

نمره ۳۵

مهلت ارسال: تا ۳۰ روز پس از اتمام دوره

فاز ۴ پروژه پایانی

هدف از این تمرین: تسلط در تمپلیت، فرم‌ها و ویوها | Django 3

شرح تمرین:

پیشنیاز این تمرین، تمرین شماره ۱ است. شما میبایست تمرین شماره ۱ را انجام داده باشید. در این تمرین تسلط شما در تمپلیت‌ها، فرم‌ها و ارتباط تمپلیت با ویو و مدل‌ها محک زده می‌شود. در این تمرین از شما می‌خواهیم صفحاتی را طراحی کنید که در ادامه به جزئیات آن می‌پردازیم.

* جهت درک بهتر و مشاهده خروجی موردنظر، از هر مدل در دیتابیس حداقل ۵ عنصر ایجاد نمایید.

* تمامی صفحات میبایست از شکل ظاهری مناسب و مدرن طراحی شده باشند:

- میتوانید از کتابخانه Bootstrap و امثال آن و یا از قالب‌های آماده استفاده کنید.
- در صورت استفاده از قالب‌های آماده و ظاهر شکیل، نمره‌ی مثبت برای شما تعلق می‌گیرد.

* شما میتوانید از پکیج‌های زیر به جهت بهبود کیفیت فرم‌های خود استفاده کنید:

- ✓ Django widget tweaks
- ✓ Django crispy forms

صفحه پروفایل

در این صفحه کاربر میتواند مشخصات فردی خود را مشاهده کند.

(دقت کنید هر کاربر میتواند فقط مشخصات خودش را ببیند و نباید بتواند مشخصات فرد دیگری را ببیند)

ویژگی‌های زیر میبایست به کاربر نمایش داده شود:

- نام کاربری
- نام
- نام خانوادگی
- ایمیل
- سن
- کدملی
- شماره موبایل
- معرفی (bio)
- جنسیت
- لینک دانلود فایل رزومه
- تاریخ عضویت (نمونه صحیح: ۱۶:۰۰ - ۱۴۰۰/۱۰/۱۰)
- در صورتیکه نویسنده بود نوشته شود شما در این سایت نویسنده هستید.

-- ترتیب فیلدها مهم نیست --

صفحه ویرایش پروفایل

در این صفحه کاربر میتواند مشخصات فردی خود را ویرایش کند.

لینک ویرایش را در صفحه پروفایل میتوانید قرار دهید.

ویژگی‌های زیر را کاربر میتواند ویرایش نماید:

- نام کاربری
- نام
- نام خانوادگی
- ایمیل
- سن
- کدملی
- شماره موبایل
- معرفی (bio)
- جنسیت
- آپلود فایل رزومه

* کاربر نمیتواند "کدملی" خود را ویرایش کند. این گزینه باید نمایش داده شود اما ویرایش آن غیرفعال باشد.

-- ترتیب فیلدها مهم نیست --

صفحه ثبت نام در سایت

در این صفحه کاربر میتواند حساب کاربری جدید در سایت ایجاد کند.

موارد زیر میبایست حین ثبت نام از کاربر گرفته شود:

- نام کاربری
- نام
- نام خانوادگی
- ایمیل
- سن
- کدملی
- شماره موبایل
- جنسیت
- آپلود فایل رزومه
- کلمه عبور
- تکرار کلمه عبور

موارد زیر میبایست کنترل شوند:

- نام کاربری انتخاب شده نباید از قبل در سایت موجود باشد.
- نام و نام خانوادگی میبایست فقط از حروف فارسی تشکیل شده باشند.
- ایمیل میبایست از فرمت صحیح برخوردار باشد.
- کلمه عبور میبایست حداقل از ۸ کاراکتر و حروف کوچک و بزرگ و اعداد تشکیل شده باشد.
- کلمه عبور با تکرار کلمه عبور میبایست یکسان باشد.

* میتوانید به جهت کاهش پیچیدگی و رعایت اصل DRY از کلاس `UserCreationForm` ارث بری کنید.

پس از ثبت نام موفق، کاربر را در سایت لاگین و به صفحه پروفایل هدایت کنید و یک پیام به جهت "موفقیت آمیز بودن ثبت نام" به کاربر نمایش دهید.

صفحه ورود به سایت

در این صفحه کاربر میتواند با وارد کردن نام کاربری و رمز عبور خود وارد سایت شود.

اگر نام کاربری در سایت موجود نبود، خطای مناسب نمایش داده شود.

اگر نام کاربری در سایت موجود بود اما رمز عبور نادرست بود، خطای مناسب نمایش داده شود.

اگر نام کاربری و رمز عبور درست بود، کاربر به صفحه پروفایل خود هدایت شود و پیام خوش آمدید نمایش داده شود.

به نکات زیر توجه و آنها را پیاده سازی کنید:

- اگر کاربر در سایت لاگین نکرده باشد، در منوی سایت لینکهای ورود و ثبت نام نمایش داده شوند.
- اگر کاربر در سایت لاگین کرده باشد، در منوی سایت لینکهای ورود و ثبت نام حذف شوند و به جای آن لینکهای پروفایل و خارج شوید نمایش داده شود.
- اگر کاربر در سایت لاگین کرده باشد، صفحه ورود و ثبت نام نباید برای اون باز شود. (در صورت وارد کردن url این صفحات، کاربر باید به صفحه پروفایل خود هدایت شود)

صفحه پست‌ها

در این صفحه میبایست تمامی پست‌هایی که در وضعیت "در حال انتشار" هستند، در سایت نمایش داده شوند. موارد زیر از هر پست میبایست نمایش داده شود:

- عکس پست
 - عنوان پست
 - نام نویسندگان (میتوانید یک تابع در model بنویسید و بین نویسندگان for بزنید و نامشان را در یک رشته ذخیره کنید).
 - تاریخ ایجاد پست (نمونه صحیح: ۱۸:۰۰ - ۱۴۰۰/۱۰/۱۰)
 - تعداد کامنت‌ها
 - توضیحات خلاصه پست
 - لینک "ادامه مطلب" (لینک صفحه جزئیات هر پست میبایست باشد که در ادامه آنرا طراحی و پیاده سازی خواهید کرد)
- به جهت بهبود و افزایش سرعت لود سایت، اتریوت "loading="lazy" را برای هرکس قرار دهید تا حین بارگذاری سایت، کاربر معطل لود عکس‌ها نشود.

نمونه:

The screenshot shows a web interface with a header containing navigation links: "تیم فنی", "نقشه راه آینده", "تماس با ما", "دوره ها", "معرفی انجمن", "بلاگ", "خانه", and a logo for "انجمن علمی مهندسی کامپیوتر CEC KIAU". Below the header, there are three article cards. Each card has a title, a date, a brief description, and a "ادامه مطلب" (Read more) button. The first card is about Artificial Intelligence, the second is about programming for beginners, and the third is about the student day. Each card also features a small image related to the topic.

صفحه جزئیات پست‌ها

در این صفحه میبایست جزئیات پست مربوطه که از طریق slug یی که در url سایت وارد شده گرفته‌ایم نمایش داده شود. (در صورتیکه هیچ پستی با slug وارد شده در دیتابیس وجود نداشته باشد و یا پست مربوطه در حالت "انتشار در سایت" نباشد، باید صفحه ۴۰۴ نمایش داده شود.)

موارد زیر میبایست در این صفحه نمایش داده شود:

- عکس پست (سایز بزرگ، وسط صفحه و lazy load باشد)
- عنوان پست
- نام نویسندگان (میتوانید یک تابع در model بنویسید و بین نویسندگان for بزنید و نامشان را در یک رشته ذخیره کنید.)
- تاریخ ایجاد پست (نمونه صحیح: ۱۸:۰۰ - ۱۴۰۰/۱۰/۱۰)
- تعداد کامنت‌ها
- توضیحات کامل پست

موارد زیر اختیاری است:

- نمایش کامنت‌های پست مربوطه
- ثبت کامنت جدید برای پست

نمونه:



شما میتوانید از قالب‌های آماده زیر استفاده کنید.

اگر از ورژن خاصی از بوت استرپ استفاده میکنید، بطور مثال اگر از بوت استرپ ۴ استفاده میکنید، دقت کنید قالب شما نیز ورژن ۴ بوت استرپ باشد.

قالب‌های صفحه ورود و ثبت‌نام:

<https://freefrontend.com/bootstrap-login-forms>

<https://colorlib.com/wp/cat/login-forms>

یک قالب ساده و مناسب سایت وبلاگ (مناسب فاز ۴ پروژه):

<https://bootstrapmade.com/free-bootstrap-template-corporate-moderna>

سایر قالب‌های سایت:

<https://bootstrapmade.com>

<https://colorlib.com/wp/free-bootstrap-۴-website-templates>

https://themewagon.com/theme_tag/bootstrap-۴-templates

دستورات پر کاربرد Django:

```
django-admin startproject my_project
```

✱ ایجاد پروژه جدید(همراه با ایجاد یک فولدر):

عبارت my_project نام دلخواهی است که شما برای پروژه انتخاب میکنید.

این دستور برای شما یک فولدر نیز ایجاد میکند و داخل آن فولدر فایل‌های پروژه را قرار میدهد. اگر بخواهیم مستقیم فایل‌های پروژه داخل همین فولدری که هستیم (دایرکتوری فعلی) ایجاد کنه، انتهای دستور یک نقطه(دات) می‌ذاریم:

```
django-admin startproject config .
```

عبارت config نام پروژه شما همیشه و چون یک فولدر با این نام ایجاد میشه و درونش فایل‌های زیرساختی پروژه ما از جمله settings.py قرار میگیره، معمولا اسمش رو می‌ذارن config اما انتخاب اسم کاملا دلخواه است.

```
python manage.py startapp app_name
```

✱ ایجاد یک اپلیکیشن جدید:

```
python manage.py runserver
```

✱ ران کردن پروژه:

اگر انتهای دستور هیچی ننویسیم، بصورت پیش‌فرض روی پورت ۸۰۰۰ اجرا میشه. میتونیم پورت خاصی رو بهش اختصاص بدیم

```
python manage.py runserver 8001
```

که در اونجا ران بشه بطور مثال:

- اگر بخواهیم سرور را از متوقف کنیم، کافیه کلیدهای ترکیبی Ctrl + C را بزنیم.

✱ ایجاد مایگریشن (پس از ایجاد یا ادیت کردن یک یا چند مدل در models.py این دستور را بزنید):

```
python manage.py makemigrations
```

✱ ذخیره جداول مایگریشن در دیتابیس (قبل از اینکار باید ابتدا دستور بالا زده شده باشد) :

```
python manage.py migrate
```

`python manage.py createsuperuser`

✱ ایجاد یک ابرکاربر (ادمین) جدید:

✱ انتقال فایل‌های استاتیک به روت پروژه (برای استفاده از این دستور لازم است که مقدار `STATIC_ROOT` در

`python manage.py collectstatic`

`settings.py` تعریف شده باشد) :

`STATIC_ROOT = os.path.join(BASE_DIR, "static")`

نحوه تعریف متغیر `STATIC_ROOT` :

`import os`

حواستان باشد قبل از آن `os` را ایمپورت کرده باشید:

`python manage.py shell`

✱ ورود به محیط شل جنگو (جهت ارتباط با دیتابیس) :

<https://www.djangoproject.com>

✱ آدرس سایت رسمی جنگو (معرفی جنگو و داکيومنت):

<https://djangopackages.org>

✱ آدرس سایت رسمی پکیج‌های جنگو:

این تمرین متناسب با مثال‌هایی که در دنیای واقعیت و صنعت بازار کار با آن روبه‌رو خواهیم شد، طراحی شده است.

با آرزوی موفقیت و پیشرفت شما عزیزان.

مدرس: عرفان اسماعیلی شیرازی