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 attribute I notice that is different between the two emails is their formatting. The ham email is written in plain, readable text, making it look like a typical message. In comparison, the spam email is formatted using HTML syntax, with tags like `<body>` and `<head>`, making it resemble a web page rather than a traditional email. The spam message also includes a suspicious looking hyperlink with an IP address, whereas the ham email links a real website with trustworthy urls.

Create your bar chart in the following cell:

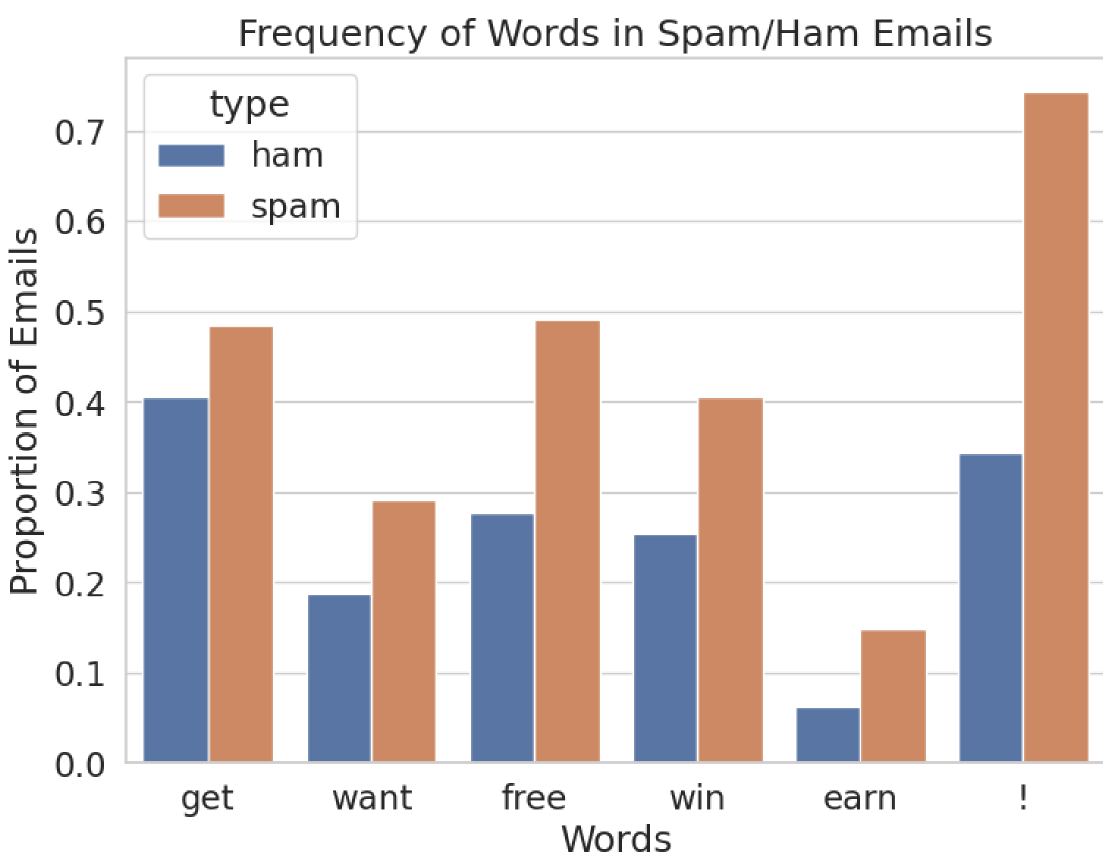
```
In [13]: train = train.reset_index(drop=True) # We must do this in order to preserve the ordering of em
         plt.figure(figsize=(8,6))

         my_words = ['get', 'want', 'free', 'win', 'earn', '!']

         matrix = words_in_texts(my_words, train['email'])

         my_df = pd.DataFrame(data=matrix, columns=my_words)
         my_df['type'] = train['spam'].replace({0: 'ham', 1: 'spam'})
         my_df = my_df.melt(id_vars='type')

         sns.barplot(data=my_df, x='variable', y='value', hue='type', ci=None)
         plt.tight_layout()
         plt.title('Frequency of Words in Spam/Ham Emails')
         plt.xlabel('Words')
         plt.ylabel('Proportion of Emails')
         plt.show()
```



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`?

A false positive happens when the model predicts spam, but the actual label is ham. Since the zero predictor always predicts ham (0) and never predicts spam (1), it will never produce a false positive. Therefore, `zero_predictor_fp` = 0.

A true positive occurs when the model predicts spam, and the actual label is also spam. Since the zero predictor never predicts spam, it will never produce a true positive either. Therefore, `zero_predictor_tp` = 0.

A false negative happens when the model predicts ham, but the actual label is spam. Since the zero predictor always predicts ham, it will misclassify every spam emails as ham. Therefore, `zero_predictor_fn` = number of spam emails in training set.

A true negative occurs when the model predicts ham, and the actual label is also ham. Since the zero predictor always predicts ham, all actual ham emails will be correctly classified. Therefore, `zero_predictor_tn` = total emails - number of spam emails.

Using these values, I can calculate the accuracy of the zero predictor as the ratio of true predictions (true positives and true negatives) to the total number of emails. So, accuracy is calculated as `zero_predictor_acc` = true negatives / total emails.

For recall, I calculate it as the ratio of true positives to the sum of true positives and false negatives. Since there are no true positives in the zero predictor, recall will be `zero_predictor_recall` = 0.

0.3 Question 6f

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

The accuracy of the logstic regression classifier `my_model` is approximately 75.8%, which performs very similarly but slightly better than the zero predictor of around 74.5%.

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.

The logistic regression classifier `my_model` may be performing poorly because the selected word can appear in both spam and ham emails. While some of the words are often associated with spam (ie. 'drug' and 'prescription'), they are not exclusive to spam content. For example, the word 'bank' might appear in spam messages related to phishing, but it could also show up in legitimate emails from financial institutions like Bank of America. This overlap reduces the model's ability to distinguish spam from ham based on these words alone, leading to lower predictive performance.

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 logistic regression classifier `my_model` over the zero predictor classifier for a spam filter. Although both models have similar accuracies, the zero predictor always classifies emails as ham, resulting in a recall of 0. It fails to identify any spam emails. This makes it unrealistic for real-world use where detecting spam is important. While `my_model` has a low recall of around 11%, it still identifies some spam. In the context of spam filtering, it is more important to catch at least some spam content, even if we detect a few false positives.