

데이터베이스 시스템 설계 프로젝트 보고서



Instagram

5조

201514755 이재열

201514757 이제석

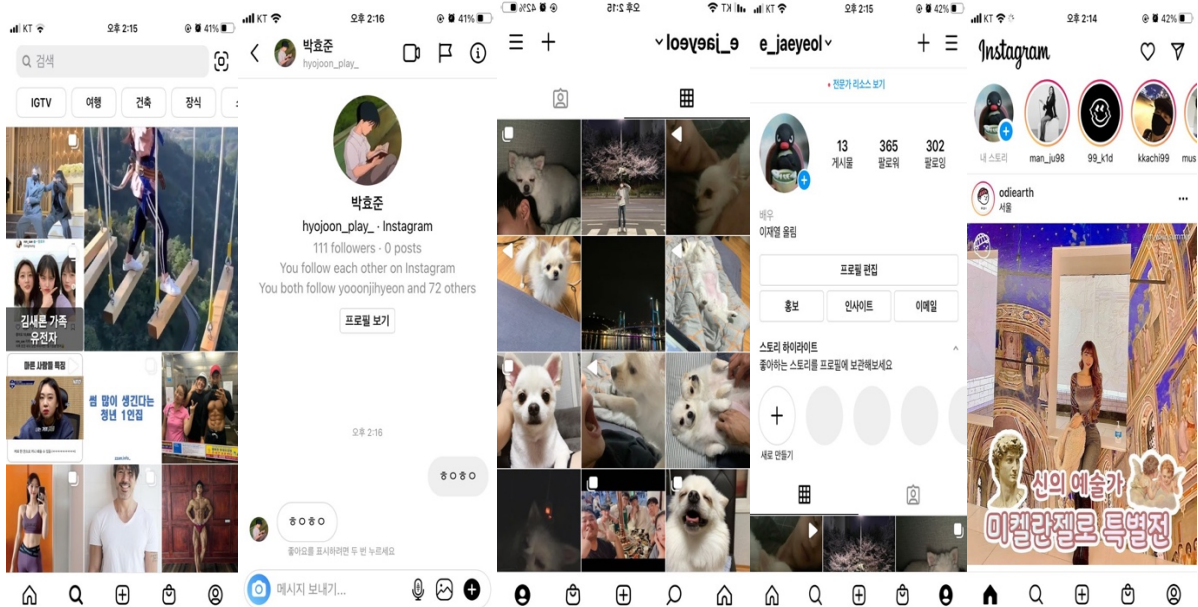
201514707 박효준

201716388 박성필

1. 프로젝트 주제

인스타그램(SNS) : 온라인 사진 공유 및 소셜 네트워크 서비스

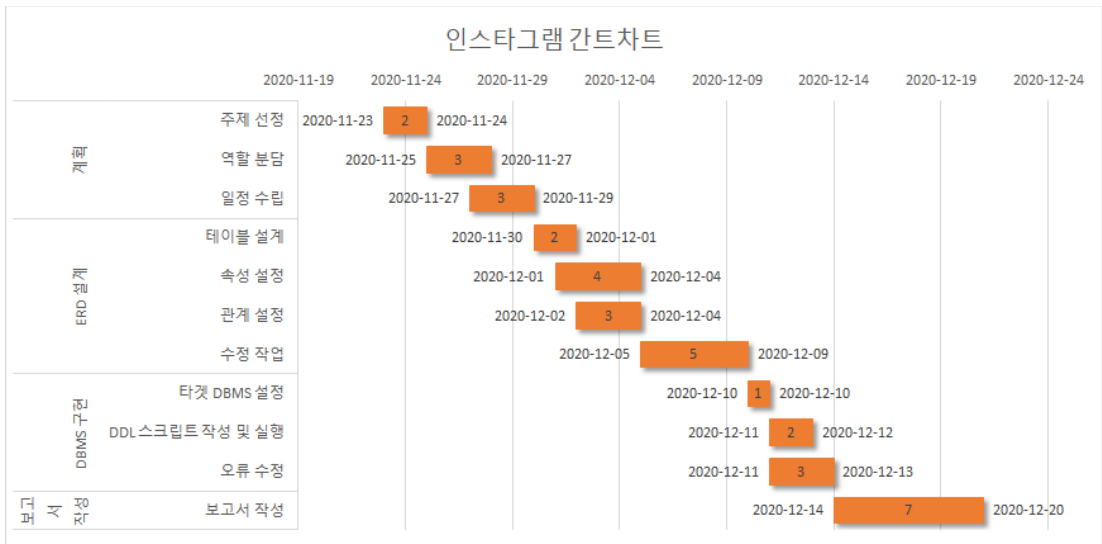
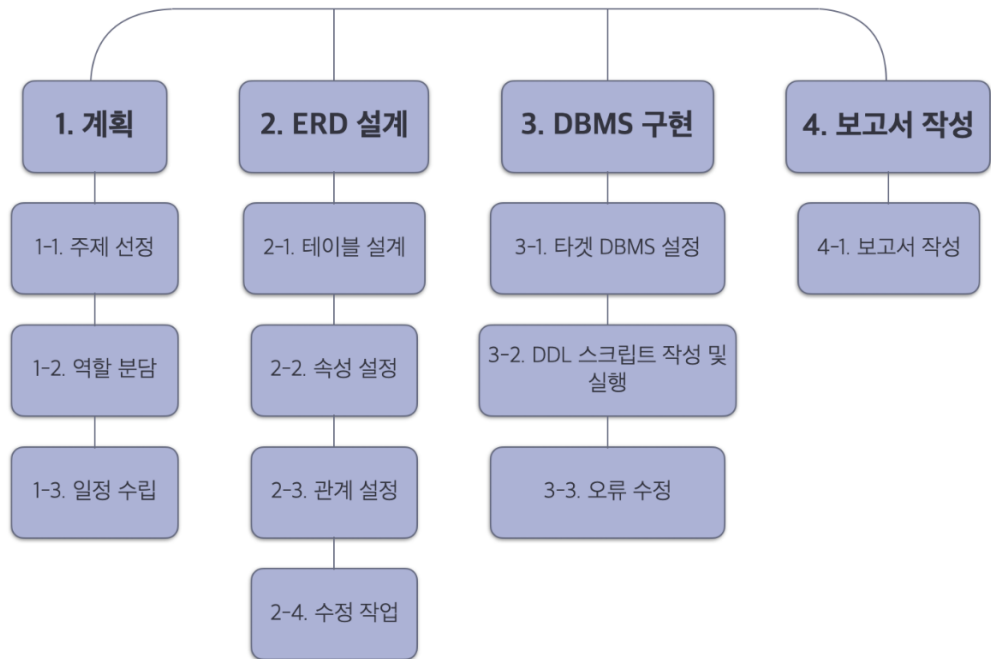
주제 선정 이유 : 현재 가장 인기가 많은 SNS 중 하나로 국내외 수많은 사용자의 데이터와 게시물 데이터 등 다양한 테이블과의 관계로 이루어져 있기 때문에 데이터베이스를 설계하고 이를 다루기에 적절하다고 생각하여 선정함.



2. 팀원 역할 분배 (LRC)

	박성필	박효준	이재열	이제석
1-1. 주제 선정	1	1	1	1
1-2. 역할 분담	1	1	1	1
1-3. 일정 수립	1	1	1	1
2-1. 테이블 설계	2	3	1	3
2-2. 속성 설정	1	2	3	3
2-3. 관계 설정	3	3	2	1
2-4. 수정 작업	3	1	3	2
3-1. 타겟 DBMS 설정	3	2	1	3
3-2. DDL 스크립트 작성 및 실행	1	3	3	2
3-3. 오류 수정	2	1	2	1
4-1. 보고서 작성	1	1	1	1
키				
1	메인 담당			
2	서브 담당			
3	리뷰			

3. WBS와 간트 차트



4. 회의록

회의록		
날짜	내용	비고
11월 22일	<ul style="list-style-type: none"> - 주제 선정(SNS - 인스타그램) - 역할 분담(테이블 설계) 박호준 : 스토리, 스토리 읽은 사람 박성필 : 유저, 팔로우 이재열 : 게시물, 댓글 이제석 : DM(Direct Message), 게시물 좋아요 	
12월 1일	<ul style="list-style-type: none"> - ERD 구현(Toad) - 테이블 설계 보충 - 테이블 별 속성 설정 및 네이밍 - 테이블 기술서 작성 - 관계 설정 - ERD 수정 	
12월 8일	<ul style="list-style-type: none"> - DBMS 구현 - ERD 수정 - DDL 스크립트 작성 	
12월 11일	<ul style="list-style-type: none"> - 테이블 속성 이름 수정 - 관계 수정 - 최종 DDL 스크립트 작성 - DDL 스크립트 실행 및 오류 수정 	
12월 19일	<ul style="list-style-type: none"> - 보고서 작성 	

5. 테이블 기술서

Name	USER	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	박성필		
Description	회원					
No	Column	Data Type	NN	KY	Default	Description
1	user_id	Bigint	O	PK		사용자 아이디
2	user_login_id	Varchar(255)	O	X		로그인 아이디
3	nickname	Varchar(255)	X	X		이름
4	website	Varchar(255)	X	X		웹사이트
5	intro	Varchar(255)	X	X		자기소개
비고						
FK (user_id) -> POST(user_id) , DM_ROOM(user_id1,user_id2) , STORY(user_id), STORY_READER(user_id) , FOLLOWING(follower_id, following_id) , LIKER(likier_id) , COMMENT(user_id)						

Name	POST	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이재열		
Description	게시물					
No	Column	Data Type	NN	KY	Default	Description
1	post_id	Bigint	O	PK		게시물 아이디
2	user_id	Bigint	O	FK		작성자 아이디
3	post_date	Timestamp	O	X	current_timestamp	게시 시간
4	post_contents	Varchar(255)	X	X		게시물 내용
비고						
FK (post_id) -> COMMENT(post_id) , MEDIA(post_id) , LIKER(post_id)						

Name	DM_ROOM	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이제석		
Description	디엠 채팅방					
No	Column	Data Type	NN	KY	Default	Description
1	dm_room_id	Bigint	O	PK		채팅방 아이디
2	user_id1	Bigint	O	FK		참가자1
3	user_id2	Bigint	O	FK		참가자2
비고						
FK (dm_room_id) ->DM_MESSAGE(dm_room_id)						

Name	DM_MESSAGE	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이제석		
Description	디엠 메세지					
No	Column	Data Type	NN	KY	Default	Description
1	message_id	Bigint	O	PK		메세지 아이디
2	dm_room_id	Bigint	O	FK		채팅방 아이디
3	dm_date	Timestamp	O	X		메세지 보낸 날짜
4	dm_contents	Varchar(255)	O	X		메세지 내용
5	sender_number	Bit(1)	O	X		보낸 사람 번호
비고						

Name	MEDIA	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이재열		
Description	게시물 사진, 동영상 링크					
No	Column	Data Type	NN	KY	Default	Description
1	media_id	Bigint	O	PK		미디어 아이디
2	post_id	Bigint	O	FK		미디어가 게시된 게시물 아이디
3	media_url	Varchar(255)	O	X		미디어 링크
비고						

Name	STORY	Table 기술서	작성일	11월22일		
System	Instagram		작성자	박효준		
Description	스토리					
No	Column	Data Type	NN	KY	Default	Description
1	story_id	Bigint	O	PK		스토리 아이디
2	user_id	Bigint	O	PFK		스토리 게시자 아이디
3	story_date	Timestamp	O	X	current_time stamp	스토리 게시 시 간
4	story_url	Varchar(255)	O	X		스토리 미디어 링크
비고						
FK(story_id) -> STORY_READER(story_id)						

Name	STORY_READER	Table 기술서	작성일	11월22일		
System	Instagram		작성자	박효준		
Description	스토리 읽은 사람					
No	Column	Data Type	NN	KY	Default	Description
1	user_id	Bigint	O	PFK		읽은 사람 아이디
2	story_id	Bigint	O	PFK		스토리 아이디
비고						

Name	FOLLOWING	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	박성필		
Description	팔로잉					
No	Column	Data Type	NN	KY	Default	Description
1	follower_id	Bigint	O	PFK		팔로우 하는 사람 아이디
2	following_id	Bigint	O	PFK		팔로우 받는 사람 아이디
비고						

Name	LIKER	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이제석		
Description	게시물 좋아요 누른 사람					
No	Column	Data Type	NN	KY	Default	Description
1	liker_id	Bigint	O	PFK		좋아하는 사람 아이디
2	post_id	Bigint	O	PFK		게시물 아이디
비고						

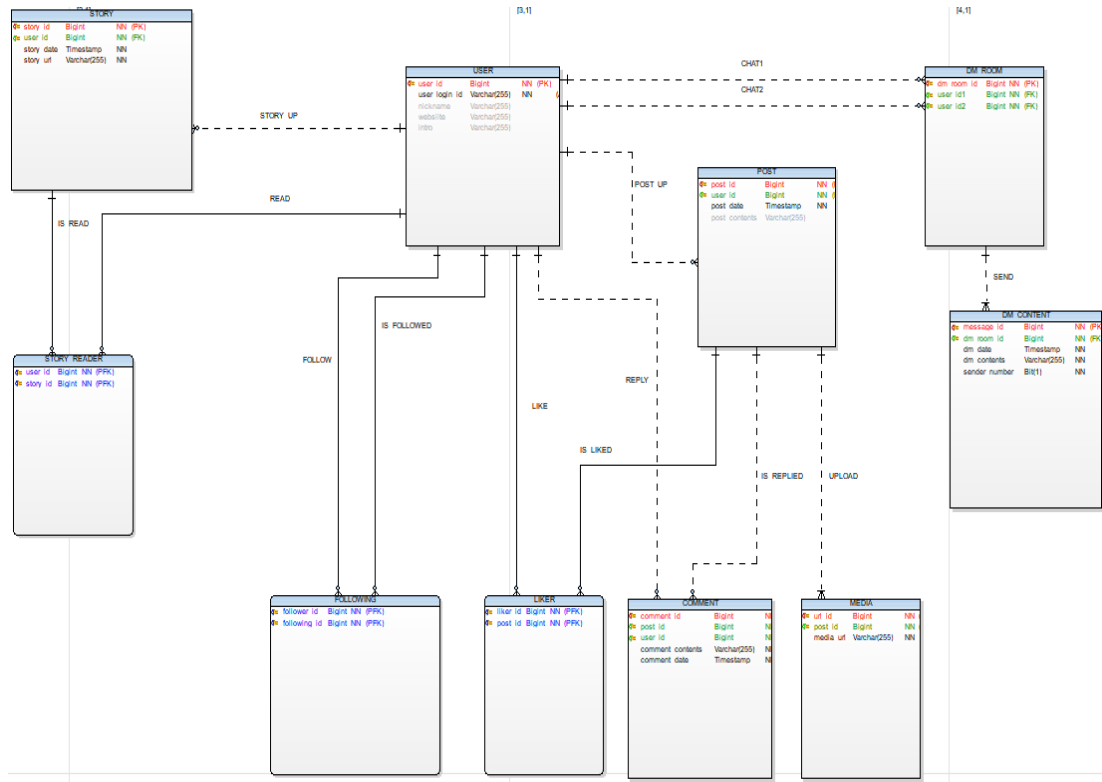
Name	COMMENT	Table 기술서	작성일	11월 22일		
System	Instagram		작성자	이재열		
Description	게시물 댓글					
No	Column	Data Type	NN	KY	Default	Description
1	comment_id	Bigint	O	PK		댓글 아이디
2	post_id	Bigint	O	FK		게시물 아이디
3	user_id	Bigint	O	FK		유저 아이디
4	comment_contents	Varchar(255)	O	X		댓글 내용
5	comment_date	Timestamp	O	X	Current_time stamp	댓글 작성 시간
비고						

6. 용어사전

도메인	물리명	데이터타입	초기값	값의범위	엔티티	테이블	설명
사용자 아이디	user_id	Bigint			USER	사용자	사용자 구분
로그인 아이디	user_login_id	Varchar(255)			USER	사용자	사용자의 로그인 정보
이름	nickname	Varchar(255)			USER	사용자	사용자 이름
웹사이트	website	Varchar(255)			USER	사용자	사용자 웹사이트
자기소개	Intro	Varchar(255)			USER	사용자	사용자 자기소개
게시물 아이디	post_id	Bigint			POST	게시물	게시물 구분
게시자 아이디	user_id	Bigint			POST	게시물	게시한 사용자 구분
게시날짜	post_date	Timestamp	현재날짜시간		POST	게시물	게시한 날짜
게시내용	post_contents	Varchar(255)			POST	게시물	게시 글
디엠 아이디	dm_room_id	Bigint			DM_ROOM	DM	디엠 구분
사용자 아이디 1	user_id1	Bigint			DM_ROOM	DM	채팅인원 1 구분
사용자 아이디 2	user_id2	Bigint			DM_ROOM	DM	채팅인원 2 구분
메시지 아이디	message_id	Bigint			DM_CONTENT	DM 메시지	메시지 구분
디엠 아이디	dm_room_id	Bigint			DM_CONTENT	DM 메시지	디엠 구분
디엠 날짜	dm_date	Timestamp	현재날짜시간		DM_CONTENT	DM 메시지	디엠 보낸 날짜
디엠 내용	dm_contents	Varchar(255)			DM_CONTENT	DM 메시지	디엠 내용
보낸 사람	sender_number	Bit(1)		(0,1)	DM_CONTENT	DM 메시지	디엠 보낸사람 구분
링크 아이디	url_id	Bigint			MEDIA	게시물내용	게시물의 내용 구분
게시물 아이디	post_id	Bigint			MEDIA	게시물내용	게시물 구분
게시물 링크	media_url	Varchar(255)			MEDIA	게시물내용	게시물 내용 링크
댓글 아이디	comment_id	Bigint			COMMENT	댓글	댓글 구분
게시물 아이디	post_id	Bigint			COMMENT	댓글	게시물 구분
댓글 작성자 아이디	user_id	Bigint			COMMENT	댓글	댓글 작성자 구분
댓글 내용	comment_contents	Varchar(255)			COMMENT	댓글	댓글 내용
댓글 날짜	comment_date	Timestamp	현재날짜시간		COMMENT	댓글	댓글 날짜
좋아하는 사용자 아이디	liker_id	Bigint			LIKER	좋아요	좋아요 누른 사람 구분
게시물 아이디	post_id	Bigint			LIKER	좋아요	게시물 구분
팔로우 받는 사람 아이디	follower_id	Bigint			FOLLOWING	팔로잉	팔로우 받는 사람
팔로우 하는 사람 아이디	following_id	Bigint			FOLLOWING	팔로잉	팔로우 하는 사람
읽은 사람 아이디	user_id	Bigint			STORY_READER	읽은사람 목록	읽은 사람 아이디
스토리 아이디	story_id	Bigint			STORY_READER	읽은사람 목록	스토리 구분
스토리 아이디	story_id	Bigint			STORY	스토리	스토리 구분
스토리 게시자 아이디	user_id	Bigint			STORY	스토리	스토리 게시한 사용자 아이디
스토리 날짜	story_date	Timestamp			STORY	스토리	스토리 올린 날짜
스토리 링크	story_url	Varchar(255)			STORY	스토리	스토리 내용 링크

7. 구축 화면

ERD



스크립트 생성 및 실행

```
CREATE TABLE `LIKER`
(
  `liker_id` Bigint UNSIGNED NOT NULL,
  `post_id` Bigint UNSIGNED NOT NULL
)
;

ALTER TABLE `LIKER` ADD PRIMARY KEY (`liker_id`, `post_id`)
;

-- Table DM_CONTENT
CREATE TABLE `DM_CONTENT`
(
  `message_id` Bigint UNSIGNED NOT NULL AUTO_INCREMENT,
  `dm_room_id` Bigint UNSIGNED NOT NULL,
  `dm_date` Timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `dm_contents` Varchar(255) NOT NULL,
  `sender_number` Bit(1) NOT NULL,
  PRIMARY KEY (`message_id`)
)
;

-- Create foreign keys (relationships) section -----
ALTER TABLE `POST` ADD CONSTRAINT `POST_UP` FOREIGN KEY (`user_id`) REFERENCES `USER` (`user_id`) ON DELETE RESTRICT ON UPDATE RESTRICT
;

ALTER TABLE `FOLLOWING` ADD CONSTRAINT `IS_FOLLOWED` FOREIGN KEY (`following_id`) REFERENCES `USER` (`user_id`) ON DELETE RESTRICT ON UPDATE RESTRICT
;

ALTER TABLE `STORY` ADD CONSTRAINT `STORY_UP` FOREIGN KEY (`user_id`) REFERENCES `USER` (`user_id`) ON DELETE RESTRICT ON UPDATE RESTRICT
;

ALTER TABLE `STORY_READER` ADD CONSTRAINT `IS_READ` FOREIGN KEY (`story_id`) REFERENCES `STORY` (`story_id`) ON DELETE RESTRICT ON UPDATE RESTRICT
;
```

```
mysql> source ~/Desktop/instagram.sql
```

```
mysql> show tables;
+-----+
| Tables_in_instagram |
+-----+
| COMMENT              |
| DM_CONTENT           |
| DM_ROOM              |
| FOLLOWING             |
| LIKER                |
| MEDIA                |
| POST                 |
| STORY                |
| STORY_READER         |
| USER                 |
+-----+
10 rows in set (0.00 sec)
```

8. 프로젝트 후기

박성필	사실 쉬운 것 같았는데 관계를 설정할 때 생각보다 어려움을 느꼈습니다. 데이터베이스를 설계할 때와, 실제 DBMS에 반영할 때에 여러 가지 사항을 고려해야함을 이번 과제를 수행하면서 느끼게 되었습니다. 특히 관계를 설정할 때 힘들었는데, 처음에 identifying과 non-identifying을 잘 구분하지 못해서 어려움이 있었고, 모델링 도구를 다루는데 익숙하지 않아서 어려움이 있었습니다. 또한 속성이나 테이블 명명 규칙에 대한 것들이 명확하지 않아서 어려움이 있었습니다. 하지만 구글링과 팀원들 간의 협력을 통해 해결하면서 프로젝트를 무사히 완수할 수 있었습니다. 그래서 값진 경험이었다고 생각합니다.
박호준	기존에 컴퓨터공학부에서는 SQL문 작성과 같은 DBMS에 대한 공부보다는 실습 위주의 수업을 하였고 이로 인해 DBMS가 무엇이고 이를 실무에 어떤 방식으로 적용하는지에 대해서 배우지 못했었습니다. 데이터베이스시스템설계 과목을 수강하고난 후에는 DBMS가 어떠한 것이고 설계가 무엇이며 이 설계를 통해서 실무에서 팀원들과 어떻게 작업을 해나가는지에 대한 학습을 하고 직접 경험해볼 수 있는 경험이라서 좋았습니다.
이재열	데이터베이스 수업만 듣고 sql문을 작성해 테이블을 만들어보고 조인을 해본 게 데이터베이스의 전부였습니다. 프로젝트를 시작하기 전에 생각한 것은 "ERD 뭐 그거 그냥 만들면되지" 이렇게 생각했었는데 직접 erd다이어그램을 만들어보면서 물리적DB와 논리적DB 사이의 테이블의 이름을 달리 하는 것, 속성 이름을 만드는 것 등 다양한 문제가 있었습니다. 속성 이름을 지으면서 어떤 이름으로 할지 고민을 많이 하였고 관계를 직접 그리는 것 또한 처음 해보았기 때문에 복잡한 테이블 사이의 관계를 어떻게 할지 고민을 많이 했습니다. 현 프로젝트에서는 간단한 테이블 설계만을 했지만 실제로는 훨씬 더 많은 테이블과 관계 속성 등으로 이루어져 있을 것이기에 생각만큼 쉬운 과목은 아니라고 생각했습니다. 데이터베이스 프로젝트를 하면서 직접 DB설계를 해볼 수 있어서 정말 좋았습니다.
이제석	컴퓨터 공학과에서 배운 데이터베이스 수업에서는 mysql를 통해 각종 sql 구문만을 배웠지만, 이번 수업을 통해 DBMS를 어떻게 사용 하는지와 전체적인 설계 흐름을 배우게 되어 좋았습니다. 또한 Toad를 통해 DBMS의 구조를 좀 더 직관적으로 설계하는 법을 알게 되고 앞으로의 다른 DBMS를 사용할 때에 많은 도움이 될 거 같아 좋았습니다.