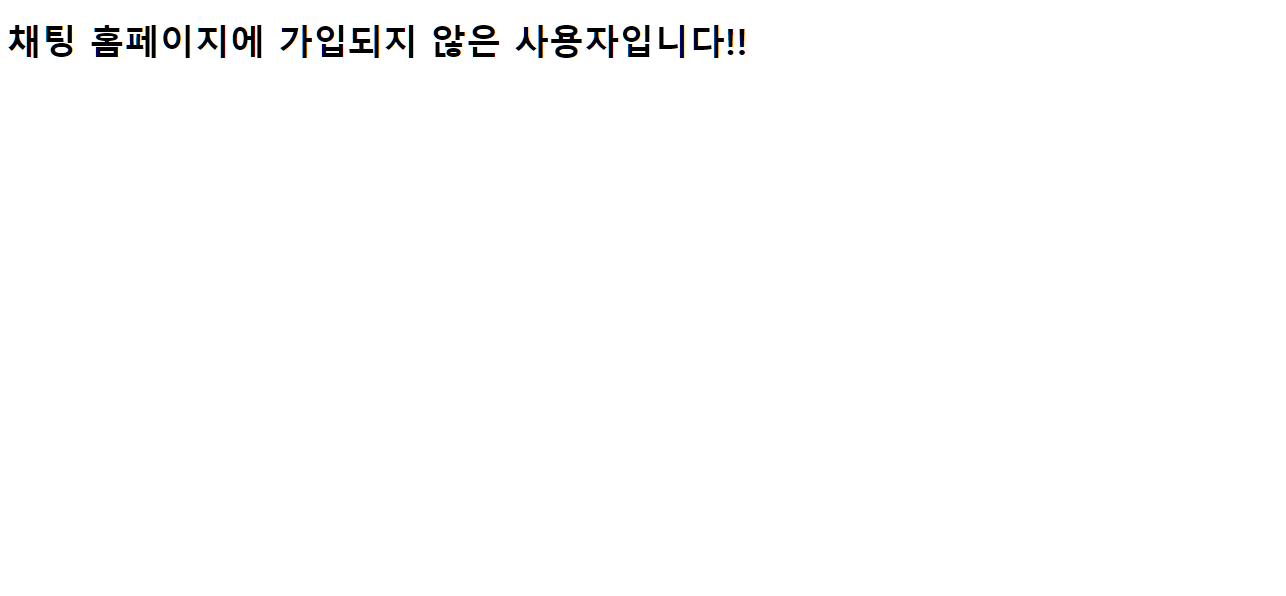


그림 1‑5 가입되지 않은 사용자와의 채팅 시

기본적인 연락처들이다. 계정이 소유한 연락처 목록이 표시되며, 채팅 버튼을 입력하면 채팅을 시작할 수 있다. 연락처 대상이 비회원이면 그림 1‑5이 표시된다.

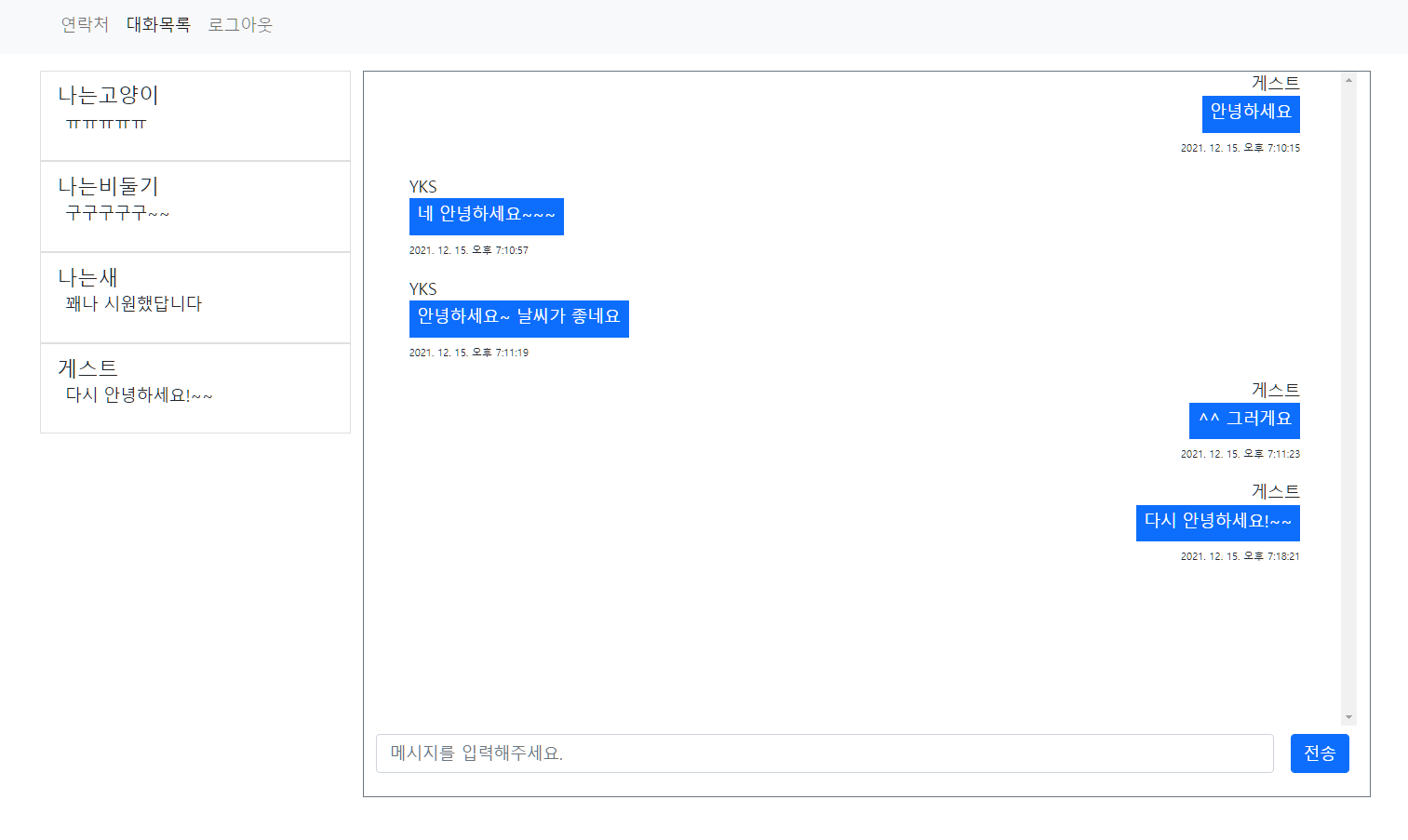


그림 1‑6 채팅 페이지

소스코드는 2.2와 2.3에 있다. 사용자가 가지고 있는 채팅방이 리스트에 표시된다. 리스트를 클릭하면 리스트가 가진 방코드로 연결되며, 채팅이 이어서 시작된다. 악의적인 사용자가 메시지 폼에 XSS스크립트를 시도해볼 수 있기 때문에 HTML엔티티로 필터링 처리했다. 보낼 수 있는 최대 문자 개수는 1000개로 제한하여 1000자가 넘어가면 자동으로 1000자 까지만 잘라서 보내진다.

# 소스코드

## 회원가입, 로그인 기능



그림 2‑1 로그인 기능 코드

기존에 가입된 계정으로 로그인한다. 회원가입 시 사용한 암호방식으로 비밀번호를 일치시킨 후 맞으면 로그인 시킨다. 로그인할 때 계정의 아이디(userid)를 세션에 추가시켜서 여러 코드에서 활용할 수 있도록 만든다.

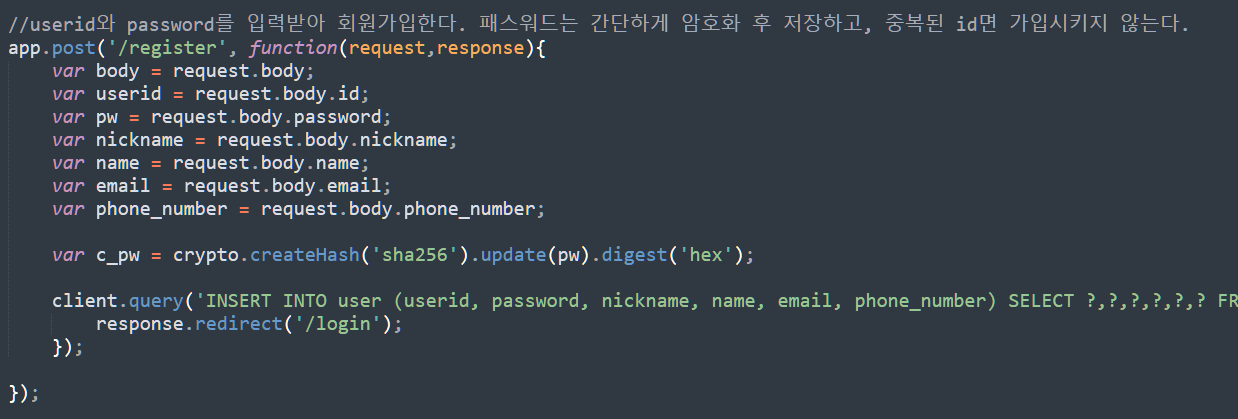


그림 2‑2 회원가입 기능 코드

회원가입에 필요한 인자를 POST로 전달받는다. 이때 비밀번호는 간단하게 암호화하고 연결된 DB에 유저생성 쿼리를 전달한다. 길어서 잘린 쿼리문은 코드 2‑1에 표기돼 있다.

*INSERT INTO user (userid, password, nickname, name, email, phone\_number) SELECT ?,?,?,?,?,? FROM dual WHERE NOT EXISTS (SELECT \* FROM user WHERE userid=? or nickname=?),[userid, c\_pw, nickname, name, email, phone\_number, userid, nickname]*

코드 2‑1 회원가입 쿼리

## 채팅 기능

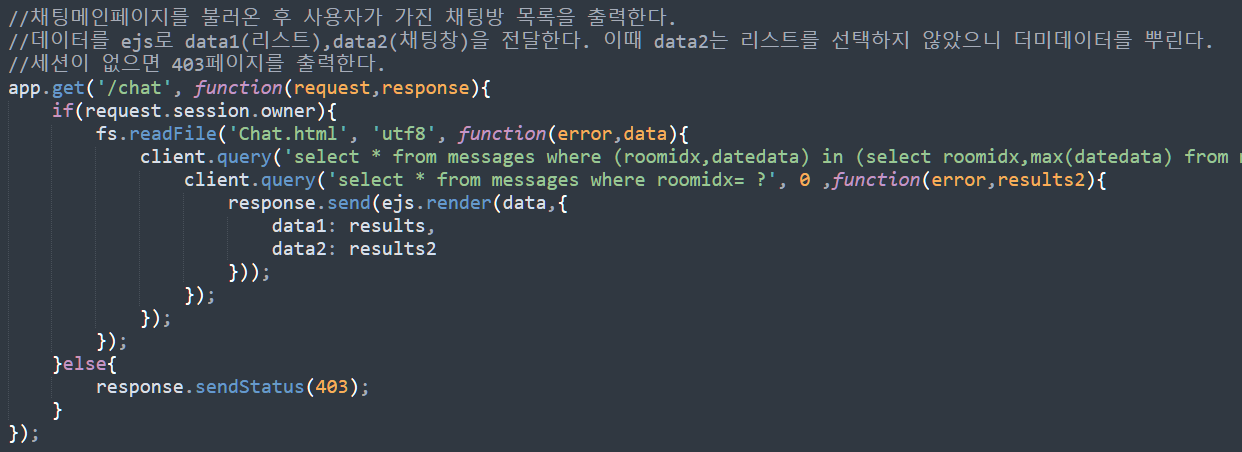


그림 2‑3 채팅 코드(메인)

채팅페이지에 접근하면 사용자가 가진 방 리스트를 질의해서 결과 값을 ejs로 넘겨준다. 이때 아직 리스트를 선택하지 않았기 때문에 채팅 방 데이터인 data2에는 더미데이터를 넘겨준다.

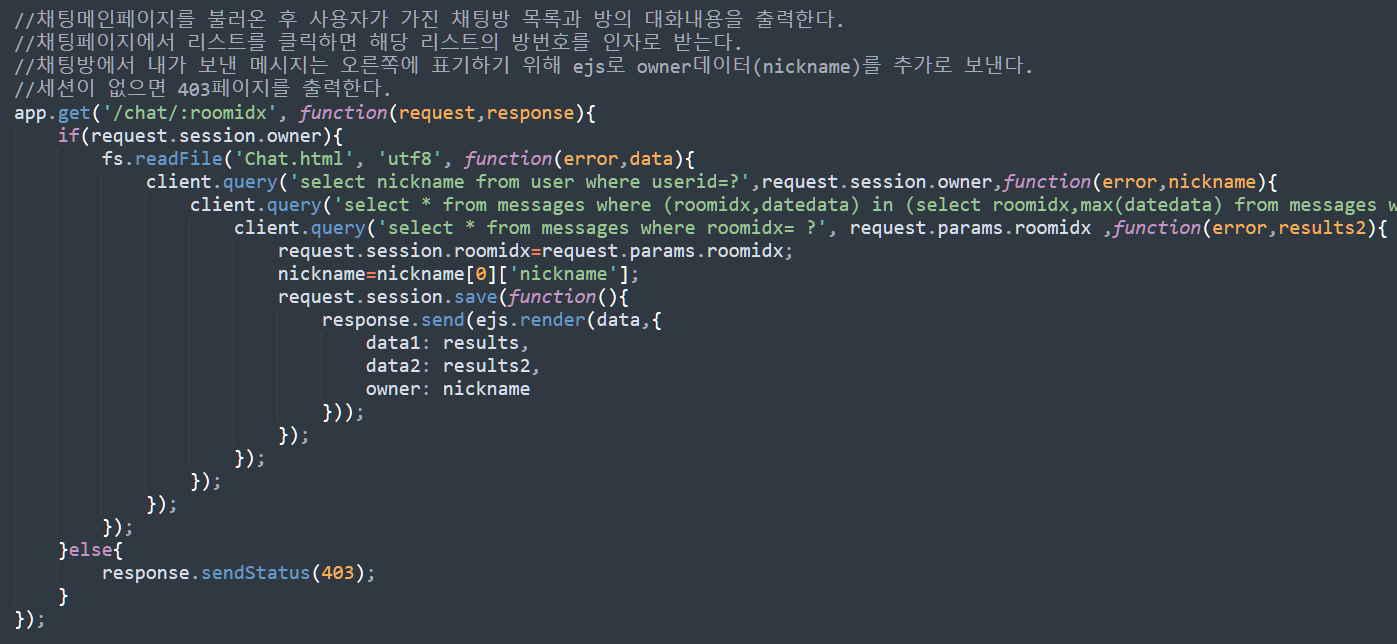


그림 2‑4 채팅 코드(방 코드 접근)

그림 2‑3이 전달한 리스트를 클릭해서 방 코드가 인자로 전달되면 작동하는 코드다. 채팅 방 데이터를 질의 후 결과값을 출력한다. 로그인한 사용자와 채팅하고 있는 상대방을 구분하기 위해서 사용자의 닉네임 값을 함께 전달한다. 길어서 잘린 쿼리문은 코드 2‑2에 표기돼 있다.

*select \* from messages where (roomidx,datedata) in (select roomidx,max(datedata) from messages where(roomidx) in (select roomidx from chatRoomAttendant where userid= ? ) group by roomidx), [request.session.owner]*

코드 2‑2 사용자가 가진 방 목록+목록에 출력 될 최근 메시지 쿼리

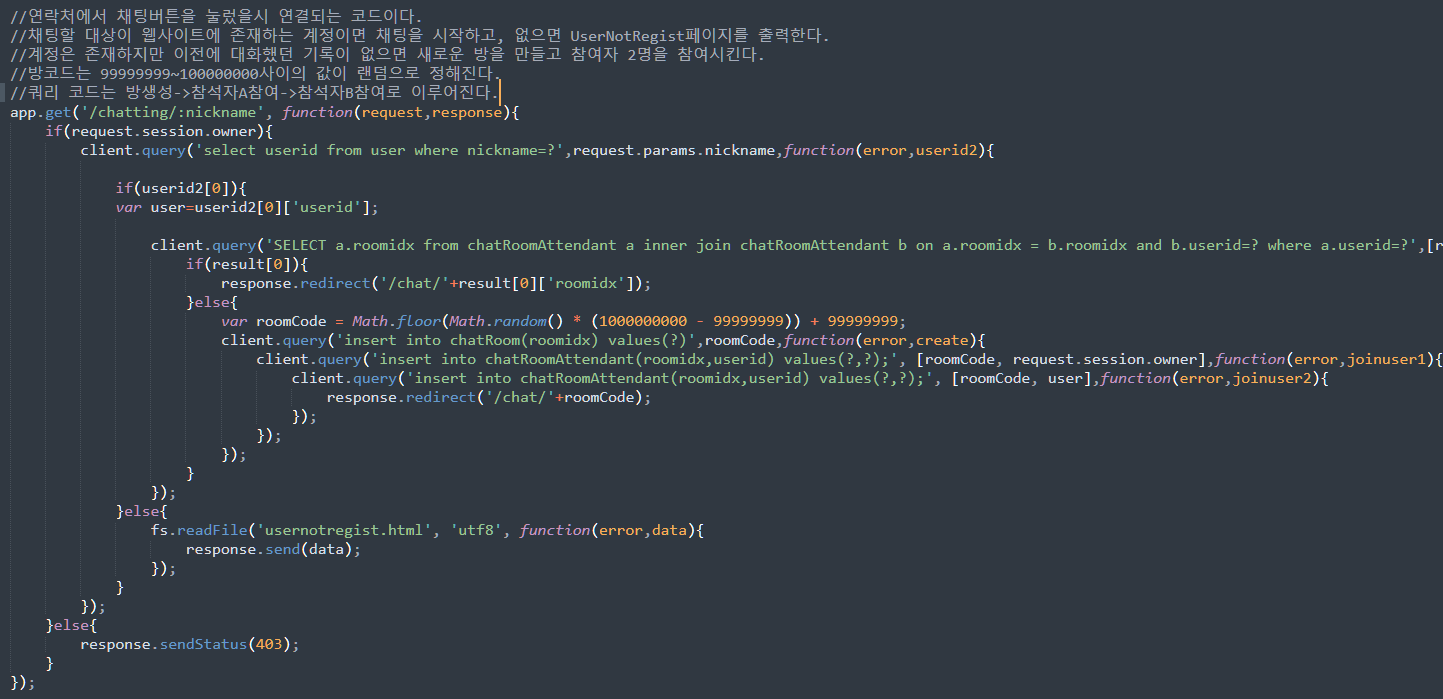


그림 2‑5 채팅 코드(연락처를 통한 채팅)

연락처 페이지에서 채팅 버튼을 누르면 실행되는 코드이다. 채팅을 진행할 상대방이 홈페이지에 가입된 사용자인지 질의한다. 아니라면 오류 페이지를 출력하고 사용자가 존재하면 채팅을 시작할 준비를 한다.

채팅 시작 과정은 4단계이다.

1. 사용자 간의 채팅 방이 생성돼 있으면 그림 2‑4에 인자를 전달한다.
2. 1.이 아니라 새로운 채팅이면 랜덤 코드로 방을 생성한다.
3. 새로 만들어진 방에 참가자 두 명을 포함시킨다.
4. 그림 2‑4에 인자를 전달한다.

인자는 방코드를 전달하게 되며 프로세스가 모두 완료되면 사용자는 채팅을 할 수 있다.

*SELECT a.roomidx from chatRoomAttendant a inner join chatRoomAttendant b on a.roomidx = b.roomidx and b.userid=? where a.userid=?,[request.session.owner, user]*

코드 2‑3 동시 소유중인 방 쿼리

코드 2‑3은 채팅을 신청한 사용자와 채팅 할 사용자가 동시에 소유하고 있는 방 코드를 질의하는 쿼리문이다. 동시에 소유한 방 코드가 없으면 채팅방이 없는 것으로 간주한다.

## 소켓(Socket) 실시간 방(room)통신

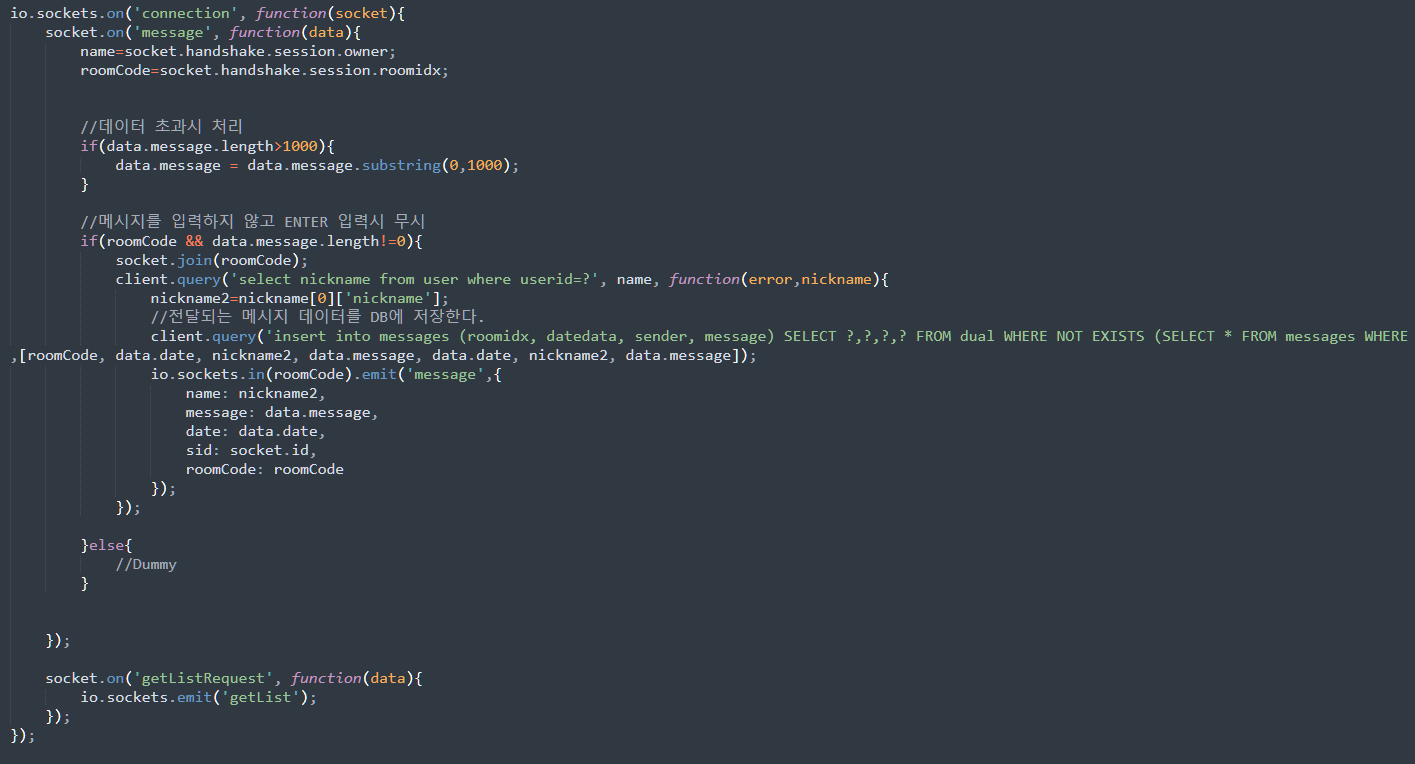


그림 2‑6 채팅 코드(실시간 통신)

실시간 방 단위 통신을 구현한 소켓 코드이다. 실시간 데이터에 포함시키기 위해 세션에서 사용자 이름과 최근에 접속한 방코드를 입력 받는다.

메시지가 1000자가 넘어가면 초과된 메시지는 모두 잘라버린다. 반대로 아무것도 입력하지 않았을 경우 보내지도 않고 저장하지도 않는다.

세션에 저장된 아이디를 DB에 질의하여 닉네임으로 바꾸고 데이터를 보내기 전 먼저 DB에 저장한다. 저장 후에는 지정된 방 번호(소켓번호)로 데이터를 전달하고 방에 ‘message’ 이벤트를 방출한다.

*insert into messages (roomidx, datedata, sender, message) SELECT ?,?,?,? FROM dual WHERE NOT EXISTS (SELECT \* FROM messages WHERE datedata=? and sender=? and message=?)*

*,[roomCode, data.date, nickname2, data.message, data.date, nickname2, data.message]*

코드 2‑4 메시지 저장 쿼리

## 소켓 통신 시 XSS스크립트 방지

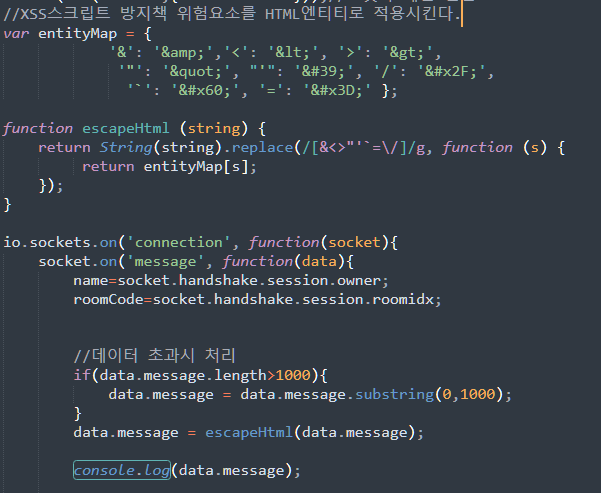


그림 2‑7 XSS스크립트 방지 코드

웹 요소로 사용될 수 있는 특정 데이터를 필터링 해서 HTML엔티티로 변환시키는 코드이다. 데이터를 처리하기 전에 먼저 데이터를 변환시킨다.

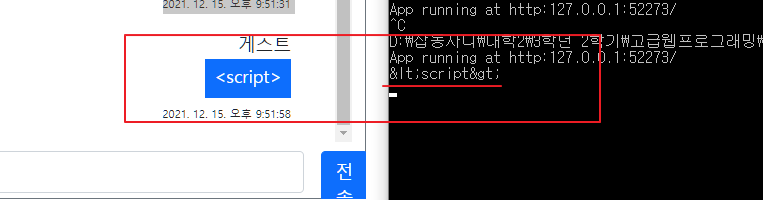


그림 2‑8 처리 결과

변환된 태그의 모습이다. 웹 브라우저 상에서는 변환된 문자 코드가 아닌 정상적인 데이터로 표기된다.

## Jquery 기능

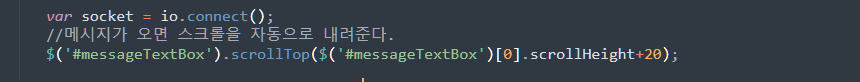


그림 2‑9 스크롤 자동 하강

Jquery에서 메시지가 전달되면 스크롤을 자동으로 내려주는 코드이다.

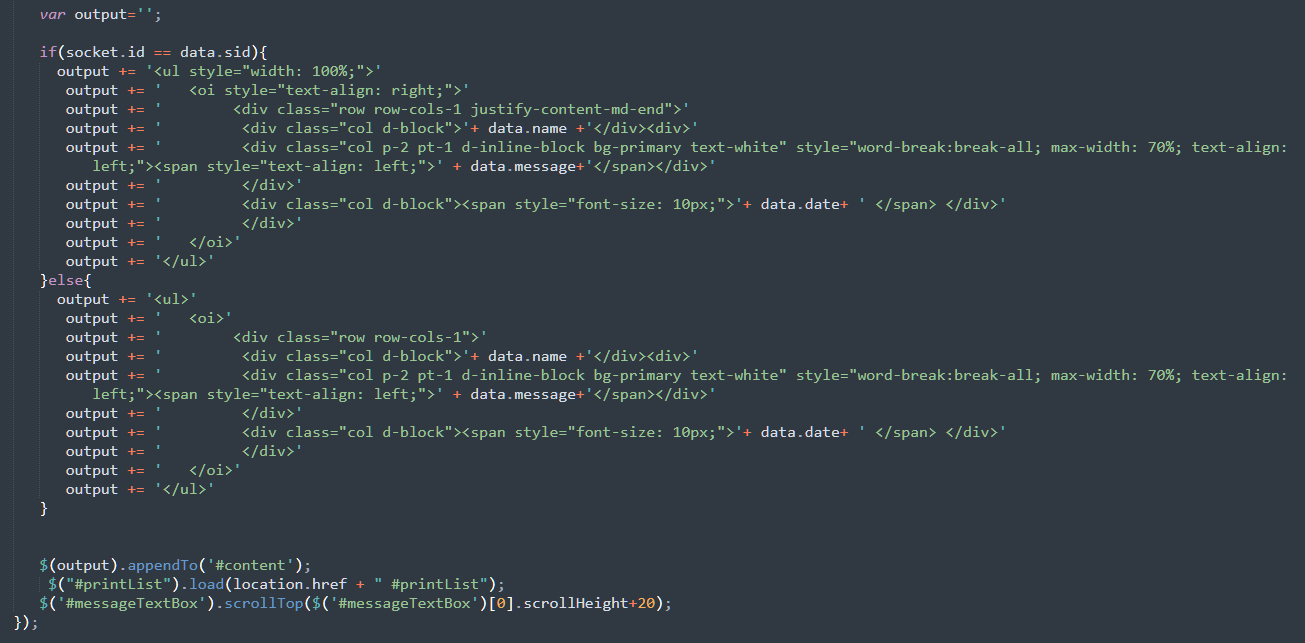


그림 2‑10 전달받은 실시간 메시지 출력

사용자가 보낸 데이터를 폼에 맞춰 정렬하기 위한 코드다. 보낸 사용자가 자신이면 오른쪽, 보낸 사용자가 다른 사람이면 메시지를 왼쪽에 출력한다.