

# 평가지

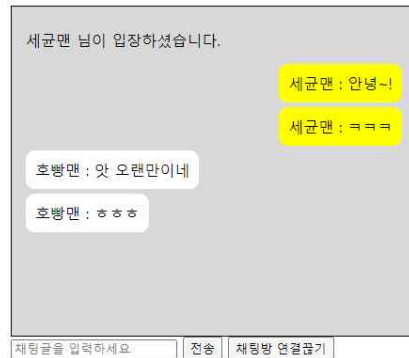
과 정 명	Spring기반자바(Java)융합개발자_스마트훈련과정	과 목 명	네트워크 프로그래밍 구현
능력단위	네트워크 프로그래밍 구현 (2001020508_14v2)	평가방법	평가자 체크리스트
능력단위 요소명	개발환경 분석하기 기능 구현하기	평가자	김상우
		평가일자	2021.07.06
평가문항 (수행내용)	<p>다음과 같은 요구사항에 맞는 실시간 네트워크 프로그램을 구현하시오.</p> <p>◆ 제작내용</p> <ol style="list-style-type: none"> <li>1) 웹소켓 기반의 간단한 실시간 채팅 프로그램을 구현한다.</li> <li>2) Tomcat WAS의 @Endpoint 애노테이션 기반으로 채팅 서버를 구현한다.</li> <li>3) 웹 브라우저에서는 자바스크립트의 WebSocket 기반으로 채팅 클라이언트를 구현한다.</li> </ol> <p>◆ 세부영역별 지시사항</p> <ul style="list-style-type: none"> <li>• 미들웨어로 Tomcat WAS를 사용하고, tomcat-websocket-api 라이브러리를 Tomcat WAS 버전에 일치되도록 사용한다.</li> <li>• 자바 서블릿 라이브러리는 4 버전 이상으로 사용한다.</li> <li>• 채팅 서버 역할의 자바 클래스에서 @OnOpen, @OnMessage, @OnClose, @OnError 이벤트 콜백 메소드를 구현한다.</li> <li>• 채팅 클라이언트 역할의 자바스크립트 코드에서 웹소켓의 onOpen, onMessage, onClose, onError 이벤트 콜백 메소드를 구현한다.</li> <li>• 채팅 서버에서 연결된 클라이언트에게 데이터를 전송하거나 또는 연결을 끊는 서버 기능을 구현한다.</li> <li>• 채팅 클라이언트에서 연결된 서버에게 데이터를 전송하거나 또는 연결을 끊는 클라이언트 기능을 구현한다.</li> </ul> <p>◆ 제출방법</p> <ol style="list-style-type: none"> <li>1) 구현한 네트워크 프로젝트의 폴더를 ZIP 파일로 압축하여 제출함</li> <li>2) 구현한 네트워크 프로젝트를 소스코드 일부와 실행화면 단위로 간단히 설명하는 스토리보드 양식의 PPT 파일을 작성하여 제출함</li> </ol>		

◆ 네트워크 프로그래밍 구현 소스코드 일부 및 실행화면 예시

방 채팅 서비스

방제목 : 어서 들어와요~!

방번호 UUID : 8f26a560-e312-4858-9080-9ceca092786b



《실시간 채팅화면 - 1》

```

@Log
@ServerEndpoint(value = "/simpleChat")
public class SimpleChatEndpoint { // 각 클라이언트와 연결될때마다 해당 Endpoint 클래스 객체가 생성됨
    private static final List<Session> sessions = new ArrayList<>(); // static으로 공유 채팅방 현재만 존재

    @OnOpen
    public void handleOpen(Session session) throws IOException {
        log.info("==== 웹소켓 클라이언트와 연결됨 =====");
        sessions.add(session);
        log.info("현재 접속 : " + session.getId());
    }

    @OnMessage
    public void handleMessage(Session session, String message) throws IOException {
        log.info("==== 웹소켓 클라이언트로부터 데이터를 받음 =====");
        log.info("메시지 전송 = " + session.getId() + " : " + message);

        // sessions 객체 기준으로 동기화(다른 스레드가 sessions 조작 못함)
        synchronized (sessions) {
            // 모든드레스들하기
            for (Session sess : sessions) {
                sess.getBasicRemote().sendText(message);
            } // for
        } // sessions
    }

    @OnClose
    public void handleClose(Session session, CloseReason closeReason) throws IOException {
        log.info("==== 웹소켓 클라이언트와 연결이 해제됨 =====");
        sessions.remove(session);
        log.info("현재 접속 : " + session.getId());
    }

    @OnError
    public void handleError(Session session, Throwable throwable) {
        log.info(throwable.getMessage());
        throwable.printStackTrace();
    }
}

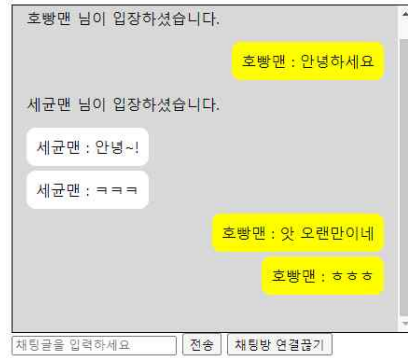
```

《채팅서버 소스코드 일부》

방 채팅 서비스

방제목 : 어서 들어와요~!

방번호 UUID : 8f26a560-e312-4858-9080-9ceca092786b



《실시간 채팅화면 - 2》

```

<script>
var websocket;
var nickname;

document.getElementById('join').addEventListener('click', function () {
    nickname = document.getElementById('nickname').value;
    connect();
    addMinEvt();
});

document.getElementById('send').addEventListener('click', function () {
    send();
});
document.getElementById('message').addEventListener('keyup', function (event) {
    if (event.keyCode === 13) {
        send();
    }
});

document.getElementById('disconnect').addEventListener('click', function () {
    disconnect();
});

function connect() {
    console.log('location.host : ' + location.host); // location.host : localhost:8082
    var url = 'ws://' + location.host + '/websocket' + '/simpleChat';
    console.log('url : ' + url);

    // 웹소켓 서버에 연결하기
    websocket = new WebSocket(url);
    websocket.onopen = onOpen;
    websocket.onmessage = onMessage;
    websocket.onclose = onClose;
    // connect
}

```

《채팅 클라이언트 소스코드 일부》

과제물  
제출  
및 보관

제출물

최종파일 : 프로젝트 ZIP 파일 / PPT 파일

평가자료 보관방법

파일 보관

평가항목

평가구 분	순 번	평가내용	평가 기준	평가결과		
				배점	예	아니오
개발환경 분석하기	1	구현 도구로 Eclipse Java EE 또는 Spring Tools Suite를 설치할 수 있다.	Y/N	5		
	2	미들웨어로 Tomcat WAS를 설치하고 이클립스 서버 개발 환경으로 설정할 수 있다.	Y/N	5		
	3	빌드 도구로 Maven 또는 Gradle을 사용하는 Dynamic Web Project를 생성할 수 있다.	Y/N	5		
	4	형상관리 도구로 Git을 사용하여 GitHub의 원격 저장소와 이클립스의 프로젝트를 연동할 수 있다.	Y/N	5		
	5	서블릿 라이브러리를 4 버전 이상으로 의존성을 설정하고 web.xml에서 web-app 요소의 버전 속성을 설정된 서블릿 버전으로 수정할 수 있다.	Y/N	5		
	6	tomcat-websocket-api 라이브러리를 Tomcat WAS 버전	Y/N	5		

		에 맞추어 의존성을 추가할 수 있다.					
	수 행 준 거	1.1 개발방법 기준에 따라서 네트워크 프로그래밍 구현을 위한 H/W(PC, workstation, server 등) 및 S/W(Unix, windows, IOS 등) 개발환경을 구축할 수 있다. 1.2 네트워크 개발환경의 프로그래밍 구현을 위하여 표준 개발 도구(commands, .net framework control, edipse 등)를 사용할 수 있다. 1.3 개발 방법론에 따라서 네트워크 프로그래밍 구현을 위한 타겟시스템 형상(Configuration)을 수정하여 보완할 수 있다.					
	기능 구현하기	7	특정 URL 주소로 클라이언트 웹소켓의 접속을 대기하는 서버 웹소켓 클래스를 작성할 수 있다.	Y/N	6		
		8	클라이언트와 연결됐을 때 필요한 서버 기능을 구현할 수 있다.	Y/N	6		
		9	클라이언트로부터 메시지를 수신받았을 때 필요한 서버 기능을 구현할 수 있다.	Y/N	6		
		10	클라이언트와 연결이 끊어졌을 때 필요한 서버 기능을 구현할 수 있다.	Y/N	6		
		11	클라이언트와 연결된 상태에서 에러가 발생했을 때 필요한 서버 기능을 구현할 수 있다.	Y/N	5		
		12	연결된 클라이언트에게 데이터를 전송하거나 연결을 끊는 서버 기능을 구현할 수 있다.	Y/N	6		
		13	클라이언트가 웹소켓을 이용하여 웹소켓 서버에 접속할 수 있다.	Y/N	6		
		14	서버와 연결됐을 때 필요한 클라이언트 기능을 구현할 수 있다.	Y/N	6		
		15	서버로부터 메시지를 수신받았을 때 필요한 클라이언트 기능을 구현할 수 있다.	Y/N	6		
		16	서버와 연결이 끊어졌을 때 필요한 클라이언트 기능을 구현할 수 있다.	Y/N	6		
		17	서버와 연결된 상태에서 에러가 발생했을 때 필요한 클라이언트 기능을 구현할 수 있다.	Y/N	5		
		18	연결된 서버에게 데이터를 전송하거나 연결을 끊는 클라이언트 기능을 구현할 수 있다.	Y/N	6		
		수 행 준 거	2.1 개발방법 기준에 따라서 네트워크 프로그래밍 응용프로그램을 구현할 수 있다. 2.2 프로그래밍 방법론에 따라서 설계내용을 바탕으로 네트워크 프로토콜을 구현할 수 있다. 2.3 구축 계획에 따라 자원관리를 위하여 데이터베이스를 구현할 수 있다. 2.4 효율적인 자원관리를 위하여 에이전트(Agent)를 구현할 수 있다. 2.5 효과적인 트래픽 분석을 위하여 네트워크 QoS 제공방안을 구현할 수 있다.				

성취기준	◆ 성취수준환산					
	점수구간	90~100점	80~89점	70~79점	60~69점	60점미만
	성취수준환산	5	4	3	2	1
		매우우수	우수	보통	미흡	매우미흡