

BROWSER automation

How can we automate such a task, and how to scrape the web like a spider

About me

Hi, i'm fakhri also called fai.

software **developer** and/or
breaker

[d()] **DASKOM 1337**



BROWSER automation

Nightmarejs

Browser automation is about automating user actions (usually) in a headless browser

Automasi browser adalah tentang mengotomasi aksi aksi user (biasanya) dengan menggunakan headless browser

Why nightmarejs ?

- **Simple** api
- **Straightforward** usage
- Perfect for trivial apps that don't need too much of code control / customization

How it works

Nightmare()

```
.goto('http://duckduckgo.com')  
.type('#search_form_input_homepage', 'daskom1337')  
.click('#search_button_homepage');
```

1. This will open up duckduckgo.com
2. Find element using selector of id = search_form_input_homepage
3. Click on element with id = search_button_homepage



01

LMS login automation

This is the first step of
autobot system

How it works

1. Open up <https://lms.telkomuniversity.ac.id/auth/oidc> that will redirect us to the office 365 login page
2. **Wait for 5000 ms** (until office 365 login page successfully opened)
3. Type the **username** on corresponding element
4. Click on next button
5. Type the **password** on corresponding element
6. Click next
7. Click no to **not remember the login session**
8. Wait for 5000 ms and wait until the LMS site rendered successfully
9. Get the **`MoodleSession`** cookie

So many possibilities of what we can
do with browser automation

And even more powerful when
combined with other stuff

WEB scraping

BeautifulSoup

Web scraping is about getting all information out of html element (usually inside a webpage)

Scraping web adalah tentang mendapatkan informasi dari elemen elemen pada html (yang biasanya diambil dari sebuah halaman web))

Alternatives to beautifulsoup

Source:

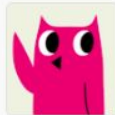
<https://stackshare.io/beautifulsoup/alternatives>

TOP ALTERNATIVES TO BEAUTIFULSOUP



Scrapy

It is the most popular web scraping framework in Python. An open source and ...



import.io

Import.io is a free web-based platform that puts the power of the machine readable ...



Portia

Portia is an open source tool that lets you get data from websites. It facilitates ...



Apifier

Apifier is a hosted web crawler for developers that enables them to extract ...



Selenium

Selenium automates browsers. That's it! What you do with that power is entirely ...



ParseHub

You can extract data from anywhere. ParseHub works with single-page apps, multi-page ...



Kimono

You don't need to write any code or install any software to extract data with ...



Octoparse

It is a free client-side Windows web scraping software that turns unstructured ...

Why don't we just use selenium ?

It's rather slow when compared to nightmare, and even harder to learn for a beginner

And also **we don't need a big test case scenario** to handle

But for the case of a big project, surely it's worth it

How to use BeautifulSoup

```
from bs4 import BeautifulSoup
soup = BeautifulSoup(html_doc, 'html.parser')
anchors = soup.find_all('a')
for i, a in enumerate(anchors):
    print(f"link ke-{i+1} = {a}")
```

For outputting all
anchor tag inside the
html_doc string

Heads up here for full documentation:
<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>



02

LMS get session key

This is the second step of
autobot system

How it works

```
res = self.__session.get(self.BASE_DOMAIN.format(self.GET_URL['home']))
soup = BeautifulSoup(res.text, 'html.parser') #parse html content from response
hidden_inp = soup.find_all(attrs={"name": "sesskey"}) #find element with name = "sesskey"
if len(hidden_inp) > 0 :
    try:
        chosen_inp = hidden_inp[0] #get the first matching element
        self.__apikey = chosen_inp.attrs['value'] #grab the value of selected element
    except :
        print('[!] Error parsing API key')
```



**AUTOBOT V0.01 Alpha
RELEASE**

[d()] DASKOM 1337

THANKS



Ask any questions ?, and hopefully
I can answer all of them

CREDITS: This presentation template was created
by **Slidesgo**, including icons by **Flaticon**, and
infographics & images by **Freepik**
Please keep this slide for attribution.