

11장. 댓글 기능



댓글 기능

- 댓글 추가 링크 화면 구현

일본 올림픽 일정

0

추천

일본 올림픽 일정을 알려주세요

수정

삭제

[댓글 추가..](#)

질문 댓글 등록하기

- Comment 모델 추가하기

```
class Comment(models.Model):  
    author = models.ForeignKey(User, on_delete=models.CASCADE)  
    content = models.TextField()  
    create_date = models.DateTimeField()  
    modify_date = models.DateTimeField(null=True, blank=True)  
    question = models.ForeignKey(Question, null=True, blank=True,  
                                on_delete=models.CASCADE)  
    answer = models.ForeignKey(Answer, null=True, blank=True,  
                              on_delete=models.CASCADE)
```

- Makemigrations, migrate 명령 실행하기

질문 댓글 등록하기

- 상세 화면에
질문 댓글 목록과
질문 댓글 등록 링크

```
<!-- 질문 댓글 Start-->
{% if question.comment_set.count > 0 %}
<div class="mt-3">
    {% for comment in question.comment_set.all %}
    <div class="comment py-2 text-muted">
        <span style="...">
            {{ comment.content }}
        </span>
        <span>
            - {{ comment.author }}, {{ comment.create_date }}
            {% if comment.modify_date %}
            (수정:{{ comment.modify_date }})
            {% endif %}
        </span>
    </div>
    {% endfor %}
</div>
{% endif %}
<div>
    <a href="{% url 'pybo:comment_create_question' question.id %}" class="small">
        <small>댓글 추가 ..</small>
    </a>
</div>
```

수정, 삭제 버튼 아래 위치

질문 댓글 등록하기

- 질문 댓글 URL 매핑 추가하기

pybo/urls.py

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

질문 댓글 등록하기

- 질문 댓글 폼 작성하기

forms.py

```
class CommentForm(forms.ModelForm):  
    class Meta:  
        model = Comment  
        fields = ['content']  
        labels = {  
            'content': '댓글내용'  
        }
```

질문 댓글 등록하기

- 질문 댓글 등록 함수 추가하기

```
@login_required(login_url='common:login')
def comment_create_question(request, question_id):
    # 질문 댓글
    question = get_object_or_404(Question, pk=question_id)
    if request.method == 'POST':
        form = CommentForm(request.POST)
        if form.is_valid():
            comment = form.save(commit=False)
            comment.author = request.user
            comment.create_date = timezone.now()
            comment.question = question
            comment.save()
            return redirect('pybo:detail', question_id=question.id)
        else:
            form = CommentForm()
    context = {'form': form}
    return render(request, 'pybo/comment_form.html', context)
```

질문 댓글 등록하기

- Comment 클래스 CSS 작성하기

base.html

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="{% static 'pybo/bootstrap.css' %}">
  <link rel="stylesheet" href="{% static 'pybo/style.css' %}">
  <title>Hello~ Pybo</title>
</head>
```

style.css

```
.comment{
  border-top:dotted 1px #ddd;
  font-size:0.7em
}
```


질문 댓글 등록하기

- 질문 댓글 등록을 위한 템플릿 작성하기

comment_form.html

```
{% extends 'base.html' %}
{% block content %}
<div class="container my-3">
  <h5 class="border-bottom pb-2">댓글 등록하기</h5>
  <form method="post" class="post-form my-3">
    {% csrf_token %}
    {% include 'form_errors.html' %}
    <div class="form-group">
      <label for="content">댓글 내용</label>
      <textarea class="form-control" name="content" id="content"
        rows="3">{{ form.content.value|default_if_none:'' }}</textarea>
    </div>
    <button type="submit" class="btn btn-primary">저장하기</button>
  </form>
</div>
{% endblock %}
```

질문 댓글 수정 / 삭제

- 댓글 수정 / 삭제 화면 구현

PyWeb

today(로그아웃)

질문/답변

나의 프로필

댓글 등록하기

댓글 내용

1년 연기됨
원래는 2020년인데 코로나로 연기

저장하기

질문/답변

나의 프로필

127.0.0.1:8000 내용:
정말로 삭제하시겠습니까?

확인

취소

일본 올

0

추천

일본 올림픽 일정을 알려주세요

수정

삭제

1년 연기됨
원래는 2020년인데 코로나로 연기
- today, 2021년 7월 26일 5:43 오전 (수정:2021년 7월 26일 5:47 오전) [수정](#) [삭제](#)
[댓글 추가..](#)

질문 댓글 수정/삭제 기능

- 상세 화면에 댓글 수정과 삭제 링크 추가

```
<span>
  - {{ comment.author }}, {{ comment.create_date }}
  {% if comment.modify_date %}
    (수정:{{ comment.modify_date }})
  {% endif %}
</span>
{% if request.user == comment.author %}
<a href="{% url 'pybo:comment_modify_question' comment.id %}" class="small">수정</a>
<a href="#" class="small delete"
  data-uri="{% url 'pybo:comment_delete_question' comment.id %}">삭제</a>
{% endif %}
```


Jquery로 삭제!!

질문 댓글 수정/삭제

- 질문 댓글 수정/삭제 URL 매핑 추가하기

```
urlpatterns = [  
    ....  
    path('comment/modify/question/<int:comment_id>/', views. comment_modify_question,  
        name='comment_modify_question'),  
  
    path('comment/delete/question/<int:comment_id>/', views. comment_delete_question,  
        name='comment_delete_question'),  
]
```

주의 요함!!



질문 댓글 수정/삭제

- 질문 댓글 수정 함수 추가하기

주의 요함!!

```
@login_required(login_url='common:login')
def comment_modify_question(request, comment_id):
    # 질문 댓글 수정
    comment = get_object_or_404(Comment, pk=comment_id)
    if request.method == "POST":
        form = CommentForm(request.POST, instance=comment)
        if form.is_valid():
            comment = form.save(commit=False)
            comment.author = request.user
            comment.modify_date = timezone.now()
            comment.save()
            return redirect('pybo:detail', question_id=comment.question.id)
        else:
            form = CommentForm(instance=comment)
    context = {'form': form}
    return render(request, 'pybo/comment_form.html', context)
```

질문 댓글 수정/삭제

- 질문 댓글 삭제 함수 추가하기

```
@login_required(login_url='common:login')
def comment_delete_question(request, comment_id):
    comment = get_object_or_404(Comment, pk=comment_id)
    comment.delete()
    return redirect('pybo:detail', question_id=comment.question.id)
```

답변 댓글 등록하기

- 답변 댓글 등록
상세 화면

```
<!-- 답변 댓글 Start-->
{% if answer.comment_set.count > 0 %}
<div class="mt-3">
    {% for comment in answer.comment_set.all %}
    <div class="comment py-2 text-muted">
        <span style="white-space: pre-line;">
            {{ comment.content }}
        </span>
        <span>
            - {{ comment.author }}, {{ comment.create_date }}
            {% if comment.modify_date %}
            (수정:{{ comment.modify_date }})
            {% endif %}
        </span>
    </div>
    {% endfor %}
</div>
{% endif %}
<div>
    <a href="{% url 'pybo:comment_create_answer' answer.id %}" class="small">
        <small>댓글 추가 ..</small>
    </a>
</div>
```

답변 댓글 등록하기

- **답변 댓글 URL 매핑 추가하기**

pybo/urls.py

```
urlpatterns = [  
    ....  
    path('comment/create/answer/<int:answer_id>/', views.comment_create_answer,  
         name='comment_create_answer'),  
]
```


답변 댓글 등록하기

- 답변 댓글 등록 함수 추가하기

```
@login_required(login_url='common:login')
def comment_create_answer(request, answer_id):
    # 답변 댓글 등록
    answer = get_object_or_404(Answer, pk=answer_id)
    if request.method == "POST":
        form = CommentForm(request.POST)
        if form.is_valid():
            comment = form.save(commit=False)
            comment.author = request.user
            comment.create_date = timezone.now()
            comment.answer = answer
            comment.save()
            return redirect('pybo:detail', question_id=answer.question.id)
    else:
        form = CommentForm()
    context = {'form': form}
    return render(request, 'pybo/comment_form.html', context)
```

답변 댓글 수정 / 삭제

● 답변 수정 / 삭제 화면 구현

PyWeb

today(로그아웃)

질문/답변

나의 프로필

댓글 등록하기

댓글 내용

17일간이네요..
여름과 함께 지나가겠네

저장하기

1개의 답변이 있습니다.

0

추천

7월 23일 ~ 8월 8일 17일간

modified at
2021년 7월 26일

수정 삭제

17일간이네요..
여름과 함께 지나가겠네
- today, 2021년 7월 26일 5:45 오전 (수정:2021년 7월 26일 5:55 오전) [수정](#) [삭제](#)

댓글 추가 ..

17일간이네요..
원래는 2020년인데 코로나로 연기
- today, 2021년 7월 26일 5:43 오전 (수정:2021년 7월 26일 5:45 오전) [수정](#) [삭제](#)

댓글 추가 ..

127.0.0.1:8000 내용:
정말로 삭제하시겠습니까?

확인

취소

답변 댓글 수정/삭제 기능

- 상세 화면에 답변 댓글 수정과 삭제 링크 추가

```
<span>
  - {{ comment.author }}, {{ comment.create_date }}
  {% if comment.modify_date %}
    (수정:{{ comment.modify_date }})
  {% endif %}
</span>
{% if request.user == comment.author %}
  <a href="{% url 'pybo:comment_modify_answer' comment.id %}" class="small">수정</a>
  <a href="#" class="small delete"
    data-uri="{% url 'pybo:comment_delete_answer' comment.id %}">삭제</a>
{% endif %}
```

답변 댓글 수정/삭제

- 답변 댓글 수정/삭제 URL 매핑 추가하기

```
urlpatterns = [  
    ...  
    path('comment/modify/answer/<int:comment_id>/', views. comment_modify_answer,  
        name='comment_modify_answer'),  
  
    path('comment/delete/answer/<int:comment_id>/', views. comment_delete_answer,  
        name='comment_delete_answer'),  
]
```

답변 댓글 수정/삭제

- 답변 댓글 수정 함수 추가하기

```
@login_required(login_url='common:login')
def comment_modify_answer(request, comment_id):
    # 답변 댓글 수정
    comment = get_object_or_404(Comment, pk=comment_id)
    if request.method == "POST":
        form = CommentForm(request.POST, instance=comment)
        if form.is_valid():
            comment = form.save(commit=False)
            comment.author = request.user
            comment.modify_date = timezone.now()
            comment.save()
            return redirect('pybo:detail', question_id=comment.answer.question.id)
    else:
        form = CommentForm(instance=comment)
    context = {'form': form}
    return render(request, 'pybo/comment_form.html', context)
```

답변 댓글 수정/삭제

- 답변 댓글 삭제 함수 추가하기

```
@login_required(login_url='common:login')
def comment_delete_answer(request, comment_id):
    # 답변 댓글 삭제
    comment = get_object_or_404(Comment, pk=comment_id)
    comment.delete()
    return redirect('pybo:detail', question_id=comment.answer.question.id)
```