



## Exercise for Unit 2

Name: Kirk Henrich C. Gamo

Date: Jan. 22, 2026

Year and Section: BSCS 3A AI

*Note: Write your answers in a one whole sheet of paper.*

1. **Updating ELIZA.** Modify the ELIZA python implementation shared to the class by performing the following:

(10 points each) Add 5 RegEx patterns to capture:

- a. I want to know the reasons why I am feeling depressed all the time.
- b. I am feeling stressed.
- c. My feelings towards my crush are invalidated.
- d. You don't understand me OR You do not understand me.
- e. I can't focus on my studies OR I cannot focus on my studies.

**BONUS! (10 points)** Modify the code so that if the user tries to repeat the same question, ELIZA will answer sarcastically on repeating the question.

```
(.venv) PS C:\Users\Acer\school\NLP> & C:/Users/Acer/school/NLP/.venv/Scripts/python.exe "c:/Users/Acer/school/NLP/CCS-249_25-26_Activities/BSCS_3A/KIRK_HENRICH_GAMO/GAMO-EXERCISE_2.py"
You: I want to know the reasons why I am feeling depressed all the time.
ELIZA: Understanding why you're feeling depressed is important. What do you think might be causing this?
You: I am feeling stressed.
ELIZA: What makes you feel stressed?
You: My feelings towards my crush are invalidated.
ELIZA: It sounds like your feelings about your crush are invalidated. Can you elaborate?
You: You don't understand me.
ELIZA: I'm trying to understand you. Can you help me see it from your perspective?
You: You do not understand me.
ELIZA: I'm trying to understand you. Can you help me see it from your perspective?
You: I can't focus on my studies.
ELIZA: What's preventing you from being able to focus on your studies?
You: I cannot focus on my studies.
ELIZA: What's preventing you from being able to focus on your studies?
You: I cannot focus on my studies.
ELIZA: Again? Really? I heard you the first time!
You: quit
ELIZA: Goodbye!
(.venv) PS C:\Users\Acer\school\NLP>
```

2. **Implementing RegEx on NLP.** Perform the following preprocessing techniques using RegEx.



- a. (10 points) Extract all of the *words starting with an upper case letter* from the text:

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do. Once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, "and what is the use of a book," thought Alice, "without pictures or conversations?"

RegEx pattern: `r'\b[A-Z][a-z]*\b'`

### Exercise A: Extract Words Starting with Uppercase Letters

Extract all of the words starting with an upper case letter from the given text.

```
# Given text
text_a = """Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothin

# RegEx pattern to match words starting with uppercase letter
pattern_a = r'\b[A-Z][a-z]*\b'

# Extract all matches
uppercase_words = re.findall(pattern_a, text_a)

print("Words starting with uppercase letter:")
print(uppercase_words)
print(f"\nTotal count: {len(uppercase_words)}")
```

Python

```
Words starting with uppercase letter:
['Alice', 'Once', 'Alice']
```

```
Total count: 3
```

- b. (15 points) Read the "melville-moby\_dick.txt" into a Python program and extract all of the instances of the word *Whale*, *Whales*, *whale* and *whales* in said source. Then replace the first 10 instances with the word "leviathan".

RegEx pattern: `r'\b[Ww]hales?\b'`



# West Visayas State University

COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY

Luna St., La Paz, Iloilo City 5000

Iloilo, Philippines

\* Trunkline: (063) (033) 320-0870 loc 1403 \* Telefax No.: (033) 320-0879

\* Website: [www.wvsu.edu.ph](http://www.wvsu.edu.ph) \* Email Address: [cict@wvsu.edu.ph](mailto:cict@wvsu.edu.ph)



```
# Read the Moby Dick text file
with open('melville-moby_dick.txt', 'r', encoding='utf-8') as file:
    moby_dick_text = file.read()

# RegEx pattern to match whale/whales (case-insensitive)
pattern_b = r'\b[ww]hales?\b'

# Extract all matches
whale_matches = re.findall(pattern_b, moby_dick_text)

print(f"Total instances of 'whale/whales' found: {len(whale_matches)}")
print(f"\nFirst 20 matches: {whale_matches[:20]}")

Total instances of 'whale/whales' found: 1489
First 20 matches: ['Whale', 'Whale', 'Whale', 'Whales', 'Whales', 'Whales', 'Whale', 'Whale', 'Whale', 'Whale',

# Replace the first 10 instances with "leviathan"
# Using a counter to track replacements
replacement_count = 0
max_replacements = 10

def replace_first_10(match):
    global replacement_count
    if replacement_count < max_replacements:
        replacement_count += 1
        return "leviathan"
    return match.group(0)

# Perform replacement
modified_text = re.sub(pattern_b, replace_first_10, moby_dick_text)

print(f"Replaced {replacement_count} instances with 'leviathan'")

# Show a sample of the modified text (first 2000 characters)
print("\nSample of modified text:")
print(modified_text[:2000])

[4] Replaced 10 instances with 'leviathan'

Sample of modified text:
The Project Gutenberg eBook of Moby Dick; Or, The leviathan

This ebook is for the use of anyone anywhere in the United States and
most other parts of the world at no cost and with almost no restrictions
whatsoever. You may copy it, give it away or re-use it under the terms
of the Project Gutenberg License included with this ebook or online
at www.gutenberg.org. If you are not located in the United States,
you will have to check the laws of the country where you are located
before using this eBook.
```



Title: Moby Dick; Or, The leviathan

Author: Herman Melville

Release date: July 1, 2001 [eBook #2701]

Most recently updated: January 19, 2025

Language: English

Credits: Daniel Lazarus, Jonesey, and David Widger

\*\*\* START OF THE PROJECT GUTENBERG EBOOK MOBY DICK; OR, THE WHALE \*\*\*

MOBY-DICK;

or, THE WHALE.

By Herman Melville

CONTENTS

ETYMOLOGY.

EXTRACTS (from Moby-Dick; Or, The Whale)

- c. (15 points) Download the NLTK package into your local environment and import *webtext* (from *nltk.corpus import webtext*). Load the *pirates.txt* and extract all of the lines spoken by the character Jack Sparrow.

RegEx pattern: `r'^(?:JACK(?:\s+SPARROW)?|SPARROW)\s*:\s*(.+)$'`



## Exercise C: Extract Jack Sparrow's Lines from Pirates.txt

Download NLTK package, import webtext, and extract all lines spoken by Jack Sparrow from pirates.txt

```
# Install NLTK (uncomment if not already installed)
# !pip install nltk
```

Python

```
# Import NLTK and download webtext corpus
import nltk
nltk.download('webtext')
```

Python

```
[nltk_data] Downloading package webtext to
[nltk_data]   C:\Users\Acer\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\webtext.zip.
```

True

```
# Import webtext corpus
from nltk.corpus import webtext

# Load pirates.txt
pirates_text = webtext.raw('pirates.txt')

# Display first 500 characters to understand the format
print("Sample of pirates.txt:")
print(pirates_text[:500])
```

Python



# West Visayas State University

## COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY

Luna St., La Paz, Iloilo City 5000

Iloilo, Philippines

\* Trunkline: (063) (033) 320-0870 loc 1403 \* Telefax No.: (033) 320-0879

\* Website: [www.wvsu.edu.ph](http://www.wvsu.edu.ph) \* Email Address: [cict@wvsu.edu.ph](mailto:cict@wvsu.edu.ph)



Sample of pirates.txt:

PIRATES OF THE CARRIBEAN: DEAD MAN'S CHEST, by Ted Elliott & Terry Rossio  
[view looking straight down at rolling swells, sound of wind and thunder, then a low heartbeat]  
Scene: PORT ROYAL  
[teacups on a table in the rain]  
[sheet music on music stands in the rain]  
[bouquet of white orchids, Elizabeth sitting in the rain holding the bouquet]  
[men rowing, men on horseback, to the sound of thunder]  
[EITC logo on flag blowing in the wind]  
[many rowboats are entering the harbor]  
[Elizabeth sitting alon

```
# RegEx pattern to extract Jack Sparrow's lines
# Pattern matches lines that start with variations of Jack's name followed by a colon
pattern_c = r'^(?:JACK(?:\s+SPARROW)?|SPARROW)\s*:\s*(.+)#'

# Extract all Jack Sparrow's lines (using MULTILINE flag)
jack_lines = re.findall(pattern_c, pirates_text, re.MULTILINE | re.IGNORECASE)

print(f"Total lines spoken by Jack Sparrow: {len(jack_lines)}")
print("\nFirst 10 lines:")
for i, line in enumerate(jack_lines[:10], 1):
    print(f"{i}. {line}")
```

Python

Total lines spoken by Jack Sparrow: 193

First 10 lines:

1. Sorry, mate.
2. Mind if we make a little side trip? I didn't think so.
3. Complications arose, ensued, were overcome.
4. Mm-hmm!
5. Shiny?
6. Is that how you're all feeling, then? Perhaps dear old Jack is not serving your best interests as captain?
7. What did the bird say?



```
# Display all Jack Sparrow's lines
print("All Jack Sparrow's lines:\n")
for i, line in enumerate(jack_lines, 1):
    print(f"{i}. {line}")
```

Python

All Jack Sparrow's lines:

```
1. Sorry, mate.
2. Mind if we make a little side trip? I didn't think so.
3. Complications arose, ensued, were overcome.
4. Mm-hmm!
5. Shiny?
6. Is that how you're all feeling, then? Perhaps dear old Jack is not serving your best interests as captain?
7. What did the bird say?
8. Ohhh!
9. It does me.
10. No! Much more better. It is a *drawing* of a key.
11. Gentlemen, what do keys do?
12. No! If we don't have the key, we can't open whatever it is we don't have that it unlocks. So what purpose we
13. You're not making any sense at all. Any more questions?
14. Hah! A heading. Set sail in a... mmm... a general... in *that* way - direction.
15. Come on, snap to and make sail, you know how this works. Come on, oy/quick, oy/quick, hey!
16. Why is the rum always gone?
17. Oh! *That's* why.
18. As you were, gents.
19. Ah!
20. Bootstrap. Bill Turner.
21. Is this a dream?
22. I thought not. If it were, there'd be rum.
23. I had some help retrieving the Pearl, by the way.
...
190. Come on... Come on...
191. Not so bad.
192. Oh!
193. Hello, beastie.
```

Output is truncated. View as scrollable element or open in a text editor. Adjust all output settings.

## Summary

### RegEx Patterns Used:

- **Exercise A:** `\b[A-Z][a-z]*\b` - Matches words starting with an uppercase letter
- **Exercise B:** `\b[Ww]hales?\b` - Matches whale/whales (case-insensitive)
- **Exercise C:** `^(?:JACK(?:\s+SPARROW)?|SPARROW)\s*:\s*(.+)$` - Matches Jack Sparrow's dialogue lines