

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Instagram Downloader برنامه

معرفی

این پروژه یک برنامه برای دانلود استوری‌ها و پست‌های کاربران اینستاگرام به صورت اتوماتیک است. این برنامه با استفاده از کتابخانه‌های `tkinter` برای ایجاد رابط کاربری گرافیکی (GUI)، `instaloader` برای تعامل با API اینستاگرام و کتابخانه‌های `os`، `datetime` و `pytz` برای مدیریت تاریخ‌ها و فایل‌ها پیاده‌سازی شده است.

کتابخانه‌ها و ابزارهای استفاده‌شده

- Tkinter :

کتابخانه استاندارد پایتون برای ایجاد رابط کاربری گرافیکی است `tkinter`. به شما این امکان را می‌دهد تا ویجت‌ها و عناصر گرافیکی را برای تعامل با کاربر ایجاد کنید. در این پروژه از `tkinter` برای ایجاد فرم‌های ورود اطلاعات و دکمه‌ها استفاده شده است.

- MessageBox :

یکی از ماژول‌های `tkinter` است که برای نمایش پیام‌های اطلاعاتی، اخطارها و خطاها به کاربر استفاده می‌شود. این ماژول برای اطلاع‌رسانی به کاربر در مورد وضعیت عملیات (مثل موفقیت‌آمیز بودن ورود به سیستم یا خطاها) استفاده شده است.

- Datetime :

این کتابخانه برای کار با تاریخ‌ها و زمان‌ها استفاده می‌شود. در این پروژه از `datetime` برای تبدیل و مقایسه تاریخ‌ها استفاده شده است.

- Os :

این کتابخانه برای کار با سیستم فایل و دسترسی به فایل‌ها و پوشه‌ها استفاده می‌شود. در این پروژه از `os` برای ایجاد پوشه‌ها و بررسی وجود فایل‌های `session` استفاده شده است.

- Pytz:

کتابخانه‌ای برای کار با مناطق زمانی مختلف است **pytz** به شما این امکان را می‌دهد تا زمان را به مناطق زمانی مختلف تبدیل کنید. در این پروژه از **pytz** برای تبدیل زمان‌های UTC به زمان محلی تهران استفاده شده است.

- Time:

این کتابخانه برای کنترل اجرای برنامه و ایجاد تاخیر زمانی استفاده می‌شود. در این پروژه از **time** برای تاخیر بین دانلودها استفاده شده است تا از درخواست‌های بیش از حد به سرور اینستاگرام جلوگیری شود.

- Instaloader:

کتابخانه‌ای است که برای دانلود پست‌ها، استوری‌ها و سایر داده‌های اینستاگرام به کار می‌رود. این کتابخانه به شما امکان می‌دهد تا بدون نیاز به استفاده از **API** رسمی اینستاگرام، داده‌های کاربران را به صورت مستقیم دریافت کنید.

توضیح کد

ایجاد شیء Instaloader:

```
# ایجاد شیء Instaloader
L = instaloader.Instaloader()
```

- این خط کد یک شیء از کلاس **Instaloader** ایجاد می‌کند که برای تعامل با **API** اینستاگرام و دانلود محتوا استفاده می‌شود.

تابع login:

```
# تابع برای لاگین با استفاده از سشن
def login():
```

- این تابع برای ورود به حساب اینستاگرام کاربر و بارگذاری یا ذخیره سشن استفاده می‌شود. این تابع ابتدا بررسی می‌کند که آیا فایل سشن برای کاربر موجود است یا خیر. اگر موجود باشد، سشن بارگذاری می‌شود؛ در غیر این صورت، کاربر لاگین کرده و سشن ذخیره می‌شود.

تابع `make_aware` :

```
# aware به naive تابع برای تبدیل تاریخ
def make_aware(date_naive, timezone_str='Asia/Tehran'):
    tz = pytz.timezone(timezone_str)
    return tz.localize(date_naive)
```

- این تابع تاریخ‌های (naive) بدون منطقه زمانی را به تاریخ‌های (aware) با منطقه زمانی تبدیل می‌کند. به طور پیش‌فرض از منطقه زمانی تهران استفاده می‌شود.

تابع `download_stories` :

```
# تابع برای دانلود استوری‌ها
def download_stories(user_target):
```

- این تابع استوری‌های کاربر هدف را دانلود می‌کند. ابتدا پروفایل کاربر هدف را دریافت کرده و سپس با استفاده از `get_stories` استوری‌ها را دانلود و در دایرکتوری مناسب ذخیره می‌کند.

تابع `download_posts` :

```
# تابع برای دانلود پست‌ها
def download_posts(user_target, specific_date_str=None, start_date_str=None, end_date_str=None):
```

- این تابع برای دانلود پست‌های کاربر هدف در یک تاریخ مشخص یا بازه زمانی مشخص استفاده می‌شود. ابتدا تاریخ‌های مشخص شده به منطقه زمانی تهران تبدیل می‌شوند و سپس پست‌ها دانلود و در دایرکتوری مناسب ذخیره می‌شوند.

تابع `validate_date` :

```
# تابع برای اعتبارسنجی تاریخ‌ها
def validate_date(date_str):
```

- این تابع برای اعتبارسنجی فرمت تاریخ‌های ورودی استفاده می‌شود. اگر تاریخ معتبر باشد، آن را به فرمت `datetime` تبدیل می‌کند؛ در غیر این صورت، خطا را به کاربر نشان می‌دهد.

تابع start_download :

```
# تابع برای شروع عملیات دانلود  
def start_download():
```

- این تابع عملیات دانلود را شروع می‌کند. ابتدا ورودی‌های کاربر مانند نام‌های کاربری هدف، نوع عملیات (استوری، پست، یا هر دو) و تاریخ‌ها را پردازش کرده و سپس عملیات دانلود را بر اساس این ورودی‌ها انجام می‌دهد.

رابط کاربری گرافیکی :

```
# ایجاد رابط گرافیکی  
root = tk.Tk()  
root.title("Instagram Downloader")  
  
root.mainloop()
```

- این بخش از کد، رابط کاربری گرافیکی برنامه را ایجاد می‌کند. این رابط شامل ورودی‌های کاربر برای نام کاربری، رمز عبور، نام‌های کاربری هدف، انتخاب نوع عملیات، و تاریخ‌ها است. دکمه‌های Login و Start Download نیز برای شروع عملیات مربوطه در نظر گرفته شده‌اند.

ورودی‌های کاربر

- Password و Username** : اطلاعات ورود به حساب کاربری اینستاگرام.
- Target Usernames** : نام‌های کاربری اینستاگرامی که می‌خواهید محتویات آن‌ها را دانلود کنید
- Operation** : نوع عملیات (دانلود استوری‌ها، پست‌ها یا هر دو).
- Date Choice** : انتخاب نوع تاریخ (تاریخ خاص یا بازه زمانی).
- Specific Date** : تاریخ خاصی که می‌خواهید پست‌های آن روز را دانلود کنید.
- End Date و Start Date** : بازه زمانی برای دانلود پست‌ها.

خروجی‌ها

فایل‌های دانلود شده (استوری‌ها و پست‌ها) در دایرکتوری‌های جداگانه‌ای با نام‌های مشخص بر اساس نام کاربری و تاریخ دانلود ذخیره می‌شوند .
به عنوان مثال، پست‌های یک کاربر در دایرکتوری با فرمت `post_{username}_{date}` ذخیره خواهند شد.

ساختار فایل‌ها و دایرکتوری‌ها

این اسکریپت فایل‌ها را در دایرکتوری‌های زیر ذخیره می‌کند:

- استوری‌ها `story_{username}_{date}` :
- پست‌ها `post_{username}_{date}` :

هر پست یا استوری دانلود شده با نامی مشخص بر اساس تاریخ و شناسه‌ی آن فایل ذخیره می‌شود.

معرفی Instaloder

Instaloder یک ابزار قدرتمند و کتابخانه پایتون است که برای دانلود محتوا از اینستاگرام استفاده می‌شود . این کتابخانه به کاربران امکان می‌دهد تا تصاویر، ویدیوها، استوری‌ها، پست‌ها و حتی پروفایل‌ها را به سادگی از اینستاگرام دانلود کنند . همچنین می‌توانید اطلاعات مربوط به پست‌ها، کپشن‌ها، هشتگ‌ها، کامنت‌ها و ... را استخراج کنید.

ویژگی‌های کلیدی Instaloder

دانلود پست‌ها **Instaloder** : می‌تواند تمامی پست‌های یک پروفایل را به همراه تصاویر و ویدیوهای مرتبط با آن‌ها دانلود کند.

دانلود استوری‌ها : امکان دانلود استوری‌ها از پروفایل‌های عمومی و خصوصی (با داشتن دسترسی) وجود دارد.

دانلود هایلایت‌ها **Instaloder** : می‌تواند هایلایت‌های پروفایل‌ها را نیز دانلود کند.

ذخیره سشن لاگین : این کتابخانه امکان ذخیره و بارگذاری سشن لاگین را فراهم می‌کند تا برای استفاده‌های بعدی نیاز به وارد کردن دوباره اطلاعات کاربری نباشد.

دانلود پست‌های با هشتگ خاص : می‌توانید پست‌هایی که دارای یک هشتگ خاص هستند را به صورت خودکار دانلود کنید.

دسترسی به اطلاعات متا **Instaloader**: به شما اجازه می‌دهد تا به اطلاعات متادیتای پست‌ها و پروفایل‌ها، مانند تعداد لایک‌ها، کامنت‌ها، و فالوورها دسترسی پیدا کنید.

مثال‌هایی از استفاده از **Instaloader**

در کد بالا از **Instaloader** برای دانلود پست‌ها و استوری‌ها استفاده شده است. به عنوان مثال:

دانلود پست‌ها: تابع `download_posts` از **Instaloader.Profile** برای دسترسی به پروفایل کاربر استفاده می‌کند و سپس از `profile.get_posts()` برای بازیابی تمامی پست‌های کاربر بهره می‌برد.

دانلود استوری‌ها: تابع `download_stories` با استفاده از **Instaloader.get_stories** استوری‌های کاربر را بازیابی کرده و دانلود می‌کند.

ذخیره سشن: در تابع `login` از قابلیت `L.save_session_to_file` برای ذخیره سشن استفاده شده است که به شما امکان می‌دهد در مراجعه‌های بعدی به برنامه نیازی به ورود مجدد نداشته باشید.

نتیجه‌گیری

این اسکریپت به گونه‌ای طراحی شده است که کاربران بتوانند به راحتی استوری‌ها و پست‌های اینستاگرام را دانلود کنند. با استفاده از کتابخانه‌های مختلف پایتون، عملیات لاگین، مدیریت تاریخ و زمان، و دانلود محتوا به سادگی انجام می‌شود. رابط کاربری گرافیکی ایجاد شده توسط **tkinter** نیز به کاربران امکان می‌دهد که بدون نیاز به دانش برنامه‌نویسی با اسکریپت تعامل کنند.