

# 8장. 추천 기능



# 추천 기능

today(로그아웃)

장고 프레임워크로 만

1

추천

127.0.0.1:8000 내용:  
정말로 추천하시겠습니까?

확인

취소

장고 프레임워크로 만든 유명 사이트가 있는지 알려주세요.

today(로그아웃)

장고 프레임워크로 만든 사이트

2

추천

장고 프레임워크로 만든 유명 사이트가 있는지 알려주세요.

## 추천 기능

- Question, Answer 모델 변경하기

다대다 관계(ManyToMany) : 글 1개에 여러 명이 추천할 수 있고, 반대로 1명은 여러 개의 글을 추천할 수 있음

1. Question 모델에 voter 필드 추가하기

```
class Question(models.Model):
    author = models.ForeignKey(User, on_delete=models.CASCADE)
    subject = models.CharField(max_length=200)
    content = models.TextField()
    create_date = models.DateTimeField()
    modify_date = models.DateTimeField(null=True, blank=True)
    voter = models.ManyToManyField(User)
```

# 추천 기능

- Question, Answer 모델 변경하기

```
class ManyToManyField(RelatedField):  
    """  
    Provide a many-to-many relation by using an intermediary model that  
    holds two ForeignKey fields pointed at the two sides of the relation.  
  
    Unless a ``through`` model was provided, ManyToManyField will use the  
    create_many_to_many_intermediary_model factory to automatically generate  
    the intermediary model.  
    """  
  
    # Field flags  
    many_to_many = True  
    many_to_one = False  
    one_to_many = False  
    one_to_one = False
```

# 추천 기능

## 2. Makemigrations 오류 해결하기

```
(mysite) C:\projects\pyweb>python manage.py makemigrations
```

```
SystemCheckError: System check identified some issues:
```

```
ERRORS:
```

```
pybo.Question.author: (fields.E304) Reverse accessor for 'pybo.Question.author' clashes with reverse accessor for 'pybo.Question.voter'.
```

```
    HINT: Add or change a related_name argument to the definition for 'pybo.Question.author' or 'pybo.Question.voter'.
```

```
pybo.Question.voter: (fields.E304) Reverse accessor for 'pybo.Question.voter' clashes with reverse accessor for 'pybo.Question.author'.
```

```
    HINT: Add or change a related_name argument to the definition for 'pybo.Question.voter' or 'pybo.Question.author'.
```

### 오류 내용

Question 모델에서 사용한 author와 voter 필드가 모두 User 모델을 참조하고 있는데, 데이터에 접근할 경우 author 필드를 기준으로 할지 voter 필드를 기준으로 할지 장고가 판단할 수 없다.

### 오류 해결

‘HINT’에 **related\_name** 옵션 추가로 해결

## 추천 기능

### 3. Question 모델에 related\_name 옵션 추가하기

```
class Question(models.Model):
    author = models.ForeignKey(User, on_delete=models.CASCADE,
                               related_name='author_question')
    subject = models.CharField(max_length=200)
    content = models.TextField()
    create_date = models.DateTimeField()
    modify_date = models.DateTimeField(null=True, blank=True)
    voter = models.ManyToManyField(User, related_name='voter_question')
```

## 추천 기능

### 4. Answer 모델에 voter 필드 추가하기

```
class Answer(models.Model):
    author = models.ForeignKey(User, on_delete=models.CASCADE,
                               related_name='author_answer')
    question = models.ForeignKey(Question, on_delete=models.CASCADE)
    content = models.TextField()
    create_date = models.DateTimeField()
    modify_date = models.DateTimeField(null=True, blank=True)
    voter = models.ManyToManyField(User, related_name='voter_answer')
```

## 추천 기능

### 5. makemigrations, migrate 명령 실행하기

```
(mysite) C:\projects\pyweb>python manage.py makemigrations
Migrations for 'pybo':
  pybo\migrations\0006_auto_20210723_0327.py
    - Add field voter to answer
    - Add field voter to question
    - Alter field author on answer
    - Alter field author on question

(mysite) C:\projects\pyweb>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, pybo, sessions
Running migrations:
  Applying pybo.0006_auto_20210723_0327... OK
```



# 질문 추천 기능 만들기

## ➤ 질문 추천 버튼 만들기 – board/detail.html

```
<div class="container my-3">
  <h2 class="border-bottom py-2">{{ question.subject }}</h2>
  <div class="row my-3">
    <div class="col-1"> <!-- 추천 영역 -->
      <div class="bg-light text-center p-3 border font-weight-bolder mb-1">
        {{ question.voter.count }}
      </div>
      <a href="#" data-uri="{% url 'board:vote_question' question.id %}"
        class="recommend btn btn-sm btn-secondary btn-block my-1">추천</a>
    </div>
    <div class="col-11"> <!-- 질문 영역 -->
      <div class="card">
        <div class="card-body">
          (...생략...)
```

# 질문 추천 기능 만들기

## ➤ 추천 버튼 확인 창 만들기

```
{% block script %}  
<script type="text/javascript">  
$(document).ready(function(){  
    //삭제 처리  
    (...생략...)  
    //추천 처리  
    $(".recommend").on('click', function(){  
        if(confirm("정말로 추천하시겠습니까?")){  
            location.href=$(this).data('uri');  
        }  
    });  
});  
</script>
```

## 질문 추천 기능 만들기

### ➤ 질문 추천 URL 매핑 추가하기

```
urlpatterns = [  
    ....  
    path('vote/question/<int:question_id>/', views.vote_question,  
        name='vote_question'),  
]
```

# 질문 추천 기능 만들기

## ➤ 질문 추천 함수 추가하기

```
from django.contrib.auth.decorators import login_required
from django.contrib import messages
```

```
@login_required(login_url='common:login')
def vote_question(request, question_id):
    question = get_object_or_404(Question, pk=question_id)
    if request.user == question.author:
        messages.error(request, '본인이 작성한 글은 추천할 수 없습니다.')
    else:
        question.voter.add(request.user)
    return redirect('pybo:detail', question_id=question.id)
```

# 자신의 글 추천시 오류 표시

## 5. 자신이 쓴 글 추천시 오류 표시하기

```
{% extends 'base.html' %}
{% block content %}
<div class="container my-3">
  <!-- 추천 오류 표시 -->
  {% if messages %}
    {% for message in messages %}
      <div class="alert alert-danger my-3">
        <strong>{{ message.tags }}</strong>
        <ul>
          <li>{{ message.message }}</li>
        </ul>
      </div>
    {% endfor %}
  {% endif %}
  <h2 class="border-bottom py-2">{{ question.subject }}</h2>
  <div class="row my-3">
    <!-- 추천 영역 -->
```

### error

- 본인이 작성한 글은 추천할 수 없습니다.

## 장고 프레임워크로 만든 사이트

1

추천

장고 프레임워크로 만든 유명 사이트가 있는지 알려주세요.

수정

삭제

1개의 답변이 있습니다.

# 질문 목록에 추천수 표시하기

번호	추천	제목	글쓴이	작성일시
32	1	아이스커피가 좋나요? 1	giyong	2021년 7월 11일 5:57 오전
31	1	블랙 커피가 건강에 좋아요? 2	giyong	2021년 7월 11일 5:57 오전
30	1	모닝 커피로 아침을 기분 좋게 1	giyong	2021년 7월 11일 5:57 오전

```
<tbody>
  {% if question_list %}
  {% for question in question_list %}
  <tr class="text-center">
    <td>{{ question_list.paginator.count|sub:question_list.
    <td>
      {% if question.voter.all.count > 0 %}
      <span class="badge badge-warning px-2 py-1">
        {{ question.voter.all.count }}
      </span>
      {% endif %}
    </td>
    <td class="text-left">
      <a href="{% url 'pybo:detail' question.id %}">
        {{ question.subject }}
```