

# 멋쟁이사자처럼 심화 스터디 가이드라인

5주차

LIKE LION

01

---

## 5주차 목표

```
week5 = %w(re del).map{ |s| 'Mo' + s }
```

- irb 나 rails console에 쳐보세요!
- 따로 Ruby 공부를 하지 않았는데, 이 코드를 보는것 만으로 이해 했다면?

당신은 4기강의를 끝까지 정말 열심히 들은 분입니다. ㅇㅈ

LIKE LION

02

---

5주차 개인 Todo



사전 필수 복습 자료

4기 강의 10주차

6(검색 구현-1), 7(검색 구현-2), 8(모델 코딩),  
10(M:N), 11(Hash, Validation) 번 강의  
다시 듣고 소화하기

\*목표 슬라이드의 Ruby 코드 다 여기에 나와요!

LIKE LION

03

---

5주차 팀 Todo



4주차 게시판 Upgrade

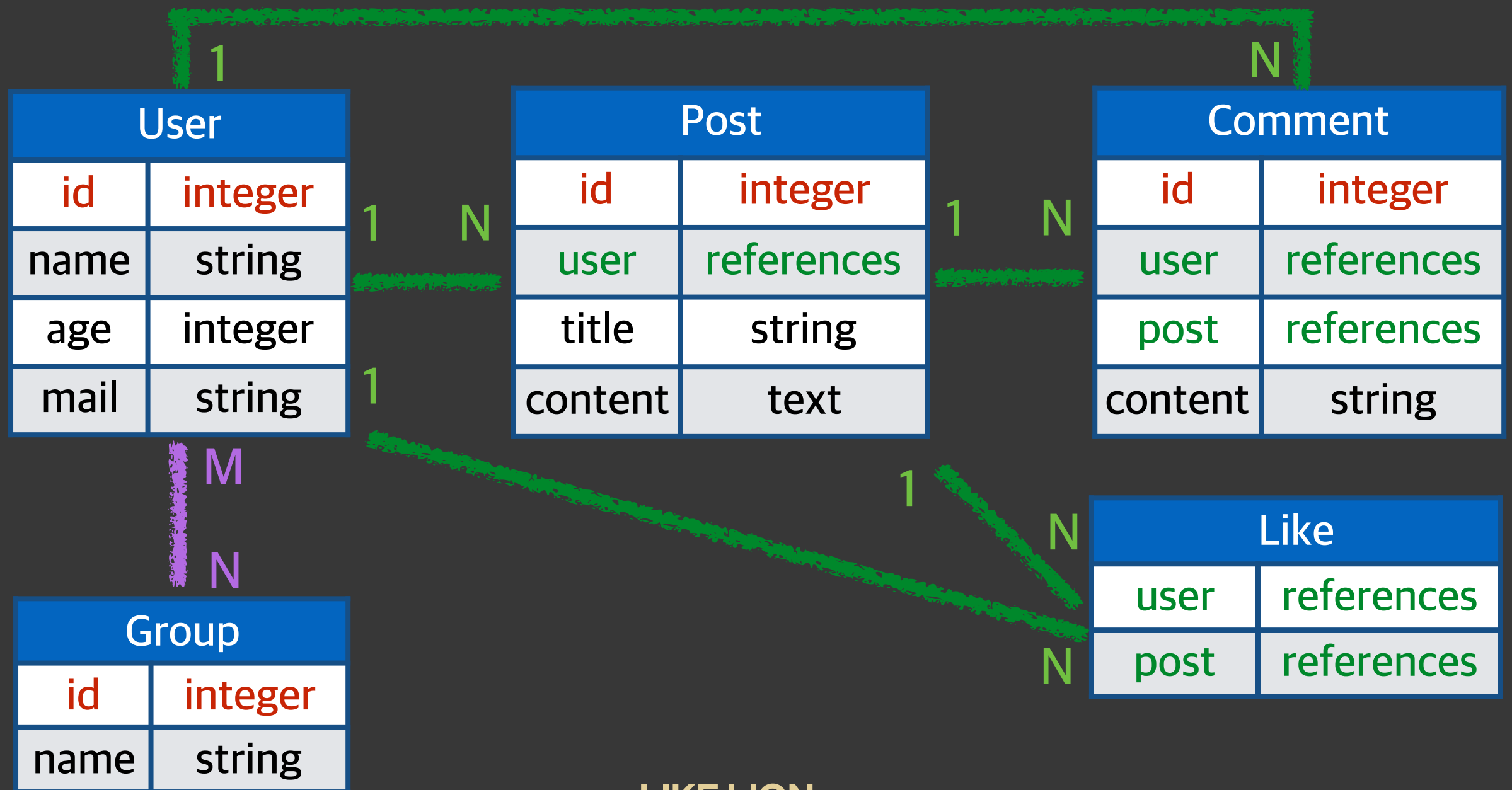
-미리 혼자 고민해 보는 시간을 가지는 걸 추천.

LIKE LION

03

저번주 과제

# ERD(Entity Relation Diagram)



LIKE LION

03

---

5주차 팀 Todo

이걸 더 Upgrade 해보자!  
일단 기존의 데이터는 다 drop 하고  
새로운 마음으로 고고싱!

..하기 전에,

LIKE LION

03

---

5주차 팀 Todo

# Validation

[http://guides.rubyonrails.org/  
active\\_record\\_validations.html](http://guides.rubyonrails.org/active_record_validations.html)

퍼펙트 루비 온 레일즈 - 5.5 (유효성 검사 구현)

03

---

5주차 팀 Todo

# Validation

여러분은 사용자를 얼마나 믿으시나요?

구글 설문지만 예로 들어도 검증을 하지 않는다면 정말 많은 오타와  
지켜지지 않는 형식으로 제출 됩니다.

네. 아무도 믿지 마세요



# Validation

jQuery Validation 처럼 스크립트로 검증할 수 있지 않나요?!

네. 클라이언트측에서 검증은 서버를 거치지 않기 때문에 서버 부하를 줄여주거나 더 빠른 피드백을 전달해 줄 수 있습니다.

다만, 스크립트가 동작하지 않을 경우 등 신뢰를 할 수 없습니다.

반드시 서버측에서도 검증을 해야합니다.

(당장 모바일 브라우저에서 옵션에 가보면 [자바스크립트 끄기] 라는 옵션이 있어요..)

# Validation

데이터 검증(validation)의 실행은

`create, create!, save, save!, update, update!`

`update_attributes, update_attributes!`

를 실행할 경우 실행되며, 유효한 경우에만 데이터베이스에 저장합니다.

물론 `save(validate: false)`를 하면 검증하지 않습니다.

03

---

5주차 팀 Todo

# Validation

우리의 Upgrade 된 Model은  
다음과 같은 데이터 검증을 필요로 합니다!

## 1. User 모델

1. **name**: 'Faker' 로 생성된 이름을 사용.  
첫 글자가 소문자일 경우에는 **대문자로 자동 변경 해서 저장**합니다.  
또한, **최대 10글자**입니다.
2. **age**: **숫자**형식으로 **20대**(20~29살)만 받습니다.
3. **mail**: '**이메일 형식**'이어야 하며, **중복은 안 됩니다**.
4. 위의 모든 항목들은 반드시 입력을 받습니다. (**not Nil**)

## 2. Post 모델

1. title: 최소 2글자, 최대 30글자입니다.
2. content: 저장하기 전에 ‘욕설 필터링’이 필요합니다.  
‘shit’, ‘fuck’, ‘hell’ 이라는 문자열이 있으면, 글자 수에 맞게 \* 으로 대체됩니다.  
(예시 => 입력: 'holy shit!' / 저장: 'holy \*\*\*\*!')
3. 위의 모든 항목들은 반드시 입력을 받습니다. (not Nil)

## 3. Comment 모델

1. **content**: 최대 200글자를 넘을 수 없습니다.  
저장하기 전에 ‘**욕설 필터링**’이 필요합니다.  
‘shit’, ‘fuck’, ‘hell’ 이라는 문자열이 있으면, 글자 수에 맞게 \* 으로 대체됩니다.  
(예시 => 입력: ‘holy shit!’ / 저장: ‘holy \*\*\*\*!’)
2. 위의 모든 항목들은 반드시 입력을 받습니다. (**not Nil**)

## 4. Group 모델

1. name: 최대 20글자를 넘을 수 없으며, 중복은 안 됩니다.
2. 위의 항목은 반드시 입력을 받습니다. (not Nil)

02

---

## 5주차 개인 과제



위의 Validation 을 마친  
각 Model 파일들을 양식에 맞춰 제출하기



04

---

## 5주차 추가자료

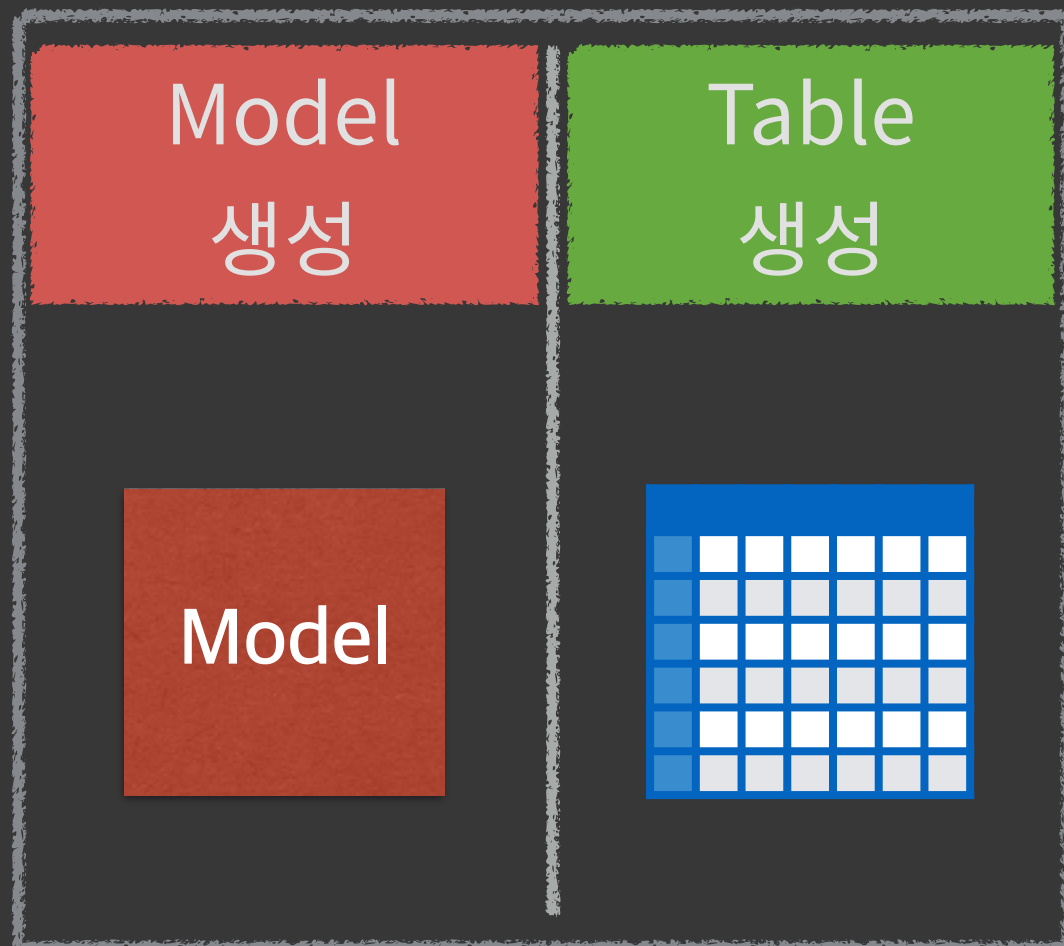


1. M:N Association
2. Validation

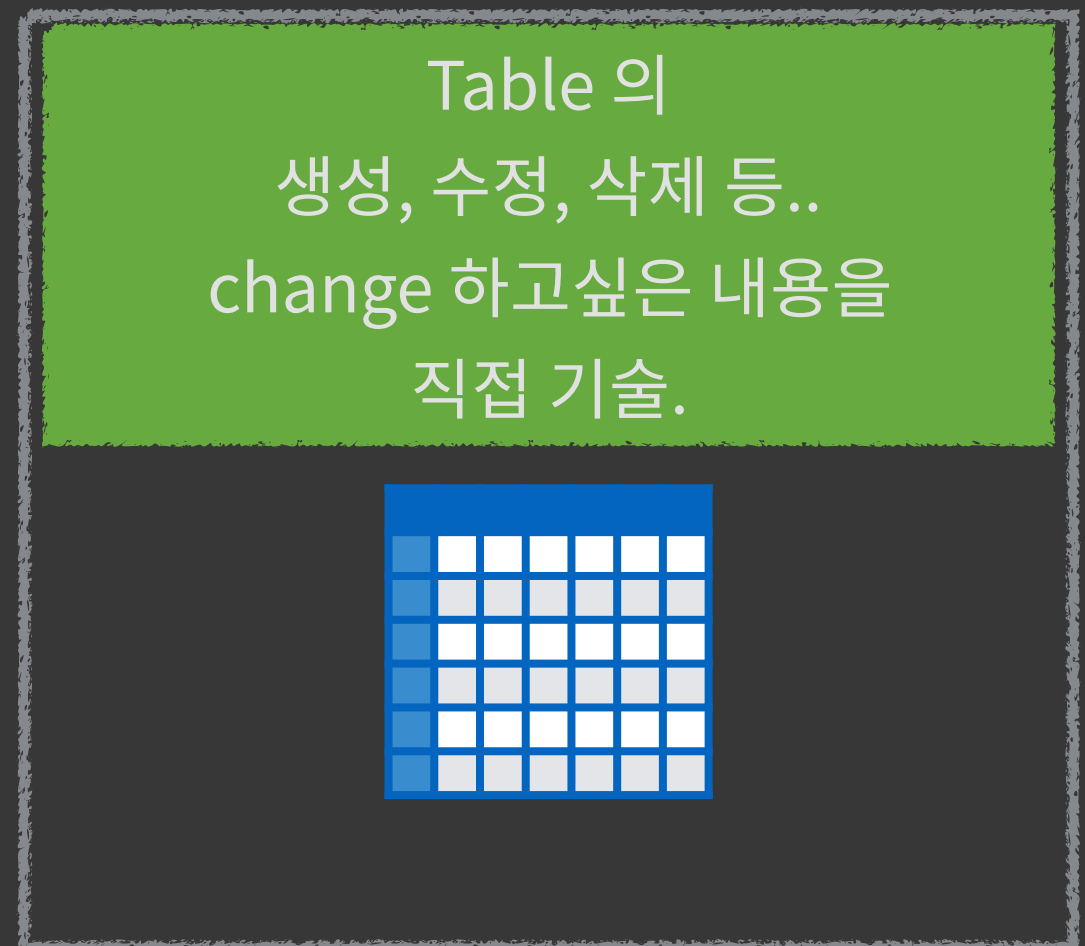
04

M : N

\$ rails g model



\$ rails g migration



## 04

---

M : N

4주차 강의 자료에서 어느정도 알아 채신 분들도 계시겠지만,

ActiveRecord::Model 없이

migration 만을 통해 DB - Table을 만들 수 있습니다.

반면,

\$ rails g model (Model 생성) 의 경우,

해당 Model 을 반영하는 table 을 만드는

migration 파일이 자동 생성됩니다.

(말이 길지만 꼭 이해하시기 바랍니다.)

그렇다면 M : N Relation을 구현하는 2가지 방법을 알아보겠습니다.

1. [http://guides.rubyonrails.org/association\\_basics.html#the-has-many-through-association](http://guides.rubyonrails.org/association_basics.html#the-has-many-through-association)  
(4기 강의에서의 M:N 은 1번 입니다.)
2. [http://guides.rubyonrails.org/association\\_basics.html#the-has-and-belongs-to-many-association](http://guides.rubyonrails.org/association_basics.html#the-has-and-belongs-to-many-association)

이번에 배우고자 하는 2가지 방법의 가장 눈에 보이는 차이는  
두 개의 테이블을 이어주는

1. **Model** 파일이 존재 하느냐 (퍼.루.레: 5.6.6)
  2. Model 파일 없이 **Table**만 존재 하느냐 (퍼.루.레: 5.6.5)
- 로 나눠서 생각해 볼 수 있습니다.

## 정리해 볼 자료

- 1번 방법과, 2번 방법의 차이와 사용하는 실제 경우.
- 왜 굳이 두가지 방법으로 나눠서 사용하는지.
- 각각의 장단점.
- 2번 방법의 경우 **migration** 파일을 작성하는 약속 등..

## Validation

## 유효성 검사 기능들 (퍼.루.레: 5.5.1)

exclusion	값이 배열 또는 범위에 포함되지 않는가
inclusion	값이 배열 또는 범위에 포함되는가
presence	값이 비어있지 않는가
absence	값이 비어있는가
uniqueness	값이 유일한가
acceptance	체크박스에 체크가 되어 있는가
confirmation	두 개의 필드가 같은가
format	정규 표현식 패턴에 맞는가
length	문자열의 길이 등
numericality	숫자의 대소, 자료형

## 04

---

# Validation

다음은 각 클래스별로 사용할 수 있는 설정 값입니다.

예를 들면, 숫자와 관련된 validation이라면 홀수냐! 짝수냐!와 같은..

(어려운 말로 매개변수..parameter..)



## 04

# Validation parameters

exclusion	in	비교 대상의 배열, 범위
inclusion	in	비교 대상의 배열, 범위
presence	-	
absence	-	
uniqueness	scope	검증에 사용 될 필드
	case_sensitive	대소문자 구분할지 (default : true)
acceptance	accept	체크 값 (default : 1)
confirmation	-	
format	with	정규표현식 패턴

## 04

# Validation parameters

length	minimum	최소 문자 길이
	maximum	최대 문자 길이
	in	문자열 길이 범위(range)
	tokenizer	문자열 분리 방법(lambda)
	is	문자열 길이(완전 일치)
	too_long	maximum 검증 오류 시 글자
	too_short	minimum 검증 오류 시 글자
	wrong_length	is 검증 오류 시 글자

## 04

# Validation parameters

numerically	only_integer	정수
	greater_than	값 초과
	greater_than_or_equal_to	값 이상
	equal_to	값 같음
	less_than	값 미만
	less_than_or_equal_to	값 이하
	odd	홀수
	even	짝수

05

## 다음주 스포일러



Rails 가 제공하는 것



내가 맹글어 쓰는 것

LIKE LION

## 06

### 정리

#### 개인별로

4기 강의 복습하고,

- app/models/user.rb
- app/models/group.rb
- app/models/post.rb
- app/models/comment.rb
- app/models/like.rb

를 제출양식에 맞춰 제출

#### 모여서

5기대상  
강의 만들어보기

- Association  
(1:1, 1:N, M:N)

#### 2/11(토) 세션에서

슬라이드 공유

&

학교별 진행상황 공유