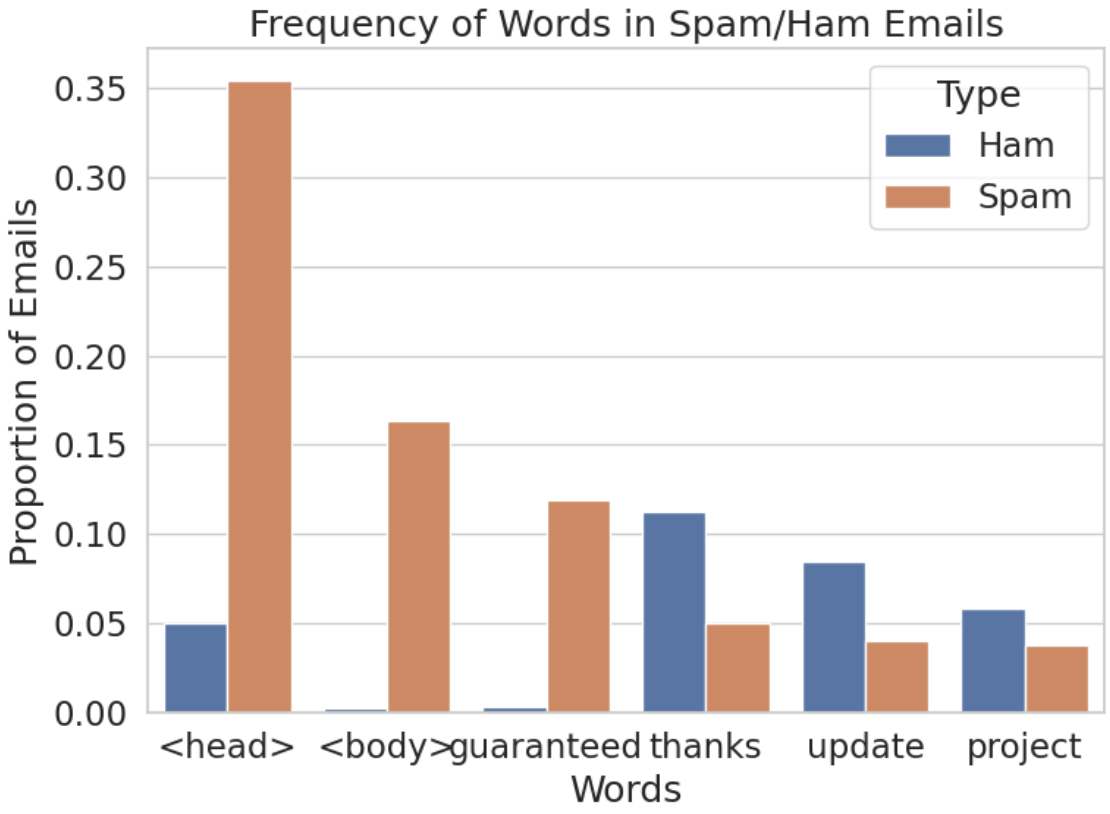
0.1 Question 1

Discuss one attribute or characteristic you notice that is different between the two emails that may allow you to uniquely identify a spam email.

One noticeable characteristic that uniquely identifies the spam email is it's in HTML format with an embedded HTML tag syntax. In contrast, The ham mail is written in plain text without much formatting, and URLS are included as simple links, not as HTML tags.

Create your bar chart in the following cell:

```
In [57]: train = train.reset_index(drop=True) # We must do this in order to preserve the ordering of em

         words_to_check = ['<head>', '<body>', 'guaranteed', 'thanks', 'update', 'project']
         for word in words_to_check:
             train[word] = words_in_texts([word], train['email']).flatten()

         ham_proportions = train[train['spam'] == 0][words_to_check].mean()
         spam_proportions = train[train['spam'] == 1][words_to_check].mean()

         df2 = pd.DataFrame({
             'Words': words_to_check * 2,
             'Proportion of Emails': pd.concat([ham_proportions, spam_proportions]).values,
             'Type': ['Ham'] * len(words_to_check) + ['Spam'] * len(words_to_check)
         })

         plt.figure(figsize=(8,6))
         sns.barplot(data=df2, x='Words', y='Proportion of Emails', hue='Type')
         plt.xlabel('Words')
         plt.ylabel('Proportion of Emails')
         plt.title('Frequency of Words in Spam/Ham Emails')

         plt.tight_layout()
         plt.show()
```


Frequency of Words in Spam/Ham Emails
Type
Ham
Spam
Proportion of Emails
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
<head>
<body>
guaranteed
thanks
update
project
Words

0.2 Question 6c

Explain your results in `q6a` and `q6b`. How did you know what to assign to `zero_predictor_fp`, `zero_predictor_fn`, `zero_predictor_acc`, and `zero_predictor_recall`?

Since this zero predictor will always predict negative (or ham) for every email, there will be no false positive so the zero_predictor_fp 0.

The number of false negatives will be equal to the number of spam emails, i.e. the positives that we falsely classified as negatives.

The accuracy will be equal to the proportion of emails that are actually ham.

The recall is the proportion of actual positives that were predicted as positives. Because this predictor will never predict positive, the recall is 0.

0.3 Question 6f

How does the accuracy of the logistic regression classifier `my_model` compare to the accuracy of the zero predictor?

Comparing the logistic regression classifier and the zero classifier, the logistic classifier has a higher accuracy ($\sim$ 76%) than the zero predictors, which can be calculated by dividing the number of true negatives (TN) by the total number of emails (TP + TN + FP + FN) ($\sim$ 73%).

0.4 Question 6g

Given the word features provided in Question 4, discuss why the logistic regression classifier `my_model` may be performing poorly.

Hint: Think about how prevalent these words are in the email set.

One reason our classifier my_model is poorly performing could be because the input features/words ('drug', 'bank', 'prescription', 'memo', 'private') we selected were more contextually related to the content of the email rather than its formatting. The first spam email in the training data is in HTML formatting. My graph in question 3 highlights that words like "head" and "body", as well as HTML specific symbols like $< >$, are far more common in spam emails compared to the words originally chosen, which were related to themes of money, private information, and urgency (typical signs of scams or spam emails). To improve the classifier's performance, we should prioritize selecting features that reflect the HTML formatting patterns prevalent in spam emails.

0.5 Question 6h

Would you prefer to use the logistic regression classifier `my_model` or the zero predictor classifier for a spam filter? Why? Describe your reasoning and relate it to at least one of the evaluation metrics you have computed so far.

I would prefer to use the zero predictor. Even though its accuracy is not as high as the logistic regression classifier (albeit not by much), it does a better job of accounting for false positives, as I would not want an important ham email falsely classified as spam.