

닷넷 프로그래밍
팀 프로젝트 최종 보고서
1조

조	1조		
	학번	학과	이름
팀장	122098	컴퓨터공학과	심우일
팀원	112228	컴퓨터공학과	최원석
	122025	컴퓨터공학과	남민수-
	122170	컴퓨터공학과	이주경
제출일자	2016.12.16		

■ 목차

1. 제목 및 개요

2. 팀 인원 및 역할 분담 내용

- 팀원

- 역할 분담

3. 주요 기능

4. MVC 패턴 분석

5. 기타

- 발표 후 추가사항

- 프로젝트에 대한 팀원 의견

* 기존 제출했던 제안서를 바탕으로 수정 된 사항은 빨간 글씨로 표기하였습니다.

1. 제목 및 개요

· 프로젝트 제목 : 삼목 게임 Tic-Tac-Toe

· 개요 :

저희 조가 개발할 게임은 삼목게임 Tic-Tac-Toe입니다. AI 기능을 가진 컴퓨터나 다른 사람과 함께 게임을 할 수 있습니다. 웹기반으로 데이터베이스 등을 이용하여 삼목게임을 구현합니다. 게임 방식은 다음과 같습니다. 우선 페이지를 시작하면 로그인 화면이 나타나고, 회원가입을 원하면 회원가입 버튼을 사용합니다. 로그인이 성공하면 컴퓨터와 게임을 할지 다른 사람과 같이 할지 선택합니다. 컴퓨터와 게임을 하도록 버튼을 누른 경우에는 자체 개발한 알고리즘으로 구현한 인공지능 컴퓨터와 경기를 합니다. 게임은 계속할 수 있고, 게임이 끝나는 경우에는 게임 기록이 로그인 정보와 함께 데이터베이스에 저장됩니다.

2. 팀 인원 및 역할 분담 내용

- 팀 구성

· 팀장 : 122098 심우일

· 팀원 : 112228 최원석

122025 남민수

122170 이주경

- 역할 분담 내용

최초 제안서에는 아래와 같이 기술 하였습니다.

심우일 : 삼목 게임 개발, 데이터베이스 설계, 제안서 작성, 보고서 작성

최원석 : 로그인 페이지, 회원가입 페이지 개발, 프로그램 테스터,
제안서 작성, 보고서 작성

남민수 : 삼목 게임 개발 및 데이터베이스 연동, 제안서 작성, 보고서 작성

이주경 : 로그인 페이지, 회원가입 페이지 개발, 프로그램 테스터,
제안서 작성, 보고서 작성

하지만 개발 과정에서 팀 구성원들과 회의 한 결과 아래와 같이 수정되었습니다.

- 심우일 : 프로젝트 기획, DB 설계 및 기타 페이지 개발, 보고서 작성
- 남민수 : 메인 게임 개발 및 데이터베이스 연동, 게임 테스터, UI/UX 디자인 설계
- 최원석 : 메인 게임 개발 및 기타 페이지 개발, UI/UX 디자인, 보고서 작성,
- 이주경 : 프로젝트 기획, DB 설계 및 연동, 보고서 작성, 게임 테스터

3. 주요 기능

4. MVC 패턴 분석

- MODEL

저희는 수업시간에 사용하는 데이터베이스 프로그램인 SQL Server 2008 R2 MSE 을 사용하여 데이터베이스를 설계하였습니다.

저의 게임에서 사용될 데이터베이스에는 회원가입을 통한 아이디와 패스워드 정보, 게임이 이루어진 이후에 남는 플레이어 정보와 승자 정보가 저장되게 됩니다.

1) 회원가입 테이블

회원가입 테이블에는 아이디와 비밀번호가 저장되게 됩니다. 기본 키는 아이디입니다. 추가로 Email 정보를 받아 저장합니다.

2) 기록보기 테이블

기록보기 테이블에는 대결한 양 플레이어(혹은 컴퓨터)의 아이디가 저장되고, 둘 중 승자에 대한 정보가 저장됩니다. 기본 키는 플레이어와1과 플레이어2가 됩니다. 이 두 키는 외래 키가 되기도 합니다.

그러나 플레이어1과 플레이어2를 기본 키로 하면 다시 게임을 진행하였을 때 정보가 저장 안 되기 때문에 튜플을 세는 속성을 추가하여 기본 키로 만들었고, 플레이어1을 외래 키로 지정하였습니다.

아래의 표는 테이블 예시입니다.

1) 회원가입 테이블

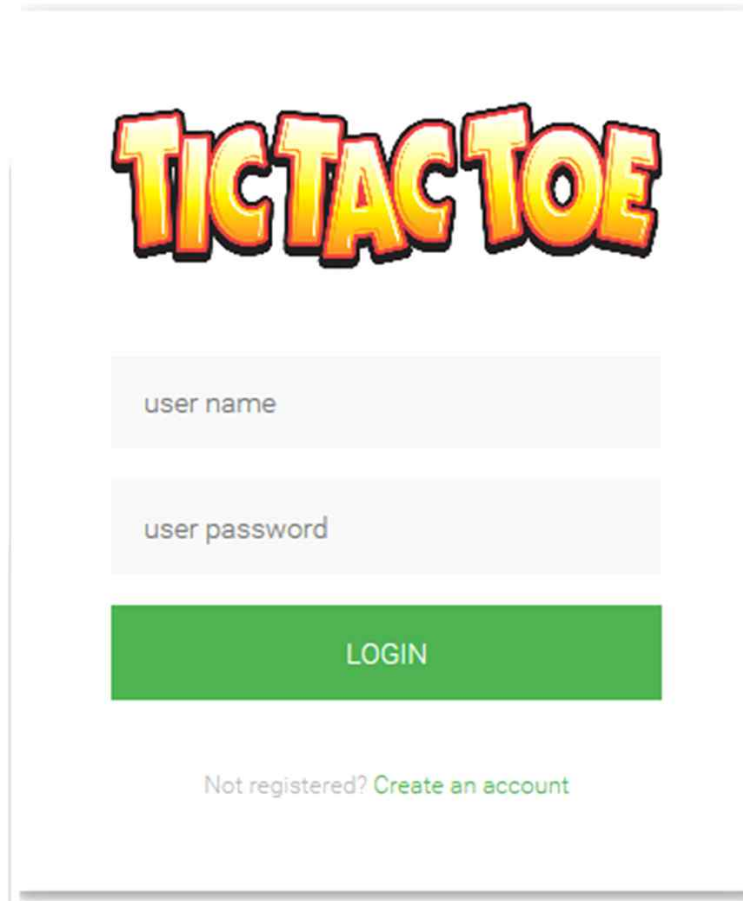
Username	Password	Email
홍길동	123123	aa@naver.com

2) 기록보기 테이블

Number	Player1	Player2	Result
9	홍길동	Guest-aaa	WIN
10	홍길동	Computer	LOSE

- VIEW & CONTROLLER

- LoginPage.aspx



LoginPage.aspx(시작 화면)

○ 프로그램 실행 시 보이는 초기 화면입니다.

- 중간에 텍스트박스 두개로 아이디와 비밀번호를 입력 받습니다.
- 아이디가 없는 유저를 위해서 Create an account 문구를 클릭하면 회원가입을 할 수 있도록 합니다.
- 아이디와 비밀번호를 모두 입력하고 로그인 버튼을 통해 로그인을 합니다.

■ LoginPage.aspx.cs

-컨트롤러

① btnLogin_Click

로그인 버튼 클릭으로 사용자가 입력한 아이디와 비밀번호가 데이터베이스에 저장된 정보와 맞는지 확인 후 맞으면 MainPage.aspx 페이지로 이동 시켜주고 그렇지 않으면 오류 메시지를 띄웁니다.

② btnJoin_Click

회원가입 버튼 클릭으로 사용자가 아이디가 없을 경우 아이디 생성을 위한 버튼이다. 회원가입 버튼을 누르게 되면 JoinPage.aspx 페이지로 이동하게 됩니다.

-함수

① private bool checkLogin()

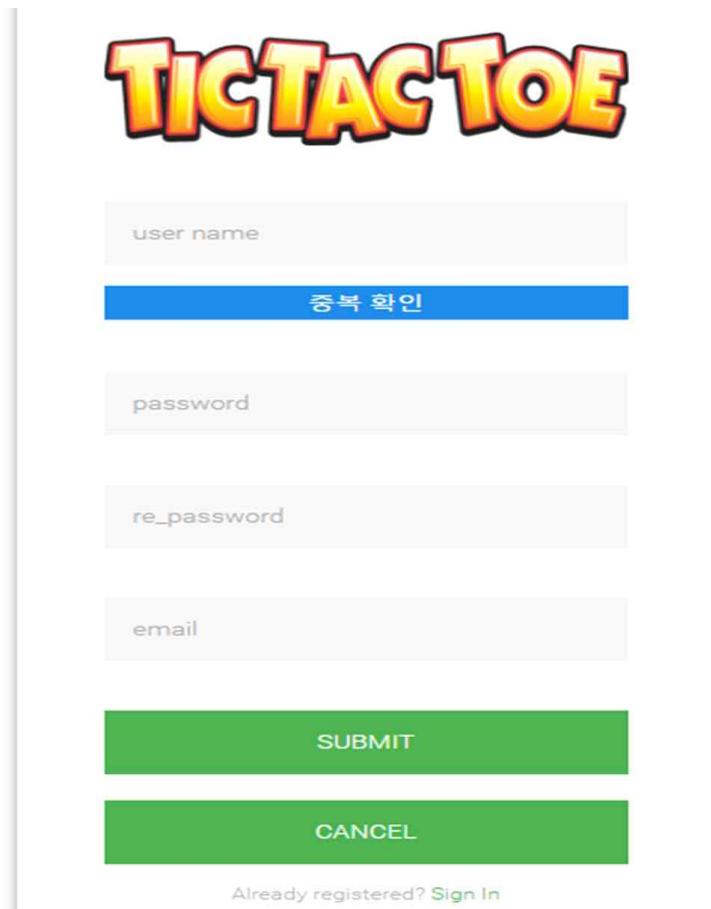
데이터베이스에 저장된 아이디와 비밀번호가 맞는지 확인합니다.

② protected void Page_Load

세션 안에 문자열로 정보가 저장되어 있으면 MainPage.aspx 페이지로 이동합니다.

Login버튼을 키보드의 enter키를 이용해서 누를 수 있게도 하였습니다.

- JoinPage.aspx



JoinPage.aspx(회원가입 페이지)

○ 회원가입을 위한 페이지입니다.

- 텍스트 박스와 버튼을 사용하여 회원가입 폼을 만듭니다.

- 아이디와 비밀번호 정보, 이메일 정보는 필수 입력 정보이므로 유효성 검사를 실시합니다.

- 아이디는 데이터베이스에 저장되지 않은 아이디이어야 하므로 중복확인을 합니다. 중복확인을 하지 않은 상태에서는 회원가입이 진행되지 않습니다.
- 비밀번호 재확인에 입력한 정보와 비밀번호에 입력한 정보가 같은지 확인하는 유효성 검사를 합니다.
- CANCEL 버튼과 Sign in 버튼을 이용하여 돌아갈 수 있습니다.

■JoinPage.aspx.cs

-컨트롤러

① btnsubmit_Click

제출을 위한 버튼으로 텍스트 박스에 입력한 아이디와 비밀번호가 모두 있으며 유효성 검사 조건이 충족되면 회원가입이 완료가 되고 LoginPage.aspx 로 돌아가 로 그인 할 수 있도록 합니다.

② btnback_Click

뒤로 가기 위한 버튼으로 클릭하면 다시 LoginPage.aspx 로 이동하게 됩니다. 아이디, 비밀번호, 비밀번호 확인은 꼭 입력해야하는 정보이기 때문에 유효성검사 컨트롤을 사용합니다. 아이디는 RequiredFieldValidator, 비밀번호는 RequiredFieldValidator, RegularExpressValidator, 비밀번호 확인은 RequiredFieldValidator, CompareValidator 컨트롤을 사용합니다.

③ btnCompareId_Click

현재 사용자가 작성한 아이디가 데이터베이스에 저장되어 있는지 확인하도록 합니다. 이미 저장되어 있는 아이디라면 중복된 ID라고 버튼 밑에 뜨고, 저장되어 있지 않다면 사용할 수 있는 ID라고 뜹니다. 중복된 ID일 경우 제출 버튼을 눌러도 가입 이 되지 않습니다.

- MainPage.aspx

○ 로그인 후에 보이는 초기 화면입니다.

- 친구와 함께 대결하기, 컴퓨터와 대결하기 : 대결하는 페이지로 넘어갑니다.
- 게임 기록 : 지금까지 대결한 기록들을 보는 페이지로 넘어갑니다. 전체 경기 기록과 사용자 자신의 기록을 볼 수 있습니다.
- 만든 이들 : 제작자 정보 페이지로 이동합니다.
- 로그아웃 : 로그아웃 후 로그인 페이지로 이동합니다.

TICTACTOE

admin님 안녕하세요~

친구와 함께 대결하기

컴퓨터와 대결하기

게임기록

만든이

로그아웃

MainGame.aspx(메인 게임 페이지)

■ MainPage.aspx.cs

- 컨트롤러

① btnGamePlayer_Click

사용자가 또 다른 사용자와 게임을 시작할 때 마우스를 클릭하면 발생하는 이벤트입니다. **친구와 함께 대결하는 정보를 세션 정보에 저장합니다.**

② btnGameCom_Click

사용자가 컴퓨터와 게임을 시작할 때 마우스를 클릭하면 발생하는 이벤트입니다. **컴퓨터와 대결하는 정보를 세션에 저장합니다.**

③ btnRecord_Click

전체 경기 기록과 자신의 기록을 보는 페이지로 이동하기 위해 클릭하면 발생하는 이벤트입니다.

④ btnMadeby_Click

제작자 정보가 저장된 페이지로 이동하기 위해 클릭하면 발생하는 이벤트입니다.

⑤ btnLogout_Click

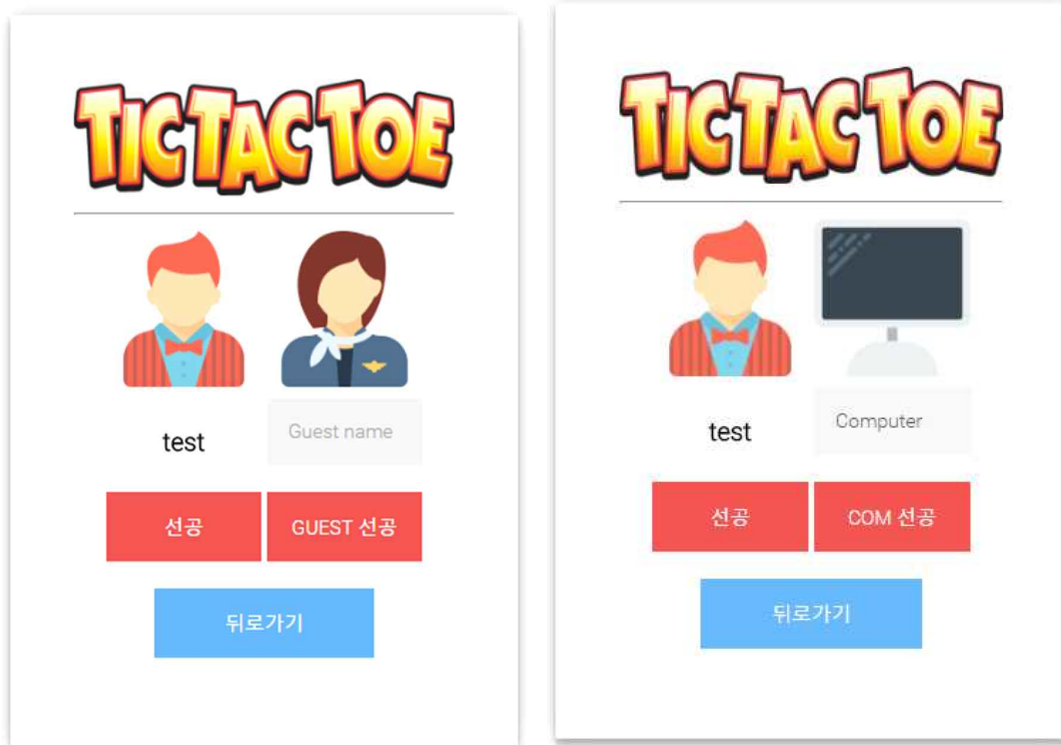
사용자가 로그아웃을 할 때 발생하는 이벤트입니다. 만든 세션들을 종료합니다.

- 함수

① protected void Page_Load

사용자 정보를 저장하는 세션에 정보가 담겨 있는 경우에만 이동 가능하도록 합니다. 그 외에는 LoginPage.aspx 페이지로 이동합니다.

- SubGamePage.aspx



SubGamePage.aspx(선포 정하기 페이지)

○ 게임 형식을 선택하는 화면입니다.

- 버튼 3개와 텍스트 박스 1개를 이용합니다.
- 화면에 출력된 선포 버튼을 통해 누가 먼저 게임을 시작할지 정합니다. 버튼을 누르면 게임이 시작됩니다.
- 텍스트 박스에는 다른 플레이어의 이름을 저장할 수 있습니다. 컴퓨터와 대결 시에는 이름을 입력할 수 없고 Computer라고 뜹니다.
- 라벨 1개를 추가하여 사용자 아이디를 출력합니다.
- 뒤로 가기 버튼을 클릭하면 메인 화면으로 이동합니다.

■ SubGamePage.aspx.cs

- 컨트롤러

① btn_Click

선공 및 후공을 선택하는 버튼의 이벤트를 처리를 담당합니다. 플레이어들의 정보는 세션으로 관리합니다. 선공을 정하면 GamePage.aspx로 이동합니다.

② btnBack_Click

이전페이지로 넘어갑니다.

- 함수

① protected void Page_Load

사용자 정보를 저장하는 세션에 정보가 담겨 있는 경우에만 이동 가능하도록 합니다. 그 외에는 LoginPage.aspx 페이지로 이동합니다. 플레이어1의 아이디를 세션으로 넘겨받고, 화면에 보이는 이미지와 글을 보이게 합니다.

- GamePage.aspx



GamePage.aspx(게임 실행 페이지)

○ 게임 실행 화면입니다.

- 화면 우측에는 어느 플레이어가 공격인가를 알려줍니다.
- 화면 좌측에는 사용자가 게임을 하는 공간입니다.

- 화면 우측 하단에는 게임을 나가는 버튼입니다.

■ GamePage.aspx.cs

- 컨트롤러

① btn_TTT_Click

사용자가 게임에서 수를 두고 싶을 때 마우스를 클릭하면 발생하는 이벤트 게임모드와 누구의 턴인지에 따라 O 또는 X로 알을 놓게 됩니다.

- 함수

① void Init()

변수들을 초기화 및 세팅합니다.

② void gameStart(int m, int firstPlayer)

모드와 선공, 후공에 따라 뷰를 업데이트하고 만약 컴퓨터가 선공이면 comAttack() 함수를 호출합니다.

③ bool checkBinggo(int player)

각 플레이어의 매 턴마다 빙고여부를 확인하여 true, false 값을 반환합니다.

④ void comAttack()

가로, 세로, 대각선을 체크한 후 공격할 수 있는 곳과 방어 할 수 있을 곳을 찾습니다. 그리고 공격을 우선순위로 말을 놓게 됩니다.

⑤ void endgame(int winner)

checkBinggo에서 받은 값이 true면 이 함수를 실행시켜 게임 결과를 저장하고 승리했다는 것을 알려줍니다.

⑥ void viewUpdateState()

삼목 판의 게임 상태, 모드, 턴을 저장하기 위한 코드를 함수로 만들어 놓은 것입니다.

- RecordPage.aspx / RecordPpage.aspx

- 경기 기록을 보는 화면입니다.

- 버튼 3개과 GridView 컨트롤을 이용합니다.

- GridView 컨트롤에서 10개씩 보이도록 합니다.

- 플레이어1, 플레이어2, 경기결과를 데이터베이스에서 출력합니다.

- 전체 기록보기와 사용자 기록보기를 통해서 볼 수 있습니다. select문을 이용하여 GridView 컨트롤에 보여줍니다.

- RecordPage.aspx와 RecordPpage.aspx는 같은 디자인이므로 하나로 묶었습니다.



RecordPage.aspx(전체 기록보기 페이지)



RecordPpage.aspx(내 기록 보기 페이지)

■ RecordPage.aspx.cs

- 컨트롤러

① btnBack_Click

나가기 버튼을 클릭하면 발생하는 이벤트로 MainPage.aspx로 이동합니다.

② btnWhole_Click

전체 기록보기 버튼을 클릭할 때 발생하는 이벤트로 전체 기록 보기 페이지 (RecordPage.aspx)로 이동합니다.

- 함수

① void Page_Load()

전체 기록 보기 버튼을 비활성화 하여 다시 클릭하지 않도록 합니다.

■ RecordPpage.aspx.cs

- 컨트롤러

① btnBack_Click

나가기 버튼을 클릭하면 발생하는 이벤트로 MainPage.aspx로 이동합니다.

② btnPersonal_Click

내 기록보기 버튼을 클릭할 때 발생하는 이벤트로 내 기록 보기 페이지

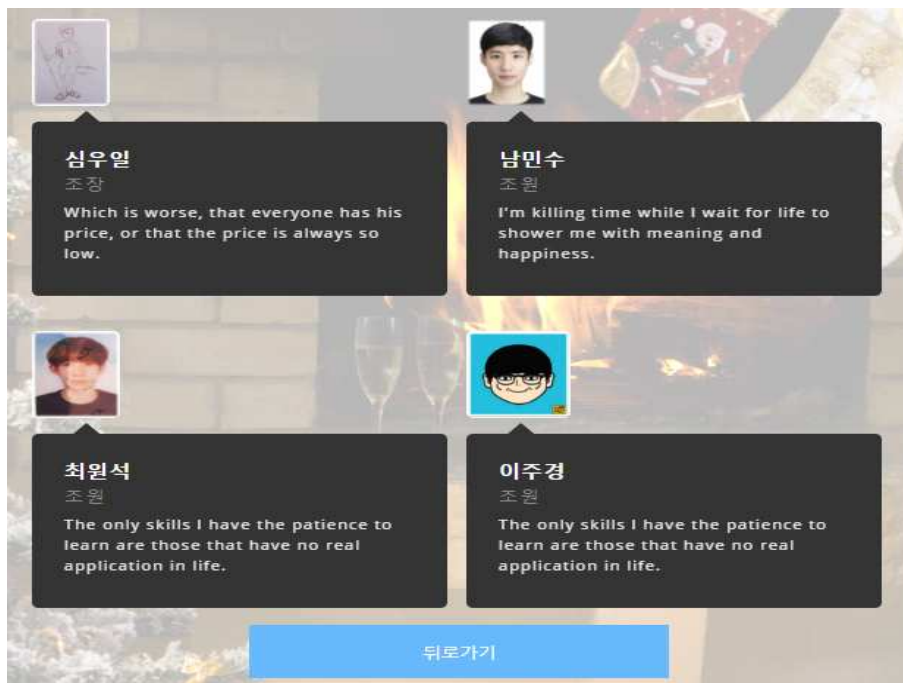
(RecordPpage.aspx)로 이동합니다.

- 함수

① void Page_Load()

페이지 로드 시 로그인 한 사용자의 기록을 화면에 보여줍니다. 내 기록 보기 버튼을 비활성화 하여 다시 클릭하지 않도록 합니다.

- Produces.aspx



Produces.aspx(제작자 정보 페이지)

○ 웹 어플리케이션 제작자 정보를 나타내는 화면입니다.

- 팀원의 학번과 이름을 나타내고, 뒤로 가기 버튼은 메인 페이지로 이동합니다.

5. 기타

- 발표 후 추가사항

코드 상 추가된 사항은 없습니다.

- 프로젝트에 대한 팀원 의견

최원석 : 웹 프로그래밍을 배웠었지만 디자인뿐만 아니라 기능적인 부분에서 이번 강의를 통해서 더 심화된 부분을 배웠던 것 같다. 힘들고 어려웠던 부분도 많았던

것 같다. 그래도 한 달이라는 시간동안 팀원들끼리 프로젝트 하나를 완성했다는 것에 보람을 느꼈으며, 교수님을 포함한 팀원들 모두 수고하셨습니다!!!

남민수 : 웹을 처음 접해보아서 감도 잘 안 오고 어려웠지만, 프로젝트를 통해 수업시간에 배운 것을 토대로 개발을 진행해보니 어느 정도 감이 잡혀서 좋았습니다.

이주경 : 닷넷이라는 언어를 이용해서 처음 프로젝트를 진행해 보았는데 수업 때 배운 내용이나 그걸 응용해서 작동하는 페이지를 직접 해보게 되어 좋았습니다.

심우일 : 실습 시간에 배운 내용을 응용하여 프로젝트를 하게 되어 흥미로웠습니다. 특히 별도의 데이터베이스 없이 세션과 쿠키만으로 정보들을 주고받는 것이 매우 흥미로웠고 좋은 경험이 되었습니다.