Assignment 4

Environment

- Windows 10 22H2(OS Build 19045.2965)
- Python 3.10.2
  - nltk 3.8.2
- Visual Studio Code 1.78.2
- Git Bash GNU Bash version 4.4.23(2)-release (x86_64-pc-msys)

Problem 1

Output

```
problem2'slinkhttps://www.notion.so/ccs-binary/hw-4-6f0a449e2
```

Explanation

`T` is input string(cryptogram). First (in `clean_text` function) I splitted them into words, and removed string inside of cypher. Below is regular expression that I used.

```
new_word = re.sub("(^.*?[AB]+?)([^AB]+?)([AB]+?.*?$)", r"\1\3
```

First group(`(^.*?[AB]+?)`) means start of word(`^`), some non-cypher characters that should not be removed(`.*?`), and repeated `A` or `B`s(`[AB]+?`). Second group means non-cypher characters that should be removed(`([^AB]+?)`). Last group(`([AB]+?.*?$)`) means repeated `A` or `B`s(`[AB]+?`), some non-cypher characters that should not be removed(`.*?`), and end of word(`$`). This code will remove non-cypher characters inside words. However, it should be applied multiple times since there can be multiple non-cypher character bundles in word. For example, word `AAbundle1Bbundle2BB` need double replacement.

After remove non-cypher characters that should be removed, in `code_to_text` function, I replaced cypher into letter. There can be non-cypher-only text(like `e07` in cryptogram) so I checked whether the word includes cypher. `(.*?)([AB]{5})(.*?)` means that part. This means 5 AB characters(`[AB]{5}`) between some non-AB characters(`.*?`), which is named `prefix` and `postfix`. I first extracted second group, which is 5 AB characters, and replaced `A` to `0`, and `B` to `1`. Why I did this is the cypher increases in alphabetical order, so it can be added to `chr(97)`, which is `a`. At the end, the function returns prefix + cypher + postfix.

Problem 2

Output

```
E-mail: heejo @korea.ac.kr, hyunwoojkim @korea.ac.kr, suhtw @

Phone Number: 02-3290-3208, 02-3290-4604, 02-3290-2397, 02-32

HomePage: https://ccs.korea.ac.kr, https://mlv.korea.ac.kr, h
```

Explanation

I opened `probelm2.html` file(filename contains typo!), and what I did is just get adjacent tags or contents(text) to needed informations. See below.

```
mails = re.findall(
        r'''<b>E-mail</b>
                    <br/>
                    <a href="mailto:(.+?)"''',
        text,
    )
    phones = re.findall(
        r"""<b>전화번호:</b>
                    (.+?)
                </dd>""",
        text,
    )
    sites = re.findall(
```

```
        r'''<b>홈페이지:</b>
                    <a href="(.+?)"''',
        text,
    )
```

`(.+?)` means any characters that appears at least one time, and match it non-greedily(in other words, lazily. it means it will matched as least as possible).

Problem 3

Output

```
# Result over 10 html files
# "words" -  frequency
1. "is" -  6
2. "vulnerability" -  5
3. "code" -  4
4. "Odoo" -  4
5. "attackers" -  3
6. "memory" -  3
7. "execution" -  3
8. "Server" -  3
9. "issue" -  3
10. "allows" -  3
```

Explanation

I opened all files, and get description strings using regualr expression. `<p data-testid="vuln-description">(.*?)</p>` is what I used. `(.+?)`, again, means any characters that appears at least one time, and match it non-greedily. I used `re.DOTALL` flag, which means `.` can be matched at `\n`.

Then, I tokenized strings using `nltk.tokenize.word_tokenize()` and attached tags using `nltk.pos_tag()`. If there is no required tools, they will downloaded automatically. I used dictionary to count noun or verb words, and made it one dict using `freq_dict = dict(freq_noun, **freq_verb)`. then I sorted it and printed top 10 .