

سلنیوم: از اتوماسیون پیشرفته تا اسکرین شات‌های هوشمند و اسکرپینگ حرفه‌ای 🚀

اگر قبلاً با سلنیوم کار کرده‌اید، این مقاله دنیای جدیدی از قابلیت‌های پنهان آن را به شما نشان می‌دهد! از تکنیک‌های حرفه‌ای اسکرین شات تا اسکرپ داده‌های پیچیده، همه را با جزئیات عملی یاد بگیرید.

🔥 سلنیوم: نه فقط برای تست، بلکه برای مهندسی داده!

بله! سلنیوم فقط برای تست خودکار نیست. تصور کنید:

- ساخت دیتابیس واقعی از قیمت مسکن با اسکرپ ۲۰ سایت املاک
- خودکارسازی پر کردن فرم‌های اداری تکراری
- مانیتورینگ لحظه‌ای تغییرات محتوای سایت‌های رقیب

🎯 هنر اسکرین شات: از پایه تا پیشرفته با ۵ لایه حرفه‌ای

🔍 ۱. اسکرین شات تمام صفحه (Full Page Screenshot)

با این تکنیک حتی قسمت‌های خارج از دید مرورگر هم ذخیره می‌شوند:

```
from selenium import webdriver

def full_page_screenshot(driver, file_name):
    total_height = driver.execute_script("return document.body.parentNode.scrollHeight")
    driver.set_window_size(1920, total_height)
    driver.save_screenshot(file_name)

driver = webdriver.Chrome()
driver.get("https://example.com")
full_page_screenshot(driver, "full_page.png")
```

🚨 ۲. اسکرین شات هوشمند با تشخیص خودکار خطا

با ترکیب try-except و اسکرین شات:

```
from selenium.common.exceptions import NoSuchElementException

:try
driver.find_element(By.ID, "element_id")
:except NoSuchElementException
driver.save_screenshot("error_"+ datetime.now().strftime("%Y%m%d_%H%M%S") + ".png")
raise
```

۳. اسکرین شت از المان‌های پویا با هایلایت

با این کد، المان مورد نظر حاشیه قرمز می‌گیرد:

```
element = driver.find_element(By.ID, "target_element")
driver.execute_script("arguments[0].style.border = '3px solid red'", element)
element.screenshot("element_shot.png")
```

۴. اسکرین شات متحرک (GIF)

با استفاده از کتابخانه imageio :

```
import imageio
from PIL import Image

[] = screenshots
:for _ in range(10)
driver.save_screenshot(f"frame_{_}.png")
screenshots.append(Image.open(f"frame_{_}.png"))

imageio.mimsave('animation.gif', screenshots, duration=0.5)
```

۵. اسکرین شات سه‌بعدی با استفاده از CSS Transform

شبیه‌سازی نمای سه‌بعدی:

```
driver.execute_script("document.body.style.transform = 'rotateY(20deg)';")
driver.save_screenshot("3d_view.png")
```

اسکراپینگ پیشرفته: عبور از تله‌های ضد بات

تکنیک‌های حیاتی:

- شبیه‌سازی رفتار انسانی:

استفاده از ActionChains برای حرکت ماوس تصادفی

```
from selenium.webdriver.common.action_chains import ActionChains
import random

actions = ActionChains(driver)
:for _ in range(5)
actions.move_by_offset(random.randint(-50,50), random.randint(-50,50))
()actions.perform
```

- دور زدن تشخیص هدلس:

افزودن ویژگی‌های مرورگر واقعی:

```
()options = webdriver.ChromeOptions
options.add_argument("--disable-blink-features=AutomationControlled")
l_argument("user-agent=Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36...")
```

- اسکراپ داده‌های جدولی پیشرفته:

استخراج داده‌های پیچیده از جداول:

```
table = driver.find_element(By.TAG_NAME, "table")
rows = table.find_elements(By.TAG_NAME, "tr")

[] = data
:for row in rows
cols = row.find_elements(By.TAG_NAME, "td")
data.append([col.text for col in cols if col.text.strip() != ''])
```

- اسکراپ صفحات با اسکرول نامحدود:

اتوماتیک کردن اسکرول تا انتهای صفحه:

```
last_height = driver.execute_script("return document.body.scrollHeight")
:while True
driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
time.sleep(2)
new_height = driver.execute_script("return document.body.scrollHeight")
:if new_height == last_height
break
last_height = new_height
```

یکپارچه سازی با سیستم های گزارش گیری

مثال با Allure Report

```
import allure

('تست لاگین')allure.feature@
:()def test_login
:try
# کد تست
:except Exception as e
, "خطای لاگین"=allure.attach(driver.get_screenshot_as_png(), name
(attachment_type=allure.attachment_type.PNG
raise e
```

تولید گزارش HTML سفارشی:

```
from jinja2 import Template

'''report_template = Template
<DOCTYPE html!>
<html>
<head>
<title/>گزارش تست<title>
<head/>
<body>
{% for img in screenshots %}
<h2>{{ img.title }}</h2>
<"img src="{{ img.path }}" width="80%>
{% endfor %}
<body/>
<html/>
(''

# تولید گزارش
]=report_html = report_template.render(screenshots
,{ 'path': 'error_login.png' , 'خطای لاگین': 'title' }
{ 'path': 'homepage.png' , 'صفحه اصلی': 'title' }
([
:with open('report.html', 'w') as f
f.write(report_html)
```

ترفندهای طلایی برای حرفه‌ای‌ها 💡

1. شبیه‌سازی تایپ واقعی:

با تاخیر تصادفی بین کی‌استروک‌ها:

```
from selenium.webdriver.common.keys import Keys
import random
```

```
:def human_type(element, text)
:for char in text
element.send_keys(char)
time.sleep(random.uniform(0.1, 0.3))
element.send_keys(Keys.RETURN)
```

2. عبور از CAPTCHA:

(فقط برای اهداف آموزشی!)

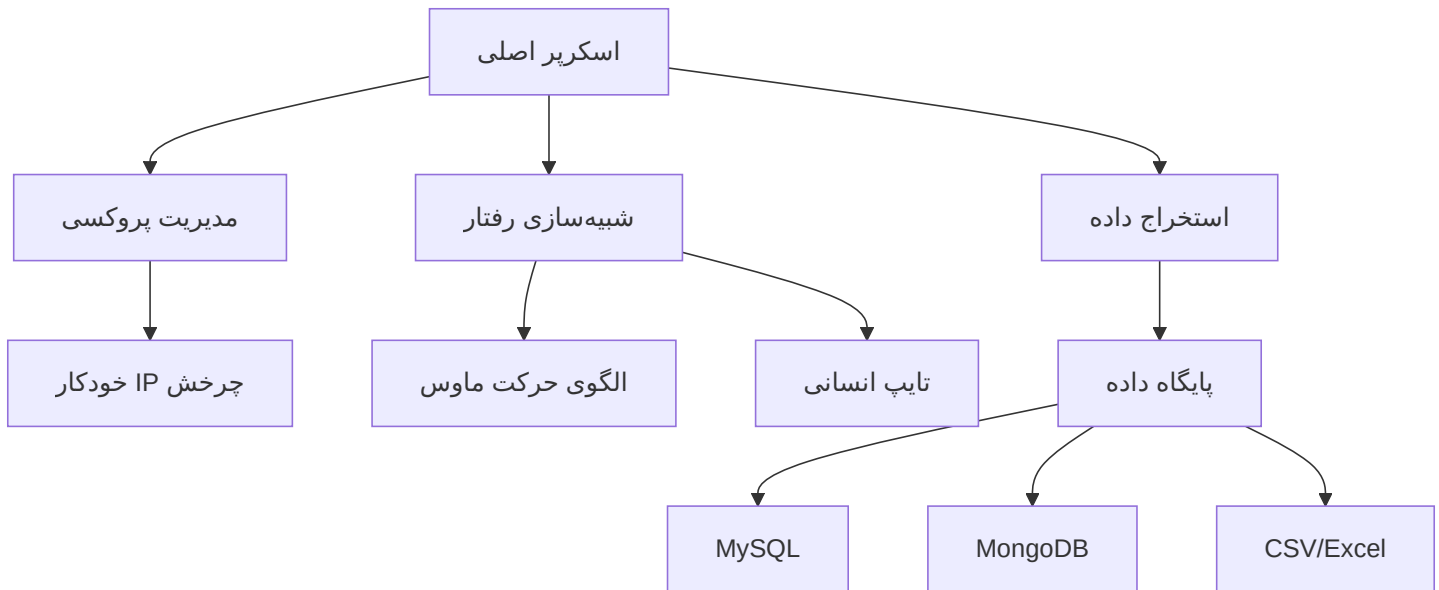
- استفاده از سرویس‌های حل CAPTCHA مانند 2Captcha
- تشخیص متن با کتابخانه pytesseract

3. مدیریت کوکی‌ها:

ذخیره و بازیابی سشن:

```
# ذخیره کوکی‌ها
import pickle
pickle.dump(driver.get_cookies(), open("cookies.pkl", "wb"))
```

```
# بارگذاری کوکی‌ها
cookies = pickle.load(open("cookies.pkl", "rb"))
:for cookie in cookies
driver.add_cookie(cookie)
```




جمع‌بندی نهایی: سلیوم به عنوان سوئیس چاقوی دیجیتال

امروز یاد گرفتیم:

- گرفتن اسکرین‌شات‌های هوشمند با قابلیت ردیابی خطا
- ساخت سیستم گزارش‌گیری خودکار حرفه‌ای
- تکنیک‌های اسکرپ داده‌های مقاوم در برابر تشخیص
- بهینه‌سازی عملکرد در مقیاس بزرگ

به یاد داشته باشید: اتوماسیون هنر است! هر پروژه نیاز به ترکیبی از خلاقیت و تکنیک‌های فنی دارد.

اگر سوالی دارید یا نیاز به راهنمایی در مورد سناریوهای خاص دارید، در کامنت‌ها مطرح کنید. در دنیای اتوماسیون موفق باشید! 

ارتباط با من

- ایمیل: darbandidr99@gmail.com
- گیت‌هاب: github.com/idarbandi
- مقاله تجربه شخصی من در یادگیری اتوماسیون مرورگر با پایتون: [مطالعه در ویرگول \(فارسی\)](#)

نوشته شده توسط: امیر حسین دربندی
پاینده باشید!