

Парсер инстаграма

Зависимости: python >= 3.8.0; requests>=2.26.0; PySocks>=1.7.1

Обращение к парсеру осуществляется из bash как python3 main.py [аргументы] в директории парсера.

Инициализация

```
python3 main.py auth {login} {password} {proxy(в формате  
'socks5h://127.0.0.1:9050', опционально) }
```

Авторизация в иг и сохранение куки для последующих запросов. Для создания нескольких корневых процессов необходимо создать несколько копий директории с парсером.

Получение актуальных stories

```
python3 main.py stories {username}
```

Получить ссылки на контент актуальных stories пользователя с именем {username}.

Возвращает список ссылок.

Получение списка альбомов highlights

```
python3 main.py hl_list {username}
```

Получить список альбомов highlights аккаунта с именем {username}.

Возвращает список словарей формата {id, title, thumbnail(url обложки)}

Получение ссылок на контент альбома highlights

```
python3 main.py hl {hl_id}
```

Получить ссылки на контент альбома highlights с {hl_id}.

Возвращает список ссылок

Получение постов, фильтр по номеру

```
python3 main.py posts {username} {start} {finish}
```

Получить список постов аккаунта с именем {username} от поста с номером {start} до поста с номером {finish}(не включая). Нумерация начинается от нуля, нулевым является самый свежий пост.

Возвращает список словарей формата {id, urls(список адресов контента поста), caption(текст поста), likes (число лайков), comments(число комментариев), timestamp(дата и время снимка в секундах от начала эпохи)}

Получение постов, фильтр по дате

```
python3 main.py posts_d {username} {start_date} {fin_date}
```

Получить список постов аккаунта с именем {username} загруженных между датами {start_date} и {fin_date} (дата и время поста в секундах от начала эпохи).

Возвращает список словарей формата {id, urls(список адресов контента поста), caption(текст поста), likes (число лайков), comments(число комментариев), timestamp(дата и время снимка в секундах от начала эпохи)}

Получение reels, фильтр по номеру

```
python3 main.py reels {username} {start} {finish}
```

Получить список reels аккаунта с именем {username} от поста с номером {start} до поста с номером {finish}(не включая). Нумерация начинается от нуля, нулевым является самый свежий reel.

Возвращает список словарей формата {id, url(адрес контента reel), likes (число лайков), comments(число комментариев), timestamp(дата и время поста в секундах от начала эпохи)}

Получение reels, фильтр по дате

```
python3 main.py reels_d {username} {start_date} {fin_date}
```

Получить список reels аккаунта с именем {username} загруженных между датами {start_date} и {fin_date}(дата и время публикации в секундах от начала эпохи). Нумерация начинается от нуля, нулевым является самый свежий reel.

Формат возвращаемого значения: список словарей формата {id, url(адрес контента reel), likes (число лайков), comments(число комментариев), timestamp(дата и время публикации в секундах от начала эпохи)}

Получение instagram user id

```
python3 main.py uid {username}
```

Получить instagram id пользователя с именем {username}

Получение аватара

```
python3 main.py ava {username}
```

Получить аватар пользователя с именем {username}

В случаях, когда статусом ответа на запрос не является 200, выводится код ответа и причина ошибки.