**Name:** Estiward Casado Antigua

**Date Created:** February 9, 2025

**Program Description:**

This program’s purpose is to scan an email inputted by the user to see how likely it is to be spam. To accomplish this, the program will scan the email for 30 words commonly used in spam emails. For each word that matches the list, a point will be added to the spam score and when the program is finished scanning, it will display whether or not the message is likely to be spam, the spam score, and what words triggered that judgement.

**Functions used in the Program (in order that they are called):**

1. **Function Name:** email\_scanner()

**Description:**

The flag words has 30 elements that a for loop iterates through and checks if the element matches any part of the string. For any match 1 more will be added to the spam score. After going through each element of the list, the total spam score will be returned.

**Parameters:**

* 1. input (str) – email string from main()

**Variables:**

1. spam\_score (int) – accumulator adds 1 every time a flagged word is found
2. found\_words (list) – when a word is found it gets added to
3. flag\_words (list) – list of words and phrases to check
4. word\_count (list) – list that has each repetition of a word

**Logical Steps:**

1. Each variable is initialized
2. A for loop is created that goes through each item in the list.
3. Findall from the real expressions module is used to create a list that has each instance of a particular word.
4. An if statement tests the length of the word\_count list. If the length of the list is 0, then the next iteration of the loop starts. If the length is not zero, then the spam score is added to from the length of the word\_count list. Then, the found\_words list is added to with the current item in the for loop.
5. After the for loop is completed, the spam score and found\_words list are returned

**Returns:** spam\_score, found\_words

1. **Function Name:** main()

**Description:**

The user is asked to input an email message, which is then passed through email\_scanner() function that returns the spam score. After receiving the spam score, the spam score will be displayed along with which words set it off and a message stating how likely the message is to be spam.

**Parameters:** None

**Variables:**

* 1. email (str) – input email from user to pass through email\_scanner()
  2. spam\_score (int) – how many times a flagged word appears in the email
  3. found\_words (list) – list of which specific words were flagged
  4. risk\_level (str) – based on scam score how risky the email is

**Logical Steps:**

1. The user is prompted to enter an email message.
2. The email scanner function is run and spam score and found words list are returned into variables.
3. An if statement is done and depending on the scam score the risk level is set
4. The risk level, spam score, and trigger words are printed.

**Returns:** None

**Logical Steps:**

1. Import real expressions
2. Call main()
3. Inside main(), email\_scanner() is called

**Repository Link:** https://github.com/coppajo/COP2373