

<DongriGo 프로젝트 오류 공유 보고서>

오류공유 보고서

제목	admin 페이지에서 post 수정 화면 진입 시 500(TypeError)
대상	admin 페이지의 post 수정
발행일자	2025/12/22
작성자	김동훈

오류 내용

TypeError at /admin/blog/post/23/change/

args or kwargs must be provided.

Request Method: GET

Request URL: http://127.0.0.1:8000/admin/blog/post/23/change/

Django Version: 6.0

Exception Type: TypeError

Exception Value: args or kwargs must be provided.

Exception Location: C:\Users\dhkim\anaconda3\envs\travel_django\Lib\site-packages\django\utils\html.py, line 137, in format_html

Raised during: django.contrib.admin.options.change_view

Python Executable: C:\Users\dhkim\anaconda3\envs\travel_django\python.exe

Python Version: 3.12.12

Python Path: ['C:\Users\dhkim\travel_site', 'C:\Users\dhkim\anaconda3\envs\travel_django\python312.zip', 'C:\Users\dhkim\anaconda3\envs\travel_django\DLLs', 'C:\Users\dhkim\anaconda3\envs\travel_django\Lib', 'C:\Users\dhkim\anaconda3\envs\travel_django', 'C:\Users\dhkim\anaconda3\envs\travel_django\Lib\site-packages']

Server time: Mon, 22 Dec 2025 19:34:18 +0900

Error during template rendering

In template C:\Users\dhkim\anaconda3\envs\travel_django\Lib\site-packages\django\contrib\admin\templates\admin\fieldset.html, error at line 23

args or kwargs must be provided.

admin 페이지, post에서 기존에 업로드 된 게시물을 수정하기 위해 해당 게시물을 클릭하면 수정 페이지로 넘어가지 않고 에러 발생

에러 메시지 요약

- Exception Type: TypeError
- Exception Value: args or kwargs must be provided
- Raised during: django.contrib.admin.options.change_view
- Template rendering: admin/includes/fieldset.html 라인 근처에서 렌더 중 예외 발생

원인 요약

Django admin의 커스텀 readonly field(rendered_preview)에서 format_html()을 인자 없이 호출함.

Django 6 기준 format_html()은 반드시 args 또는 kwargs로 값을 주입하는 방식으로 사용되어야 하며, format_html("<div>...</div>")처럼 문자열만 단독으로 호출하면 TypeError 발생.

```
def rendered_preview(self, obj: Post):
    html = obj.rendered_content()
    return format_html(
        '<div style="...">{}</div>',
        format_html(html) # args/kwargs 없는 format_html 호출 → Django 6에서 TypeError
    )
```

시정조치

rendered_preview에서 내부 HTML을 다시 format_html(html)로 감싸지 않고, 이미 서버 측에서 sanitize된 HTML(rendered_content())이라는 전제 하에 mark_safe(html)로 안전한 HTML 문자열임을 명시한 뒤 바깥 wrapper는 format_html()로 감싸되 주입 인자를 사용하도록 수정.

세부내용

수정 후 코드

```
from django.utils.html import format_html
from django.utils.safestring import mark_safe

def rendered_preview(self, obj: Post):
    if not obj or not obj.pk:
        return "저장 후 미리보기가 표시됩니다."

    html = obj.rendered_content() # bleach 등으로 정제된 HTML
    return format_html(
        '<div style="max-height:280px;overflow:auto;border:1px solid #ddd;padding:10px;border-radius:8px;">{}</div>',
        mark_safe(html),
    )
```

분석 및 근거

- Django admin은 `readonly_fields` / `fieldsets`에 포함된 커스텀 메서드를 템플릿 렌더링 과정에서 호출
- 해당 메서드 내부에서 `format_html()` 사용 방식이 Django 6 규칙과 충돌하여, 렌더링이 즉시 중단됨.
- 동일 패턴은 `format_html(f'...')`처럼 f-string으로 이미 합쳐진 HTML을 넘겨도 같은 위험이 있음
- 규칙: `format_html()`은 템플릿 문자열 + 값 주입 인자 패턴을 지켜야 안전하고 버전 영향이 적음

재발 방지

- `format_html()` 사용 규칙 표준화:
 1. `format_html("<tag>{}</tag>", value)` 형태로만 사용
 2. 단순히 `safe` 처리 목적이라면 `mark_safe()` 또는 `format_html({}, mark_safe(html))` 사용
 3. f-string으로 만들어진 HTML을 `format_html(f'...')`로 감싸지 않기
- 코드 리뷰 체크리스트에 추가: admin.py에서 `format_html()` 호출 시, 인자 주입이 있는지 확인
- 필요시 정적 검색 명령(windows/powershell):
`Select-String -Path .\apps\blog\admin.py -Pattern "format_html\(" -Context 0,2`

