



CRUD

create

데이터를 생성

update

데이터를 수정

read

데이터를 읽기

delete

데이터를 지우기

새 글 작성하기

new form

새로운 글을 입력받는 창을 만들어봅시다.

데이터 전송 방식

GET

POST



GET

보내는 데이터가 url을 통해 노출된다.(보안에 취약)
그렇기에 많은 양의 데이터를 보내기에 적합하지 않음.



 <https://www.google.com/search?q=이두희>



POST

url을 통해 데이터가 노출되지 않는다.
ex. 로그인을 할 때 비밀번호가 url에 표시되는 일이 없다.

```
<form action="{% url 'new' %}" method="POST">
    {% csrf_token %}
    title : <input type="text">
    content : <textarea name="" id="" cols="30" rows="10"></textarea>
</form>
```

csrf token?

- 사이트 간 요청 위조(CSRF)를 막기 위해 넣는 토큰



views.py

views.py를 통해서 객체를 생성하는 것을 해봅시다!

방법1

new 액션과 create 액션 만들기

방법2

**new액션 하나만 작성하여
method 방식에 따라 처리를 다르게 하기**

```
def new(request):  
    return render(request, 'blog/new.html')  
  
# new.html submit  
def create(request):  
    blog = Blog()  
    blog.title = request.GET['title']  
    blog.body = request.GET['body']  
    blog.pub_date = timezone.datetime.now()  
    blog.save()  
    return redirect('/blog/' + str(blog.id))
```



delete

views.py를 통해서 객체를 삭제해봅시다!

views.py에서 객체를 불러온 뒤 삭제를 합니다.

```
def delete(request, post_id):  
    delete_post=Post.objects.get(id=post_id)  
    delete_post.delete()  
    return redirect('home')
```



update

views.py를 통해서 객체를 수정해봅시다!

```
def edit(request, post_id):
    edit_post = Post.objects.get(id=post_id)
    return render(request, 'edit.html', {'post':edit_post})

def update(request, post_id):
    update_post = Post.objects.get(id=post_id)
    update_post.title = request.POST['title']
    update_post.writer = request.POST['writer']
    update_post.body = request.POST['body']
    update_post.save()
    return redirect('home')
```


실습!

하다보면 view에서 데이터를 처리하는 것은 오히려 익숙할 것 같은데,
urls.py, views.py, templates의 데이터를 주고받는 과정이 정말 헷갈립니다.
해커톤 전에 과제를 한 번 해보면서 블로그 작성을 진행하는 것을 추천드립니다!

app 이름 : blogapp

models.py

- class명 Blog

- 필요한 컬럼

 - title : 제목

 - writer : 작성자

 - body : 내용

views.py

- 필요한 함수
- home : 블로그 글 목록을 보여줍니다. 제목을 클릭하면 상세 페이지로 갈 수 있습니다.
- detail : 각각의 블로그 글의 제목, 내용, 작성자를 표시해줍니다.
수정 버튼, 삭제 버튼이 있습니다.
- new : 새로운 글을 받는 form이 있는 html을 띄웁니다.
- create : 새로운 데이터를 모델에 작성해줍니다.
- edit : 수정 페이지가 나타납니다.
- update : 수정된 데이터를 모델에 반영합니다.
이후 home으로 redirect 합니다.
- delete : 해당 데이터를 삭제한 뒤 home으로 redirect 합니다.

Thank u ;-)

likelion_kmu_7



이철희