



به نام خدا
مکتب شریف



بوتکمپ پایتون ۱۰۲ پروژه‌ی گروهی

git	موضوعات تمرین
Regex	
API	

در این پروژه ترکیبی از مطالب مطرح شده در جلسات Git , Regex و کار کردن با api ها مورد سوال قرار گرفته است. همچنین فایل هایی مرتبط با پروژه در پوشه front قرار گرفته است که با توجه به صورت سوال باید از این فایل ها استفاده کرد.

۱- Clone کردن فایل ها

ابتدا repository موجود در این [لینک](#) را در repository گروهی خود fork کنید. سپس هریک از اعضای گروه repository ای که fork شده را با استفاده از دستور clone روی سیستم خود کپی کنند.

۲- مرتب کردن فایل ها

در فایل های clone شده پوشه ای به نام front وجود دارد که شامل فایل های مختلف است. اما مشکلی که در این فایل ها دیده می شود این است که تصاویر اصلی و Icon ها در یک پوشه قرار گرفته است و این موضوع باعث می شود نظم پروژه از بین برود. برای همین تصمیم گرفته شده که Icon ها در یک پوشه ای جداگانه ذخیره شوند و سایر تصاویر در پوشه ای جداگانه.

۲-۱

در این قسمت از پروژه اعضای گروه باید برنامه ای بنویسند که با استفاده از امکانات کتابخانهی pathlib و متد glob بتواند تمامی فایل های موجود در پوشه ای images را به ترتیب حروف الفبا نمایش دهد.

۲-۲

بعد از نمایش اسامی فایل ها، باید کدی نوشته شود که بتواند فایل های icon را از سایر فایل های تصویری جدا کند. برای این کار توجه داشته باشید که تمامی icon ها به صورتی نام گذاری شده اند که یکی از الگوهای زیر در آنها رعایت شده است

- در انتهای اسم فایل و قبل از پسوند فایل الگوی icon- مشاهده می شود
- در انتهای اسم فایل و قبل از پسوند فایل الگوی icon-1- مشاهده می شود

با استفاده از خروجی مرحله ی قبل و نوشتن الگویی در تابع glob برنامه ای بنویسید که بتواند تمامی فایل های Icon را از فایل های غیر Icon تشخیص دهد. آدرس فایل های خروجی را در فایلی با نام Icons.txt ذخیره کنید.

۲-۳

با استفاده از فایل خروجی مرحله ی قبل در ترمینال لینوکس کاری کنید که فایل های موجود در فایل Icons.txt در پوشه ای جداگانه با نام Icons ذخیره شوند.

۲-۴

چون در مرحله ی قبل محل فایل های Icon جابه جا شد فایل های html دیگر قابلیت اتصال به فایل های Icon را ندارند. برنامه ی پایتونی بنویسید تمامی فایل های html موجود در پوشه ی front را پیدا کند(با استفاده از glob و pathlib) و در هر یک از فایل های پیدا شده جستجو کنید و به دنبال تصویر بگردید که src آنها با استفاده از فایل های icon آدرس دهی شده باشند. سپس آدرس جدید فایل های Icon را در این فایل ها جایگزین کنید. به گونه ای که فایل های HTML به درستی load شوند.

۲-۵

هر عضو گروه branch متعلق به خود را در repository ایجاد کند و فایل about.html را با استفاده از مشخصات خود تغییر دهد. این تغییرات دلخواه است. برای مثال می توانید مشخصات خود را در قسمت زیر قرار دهید



About Us

>Lorem ipsum dolor sit amet, consectetur
adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis

تغییرات اعمال شده را ذخیره کنید و در branch ای که با نام خود ایجاد کرده اید commit کنید.

۲-۶

کد نوشته شده را روی repository مشترک گروه خود ارسال کنید. نهایتا سرگروه هر تیم با کمک اعضای تیم پروژه ی نهایی را با merge کردن تمامی branch های موجود و رفع conflict های احتمالی ایجاد کند. پروژه ی نهایی را در شاخه ی اصلی مخزن کد گروهتان commit کنید.

۳- استفاده از api گیت هاب

در این قسمت می‌خواهیم با استفاده از کتابخانه‌ی request اطلاعات مخزن پروژه‌ی قبلی را بررسی کنیم. با استفاده از کتابخانه requests یک http request با متد get به آدرس زیر بزنید تا اطلاعات مخزن‌های عمومی شما نمایش داده شود

```
https://api.github.com/users/{user}/repos
```

برای بررسی مخزن‌های خصوصی باید token برای احراز هویت داشته باشید. با استفاده از این [لینک](#) یک token گیت هاب بسازید.

اگر بخواهید به api‌هایی دسترسی داشته باشید که به صورت خصوصی هستند باید به نحوی احراز هویت شوید. این احراز هویت در حالت عادی با استفاده از username و password است که در صفحات login وارد میکنید. اما در زمان استفاده از کتابخانه request با استفاده از token ساخته شده از مرحله قبل میتوانیم احراز هویت را انجام دهیم. برای این کار باید اطلاعات token را در قسمت headers موجود در request قرار دهیم. با مطالعه‌ی این [لینک](#) در این مورد تحقیق کنید. دقت داشته باشد که نهایتاً اطلاعات headers به صورت زیر خواهد بود:

```
headers = {'Authorization': f'Bearer {token}',  
           'Accept': 'application/vnd.github+json',}
```

با ارسال request به آدرس زیر تمامی اطلاعات commit‌های انجام شده روی repository خود را مشاهده کنید.

```
url = "https://api.github.com/repos/{username}/{repo}/commits"
```

اطلاعات commit‌های موجود روی repository مربوط به این پروژه را دریافت کرده و بر اساس نویسنده‌ی مرتبط کنید و در فایل json ذخیره کنید.