




Python Discord Bot







DAY 07

All about Web-scraping

Contents

Exercise 01: \$scrape	3
Exercise 02: \$programming	4
Exercise 03: \$currency	6
Exercise 04: \$job	8
 Bonus 01: \$movie	11


Exercise 01: \$scrape

	Difficulty:	★ ★ ★
	Objectives:	Use a command to instruct the discord bot to scrape the first paragraph from selected website.
	Skills required:	<ul style="list-style-type: none">➤ Beautiful soup➤ Inspect Html➤ Python requests get method
	Resources:	<p>Selected website for web scraping: Talk about Programming</p> <p>For your reference: Beautiful Soup Documentation: Going down</p>
	Search keywords:	<ul style="list-style-type: none">➤ Python requests get method➤ Implementing Web Scraping in Python with BeautifulSoup
	Command required:	\$scrape







Code snippet:

```
Import requests
From bs4 import BeautifulSoup
```

Output example:

 **Code4Teens** BOT Today at 00:24
Programming is something vast and rather individual as each developer chooses tools that are most convenient for them.

Exercise 02: \$programming

	Difficulty:	★ ★ ★
	Objectives:	Use a command to instruct the discord bot to scrape the “Designed by” of the programming language in the table from selected website.
	Skills required:	<ul style="list-style-type: none"> ➤ Beautiful soup (Find next) ➤ Inspect Html ➤ Python requests get method ➤ Python string format method
	Resources:	<p>Selected website for web scraping: Talk about Programming</p> <p>For your reference:</p> <ul style="list-style-type: none"> ➤ Beautiful Soup Documentation: find_all_next() and find_next() ➤ Python f-Strings: The Pesky Details
	Search keywords:	<ul style="list-style-type: none"> ➤ Python requests get method ➤ Implementing Web Scraping in Python with BeautifulSoup ➤ BeautifulSoup find next
	Command required:	\$programming



Code snippet:

```
Import requests
From bs4 import BeautifulSoup

...

output = f"**__{{title}}__** \nPython : {gettext_python} \nC :
{gettext_c} \nC++ : {gettext_cpp} \nJava : {gettext_java}
\nJavaScript : {gettext_js} \nR : {gettext_r} \n"

return (output)
```



Output example:



Code4Teens BOT Today at 00:37

Designed by.

Python : Guido van Rossum

C : Dennis Ritchie







C++ : Bjarne Stroustrup

Java : James Gosling

JavaScript : Brendan Eich

R : Ross Ihaka and Robert Gentleman

Exercise 03: \$currency

	Difficulty:	★★★★
	Objectives:	<p>Use a command to instruct the discord bot to scrape the exchange rate of US Dollar from the selected website.</p> <p>! You must use web scraping to scrape the Currency Name (USD Big) and the Exchange Rate (We Buy Rate).</p>
	Skills required:	<ul style="list-style-type: none"> ➤ Beautiful soup (Going down & Going sideways) ➤ Inspect Html ➤ Python requests get method ➤ Python string format method ➤ Python Split and Join
	Resources:	<p>Selected website for web scraping: Daily Exchange Rates</p> <p>For your reference:</p> <ul style="list-style-type: none"> ➤ Beautiful Soup Documentation: Going down ➤ Beautiful Soup Documentation: Going sideways
	Search keywords:	<ul style="list-style-type: none"> ➤ Python requests get method ➤ Implementing Web Scraping in Python with BeautifulSoup ➤ Python BeautifulSoup Navigating tree sideways ➤ String split and join in python
	Command required:	\$currency



Code snippet:

```
Import requests
From bs4 import BeautifulSoup

...

output = "Currency: {}\nExchange Rate (MYR):
{}".format(update_currency, rate)
```



Output example:









Code4Teens BOT Today at 00:43

Currency: US DOLLAR

Exchange Rate (MYR): 4.1179

CODE 4
TEENS

Exercise 04: \$job

	Difficulty:	★ ★ ★ ★
	Objectives:	<p>Part 1: Scrape a list of All Job Title, Company Name and Salary, print the result on the terminal.</p> <p>Part 2: Use a command to instruct the discord bot to scrape the info below:</p> <ol style="list-style-type: none"> 1. 1st and 2nd Job title, Company name and Salary in the selected webpage. 2. Output the result in an embed message form. <p>*Please refer to the output example below.</p>
	Skills required:	<ul style="list-style-type: none"> ➤ Web scraping ➤ For Loops ➤ List in Python ➤ Join function
	Resources:	<p>Selected website for web scraping: Looking for Jobs</p> <p>For your reference:</p> <ul style="list-style-type: none"> ➤ Beautiful Soup: Build a Web Scraper With Python ➤ Extracting Data from HTML with BeautifulSoup
	Search keywords:	<ul style="list-style-type: none"> ➤ Scraping data with multiple values within same class ➤ Converting list into string
	Command required:	\$job



Code snippet:

```
Import requests
From bs4 import BeautifulSoup

Part 1:
#Scrape all Jobs, Companies and Salaries using loop

#Put the result into a list and convert them into String

#Print the result in the terminal

Part 2:
#Compile the Joblist

#Send the list out as an embed message on Discord
```



Output example:

1

Part 1:

```
Job: Software Engineer
Job: Python Web Development Team Lead
Job: Consulting Analyst
Job: Application Developer / Programmer
```

```
Company: Dokonee
Company: iZyoo Platform Sdn Bhd
Company: Strella Consulting
Company: Plentisoft Sdn Bhd
```

```
Salary: RM6000 - RM12000
Salary: RM6000 - RM10000
Salary: RM2500 - RM3000
Salary: RM4000 - RM7000
```

2

Part 2:

Code4Teens **BOT** Today at 2:52 PM












Job List:

Job1:
Software Engineer, Dokonee
RM6000 - RM12000

Job2:
Python Web Development Team Lead, iZyoo Platform Sdn Bhd
RM6000 - RM10000

Bonus 01: \$movie

!! This is a BONUS question.

	Difficulty:	    
	Objectives:	Use a command to instruct the discord bot to scrape the storyline of the movie.
	Skills required:	<ul style="list-style-type: none"> ➤ String split or String slice ➤ Python requests get method ➤ Python string format method ➤ BeautifulSoup (Extract URL & Going down) ➤ Inspect Html ➤ Error Handling
	Resources:	<p>Selected website for web scraping: https://www.imdb.com/find?q={}&s=tt&ttype=ft&ref=fn_ft</p> <p>For your references:</p> <ul style="list-style-type: none"> ➤ Beautiful Soup Documentation: Quick Start ➤ Beautiful Soup Documentation: Going down
	Search keywords:	<ul style="list-style-type: none"> ➤ Python requests get method ➤ Python string format method ➤ Implementing Web Scraping in Python with BeautifulSoup ➤ Extract links from webpage using BeautifulSoup
	Command required:	<p>\$movie [name]</p> <p>a) Your bot must able to handle the error handling part: response when no movie name was inserted.</p>



Code snippet:

```
Import requests
From bs4 import BeautifulSoup

...

# Error Handling

...

# Search movie name via the search URL

...

# Extract link of the first movie from the results

...

# Scrape the storyline by using the new movie link

...

output = f"Movie Name: {movie_name} \nStoryline: {storyline}"
```



Output example:



Code4Teens BOT Today at 00:05

Movie Name: Girl From Nowhere

Storyline: A mysterious, clever girl named Nanno transfers to different schools, exposing the lies and misdeeds of the students and faculty at every turn.



Error Handling Example:



Code4Teens BOT Today at 01:48

Please enter a movie name.