

- 반드시 row col에 관하여.

1. row

- 1) row는 한 줄을 만들 때 쓰는 클래스
- 2) 이 안에 여러개의 열을 배치할 수 있다
- 3) 반드시 row 안에 col 클래스가 들어야 한다.

2. col

- 1) 열의 배치를 정하는 클래스
- 2) 한 row은 col-1 ~ col-2 까지 12칸으로 나뉘어 있다.

ex) `<div class="row">`

`<div class="col-1">` 추천아이템 `</div>` → 좁은 화면

`<div class="col-11">` 추천 질문 내용 `</div>` → 넓은 11칸
`</div>`

- 앵커기능으로 스크롤 초기화 문제 해결.

- 1) 질문 상세 페이지에 앵커 엘리먼트 추가하기.

↳ `` ``

- 2) redirect 수정

↳ 기존 코드

```
return redirect('pybo:detail', question_id=question.id)
```

↳ 적용코드

```
return redirect('{}/#answer-{}'.format(
    resolve_url('pybo:detail', question_id=question.id), answer.id))
```

→ #뒤에 있는 값을 프미기도 식별자로 구분한다.

- 검색 기능

from django.db.models import Q → 복잡한 쿼리(조건들)을 작성할 때 사용.
여러 조건을 OR, AND 등으로 연결

kw = request.GET.get('kw', '') # 검색어

↳ 사용자가 /page?kw=검색어 처럼 입력했을 때, kw 파라미터 값을 가져온다.

if kw: → 검색어가 있다면

question_list = question_list.filter(

Q(subject__icontains=kw) ; # 제목 검색

Q(content__icontains=kw) ; # 내용 검색

Q(author__username__icontains=kw) ; # 질문 글쓴이 검색

Q(answer__author__username__icontains=kw) # 답글 글쓴이 검색

).distinct() → 중복 결과 제거

- Shell을 통해 답변을 여러개 저장하려고 했을 때, 오류 발생

↳ Test를 위해 질문 '답변 페이징 Test' 질문 생성한 후 답변을 50개 생성하려고 했으나 오류 발생.

↳ 오류 원인: Answer 모델의 question 필드가 ForeignKey 인데, filter의 결과인 queryset 전체를 넣으려고 해서 문제 발생.

↳ 해결방법: first를 사용하거나, get으로 question 값을 받아야 한다.

혹은 author 값을 넣어주어야지 오류 발생하지 않음.

↳ Test 방법.

1-1) 기존 User 가져오기.

from django.contrib.auth.models import User

user = User.objects.first() # DB에 저장된 첫번째 사용자 가져오기.

1-2) 새로운 User 객체 만들기.

from django.contrib.auth.models import User

user = User.objects.create_user(username='testuser', password='')

2) Shell에서 답변 여러개 생성하기.

```
from pybo.models import Question, Answer
```

```
from django.utils import timezone
```

```
from django.contrib.auth.models import User
```

```
q = Question.objects.filter(subject__contains='답변 피어싱').first()
```

```
user = 유저 지정
```

```
answers = []
```

```
for i in range(10):
```

```
    a = Answer(
```

```
        author = user,
```

```
        question = q,
```

```
        content = f'Test 답글 내용 {i+1}',
```

```
        create_date = timezone.now())
```

```
)
```

```
answers.append(a)
```

```
Answer.objects.bulk_create(answers)
```

→ 여러 객체를 리스트로 만들어서 한번에 SQL-INSERT 쿼리로 DB에 저장.