

Debugging



| Background

- ✓ Django

| Goal

- ✓ 웹 페이지를 구현 할 수 있다.
- ✓ 에러 상황을 수정 할 수 있다.

| 환경 설정

- 1) 개발환경 : python 3.8+, Django 3.2.X, vscode
- 2) 프로젝트 다운로드 : 프로젝트 압축파일을 다운로드 후 압축해제한다.
- 3) 코드 작성시 작성한 코드에 대한 해설을 주석 형태로 상세하게 작성한다.
- 4) 제출 : venv 폴더를 제외한 프로젝트 폴더를 압축하고 지역0반_홍길동 형식으로 이름을 변경하여 제출한다.
- 5) 제시된 requirements.txt 이외의 패키지 설치는 불가능합니다. (ex. django-seed, django-extensions)

| 주의사항

- 1) python manage.py migrate 명령어를 이용하여 DB를 생성합니다.
- 2) 테스트를 위한 계정은 python manage.py createsuperuser 명령어를 이용하여 생성합니다.
- 3) 주어진 내용을 제외한 어떠한 문서나 소스코드도 참고 할 수 없습니다.
- 4) 모든 문항을 해결할 시 총 점수는 90점 입니다.
- 5) 제출 시간에 따라 배점 10점이 차등 부여됩니다.
- 6) 각 문항은 개별 문항으로 취급합니다. (이전 문항을 해결하지 않아도 다음 문항으로 넘어갈 수 있습니다.)
- 7) views.py에 작성 되어있는 데코레이터 수정을 금지합니다.
- 8) 문제에 명시된 파일이 있는 경우, 반드시 해당 파일을 수정합니다.

소스코드 유사도 판단 프로그램 기준 코드가 유사하여 부정 행위로 판단 시, 0점 처리 및 학사 기준에 의거 조치 실시 예정

Debugging



| 문제1. accounts signup (5)

다음의 상황을 해결합니다.

1. `/accounts/signup/` 경로로 이동하여 회원가입을 시도합니다.
2. 올바른 정보를 입력하는 경우 회원가입을 성공합니다.
3. 비밀번호와 비밀번호 확인을 **다르게 입력한 경우**, 오류 메시지를 포함한 **회원가입 페이지가 반환 되어야** 하지만 아래의 이미지와 같은 에러가 발생합니다.
4. 비밀번호와 비밀번호 확인을 다르게 입력한 경우 오류 메시지와 함께 다시 회원가입 페이지가 반환 될 수 있도록 **accounts/views.py** 를 수정합니다.

ValueError at /accounts/signup/

The view `accounts.views.signup` didn't return an `HttpResponse` object. It returned `None` instead.

Request Method: POST

Request URL: `http://127.0.0.1:8000/accounts/signup/`

Django Version: 3.1.12

Exception Type: ValueError

Exception Value: The view `accounts.views.signup` didn't return an `HttpResponse` object. It returned `None` instead.

Debugging



| 문제2. accounts profile (5)

다음의 상황을 해결합니다.

1. 로그인을 진행합니다.
2. 내 프로필 링크를 클릭하여 프로필 페이지로 이동하면, 프로필 정보가 보이지 않습니다.
3. 아래와 같이 올바르게 프로필 내용이 보일 수 있도록

accounts/templates/accounts/profile.html 템플릿을 수정합니다.

Hello, ssafy

[내 프로필](#)

Logout

[회원정보수정](#)

회원탈퇴

ssafy님의 프로필

팔로워 : 0 / 팔로우 : 0

ssafy이 작성한 게시글

ssafy이 작성한 댓글

ssafy이 좋아요를 누른 게시글

[\[back\]](#)

Debugging



| 문제3. accounts update (5)

다음의 상황을 해결합니다.

1. 로그인을 진행합니다.
2. 회원정보수정 링크를 클릭하거나 경로(/accounts/update/)로 직접 이동합니다.
3. 아래 그림처럼 모든 정보가 아닌, **이메일 주소**만 변경할 수 있도록 **accounts/forms.py** 를 수정합니다.

Hello, ssafy

[내 프로필](#)
[Logout](#)
[회원정보수정](#)
[회원탈퇴](#)

회원정보수정

비밀번호:

알고리즘: pbkdf2_sha256 반복: 216000 솔트: lyLWff***** 해시: UbnEs+*****

원본 비밀번호는 저장되지 않으므로, 해당 사용자의 비밀번호를 확인할 수 없습니다. 다만 이 [풀](#)을 사용하여 비밀번호를 변경할 수 있습니다.

마지막 로그인: 2022-04-28 19:45:03

최상위 사용자 권한: ☐ 해당 사용자에게 모든 권한을 허가합니다.

그룹:

↑

accounts | 사용자 | Can add user
accounts | 사용자 | Can change user
accounts | 사용자 | Can delete user
accounts | 사용자 | Can view user

이 사용자가 속한 그룹. 사용자는 그룹에 부여된 모든 권한을 물려 받습니다.

사용자 권한: accounts | 사용자 | Can view user 이 사용자를 위한 특정 권한.

사용자 이름: ssafy 150자 이하 문자, 숫자 그리고 @/./+/_만 가능합니다.



Hello, ssafy

[내 프로필](#)
[Logout](#)
[회원정보수정](#)
[회원탈퇴](#)

회원정보수정

이메일 주소:

비밀번호:

비밀번호가 설정되지 않습니다.

원본 비밀번호는 저장되지 않으므로, 해당 사용자의 비밀번호를 확인할 수 없습니다. 다만 이 [풀](#)을 사용하여 비밀번호를 변경할 수 있습니다.

[제출](#)

[back](#)

Debugging



| 문제4. articles list (20)

다음의 상황을 해결합니다.

1. 로그인 후 게시글 작성 페이지(</articles/create/>)에서 새로운 글을 작성합니다.
2. 전체 게시글 페이지(</articles/>)로 이동합니다.
3. 아래와 같이 게시글 전체 목록이 화면에 보일 수 있도록 [articles/templates/articles/index.html](/articles/templates/articles/index.html) 템플릿을 수정합니다.

Hello, ssafy

[내 프로필](#)

[Logout](#)

[회원정보수정](#)

[회원탈퇴](#)

Articles

[CREATE](#)

작성자: [ssafy](#)

글 번호: 5

글 제목: Where can I get some?

글 내용: There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All the Lorem Ipsum generators on the Internet tend to repeat predefined chunks as necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful of model sentence structures, to generate Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is therefore always free from repetition, injected humour, or non-characteristic words etc.

[좋아요](#)

[DETAIL](#)

작성자: [ssafy](#)

글 번호: 4

글 제목: Where does it come from?

글 내용: Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1.10.32.

Debugging



| 문제5. articles delete (5)

다음의 상황을 해결합니다.

1. 게시글 상세보기 페이지(/articles/:pk/)로 이동합니다.
2. 게시물의 삭제 버튼을 클릭하면 정상적으로 삭제되도록 **articles/views.py** 와 **articles/templates/articles/detail.html** 템플릿을 수정합니다.

| 문제6. comments list (15)

다음의 상황을 해결합니다.

1. 게시글 상세 페이지로 이동합니다.
2. 댓글 목록이 화면에 보일 수 있도록 **articles/views.py** 혹은 **articles/templates/articles/detail.html** 템플릿을 수정합니다. (2개의 파일 모두 수정 가능)

Debugging



| 문제 7. articles like (5)

다음의 에러 상황을 해결합니다.

1. 로그인 후 </articles/> 경로로 이동합니다.
2. 해당 페이지에서 [좋아요](#) 버튼을 클릭했을 때 아래의 이미지와 같은 에러가 발생합니다.
3. 좋아요 기능이 올바르게 작동할 수 있도록 [articles/views.py](#) 를 수정합니다.

AttributeError at /articles/6/likes/

'User' object has no attribute 'like_articles'

Request Method: POST

Request URL: http://127.0.0.1:8000/articles/6/likes/

Django Version: 3.1.7

Exception Type: AttributeError

Exception Value: 'User' object has no attribute 'like_articles'

Debugging



8번 ~ 10번은 서술형 문제입니다. 제공된 서술형.md 파일에 답안을 작성 바랍니다.

| 문제8. 서술형 (10)

전체 게시글 페이지(/articles/) 에서 **좋아요**를 클릭 했을 때, 페이지가 새로 고침 되는 현상이 발생한다. 이러한 현상이 나타나는 이유와 새로 고침 되지 않도록 할 수 있는 방법을 서술하시오.

(프로젝트 파일은 수정하지 않고 마크다운 파일에 답변만 작성합니다.

50자 이내로 작성하며, 부분점수 인정됩니다.)

| 문제9. 서술형 (10)

team 앱의 models.py 에 정의한 Team 모델을 마이그레이션 하기 위해 **makemigrations** 명령어를 작성했습니다.

이때, 아래의 이미지와 같이 변경사항이 발생하지 않은 이유 두 가지를 모두 서술하시오.

(프로젝트 파일은 수정하지 않고 마크다운 파일에 답변만 작성합니다.

50자 이내로 작성하며, 부분점수 인정됩니다.)

```
$ python manage.py makemigrations
No changes detected
```

| 문제10. 서술형 (10)

현재 프로젝트에서는 회원 가입 기능을 구현하기 위하여 **accounts** 앱의 **views.py** 에서 **CustomUserCreationForm** 을 사용하고 있습니다.

이때, 장고에서 제공되는 **UserCreationForm**(from django.contrib.auth.forms import UserCreationForm)을 바로 사용하지 않고, **CustomUserCreationForm**(from .forms import CustomUserCreationForm)을 사용한 이유를 서술하시오.

(프로젝트 파일은 수정하지 않고 마크다운 파일에 답변만 작성합니다.

50자 이내로 작성하며, 부분점수 인정됩니다.)