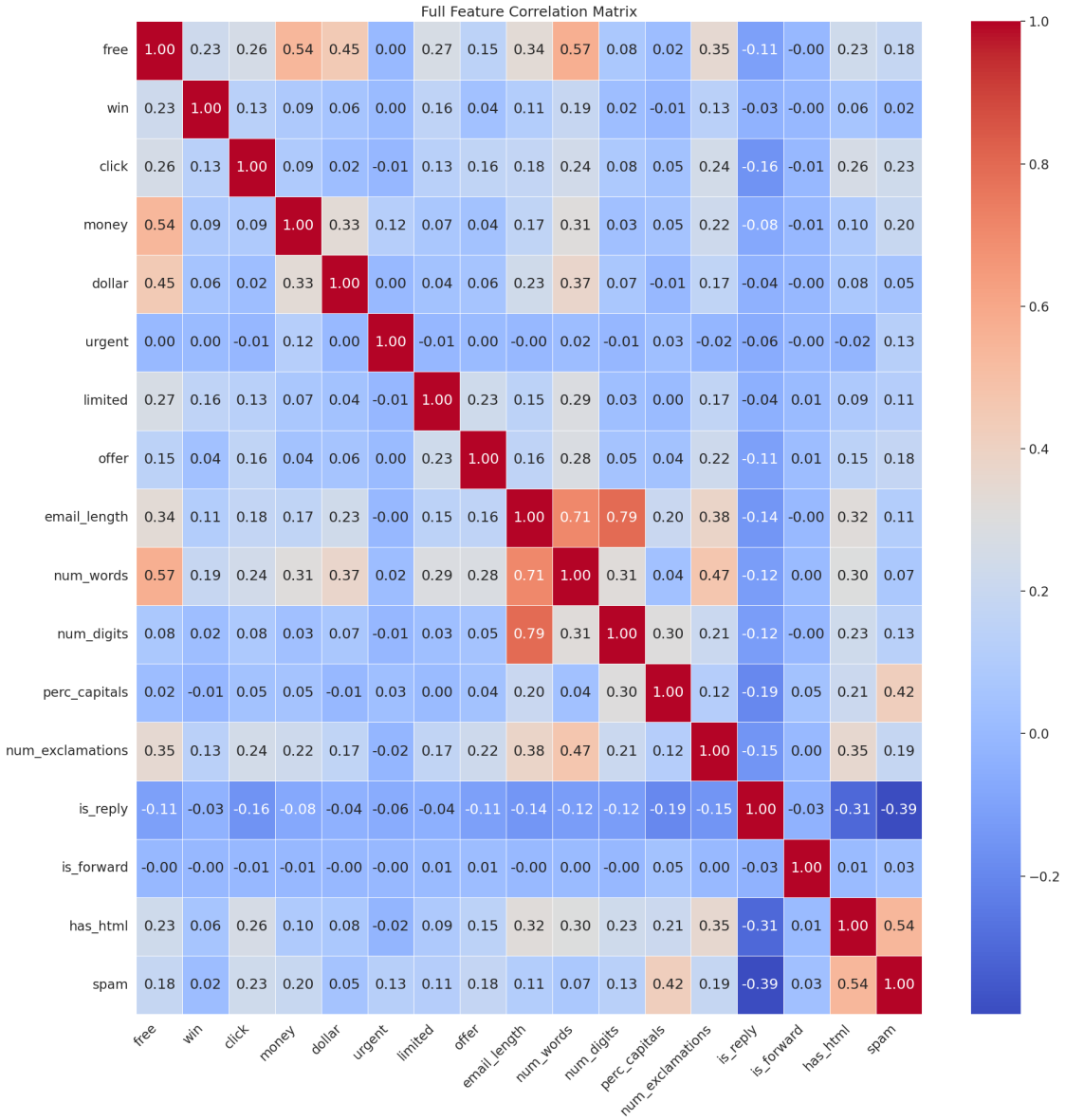
0.1 Question 1a

Generate your visualization in the cell below.

In [56]:
```
with zipfile.ZipFile('spam_ham_data.zip') as item:
    with item.open("train.csv") as f:
        og_training_data= pd.read_csv(f)

og_training_data = og_training_data.fillna('')

from sklearn.model_selection import train_test_split
og_train, og_val = train_test_split(og_training_data, test_size = 0.1, random_state = 42)

# We must do this in order to preserve the ordering of emails to labels for words_in_texts.
og_train = og_train.reset_index(drop = True)

keywords = ['free', 'win', 'click', 'money', 'dollar', 'urgent', 'limited', 'offer']
for word in keywords:
    og_train[word] = og_train['email'].apply(lambda x: x.lower().split().count(word))

og_train['perc_capitals'] = og_train['email'].apply(
    lambda x: sum(1 for c in x if c.isupper()) / len(x) if len(x) > 0 else 0
)

og_train['email_length'] = og_train['email'].apply(len)
og_train['num_digits'] = og_train['email'].apply(lambda x: sum(c.isdigit() for c in x))
og_train['num_exclamations'] = og_train['email'].apply(lambda x: x.count('!'))
og_train['has_html'] = og_train['email'].apply(
    lambda x: int('<html' in x.lower() or '<a ' in x.lower() or '<img' in x.lower() or '<div' i
)
og_train['num_words'] = og_train['email'].apply(lambda x: len(x.split()))

og_train['is_reply'] = og_train['subject'].apply(lambda x: int(x.strip().lower().startswith('su
og_train['is_forward'] = og_train['subject'].apply(lambda x: int(x.strip().lower().startswith(

features = [word for word in keywords] + ['email_length','num_words', 'num_digits', 'perc_capi
correlation_matrix = og_train[features].corr()

plt.figure(figsize=(20, 20))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=0.5)
plt.title("Full Feature Correlation Matrix")
plt.xticks(rotation=45, ha='right')

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


Full Feature Correlation Matrix
free 1.00 0.23 0.26 0.54 0.45 0.00 0.27 0.15 0.34 0.57 0.08 0.02 0.35 -0.11 -0.00 0.23 0.18
win 0.23 1.00 0.13 0.09 0.06 0.00 0.16 0.04 0.11 0.19 0.02 -0.01 0.13 -0.03 -0.00 0.06 0.02
click 0.26 0.13 1.00 0.09 0.02 -0.01 0.13 0.16 0.18 0.24 0.08 0.05 0.24 -0.16 -0.01 0.26 0.23
money 0.54 0.09 0.09 1.00 0.33 0.12 0.07 0.04 0.17 0.31 0.03 0.05 0.22 -0.08 -0.01 0.10 0.20
dollar 0.45 0.06 0.02 0.33 1.00 0.00 0.04 0.06 0.23 0.37 0.07 -0.01 0.17 -0.04 -0.00 0.08 0.05
urgent 0.00 0.00 -0.01 0.12 0.00 1.00 -0.01 0.00 -0.00 0.02 -0.01 0.03 -0.02 -0.06 -0.00 -0.02 0.13
limited 0.27 0.16 0.13 0.07 0.04 -0.01 1.00 0.23 0.15 0.29 0.03 0.00 0.17 -0.04 0.01 0.09 0.11
offer 0.15 0.04 0.16 0.04 0.06 0.00 0.23 1.00 0.16 0.28 0.05 0.04 0.22 -0.11 0.01 0.15 0.18
email_length 0.34 0.11 0.18 0.17 0.23 -0.00 0.15 0.16 1.00 0.71 0.79 0.20 0.38 -0.14 -0.00 0.32 0.11
num_words 0.57 0.19 0.24 0.31 0.37 0.02 0.29 0.28 0.71 1.00 0.31 0.04 0.47 -0.12 0.00 0.30 0.07
num_digits 0.08 0.02 0.08 0.03 0.07 -0.01 0.03 0.05 0.79 0.31 1.00 0.30 0.21 -0.12 -0.00 0.23 0.13
perc_capitals 0.02 -0.01 0.05 0.05 -0.01 0.03 0.00 0.04 0.20 0.04 0.30 1.00 0.12 -0.19 0.05 0.21 0.42
num_exclamations 0.35 0.13 0.24 0.22 0.17 -0.02 0.17 0.22 0.38 0.47 0.21 0.12 1.00 -0.15 0.00 0.35 0.19
is_reply -0.11 -0.03 -0.16 -0.08 -0.04 -0.06 -0.04 -0.11 -0.14 -0.12 -0.12 -0.19 -0.15 1.00 -0.03 -0.31 -0.39
is_forward -0.00 -0.00 -0.01 -0.01 -0.00 -0.00 0.01 0.01 -0.00 0.00 -0.00 0.05 0.00 -0.03 1.00 0.01 0.03
has_html 0.23 0.06 0.26 0.10 0.08 -0.02 0.09 0.15 0.32 0.30 0.23 0.21 0.35 -0.31 0.01 1.00 0.54
spam 0.18 0.02 0.23 0.20 0.05 0.13 0.11 0.18 0.11 0.07 0.13 0.42 0.19 -0.39 0.03 0.54 1.00
free win click money dollar urgent limited offer email_length num_words num_digits perc_capitals num_exclamations is_reply is_forward has_html spam
1.0 0.8 0.6 0.4 0.2 0.0 −0.2

0.2 Question 1b

In two to three sentences, describe what you plotted and its implications with respect to your features.

I have plotted a heatmap showing the correlation among all of my chosen features against each other as well as their correlation with whether an email is spam. My plot suggests that of the features I've chosen, whether the email contains html and the percentage of capital letters in the email are moderately correlated with spam emails, while whether or not the email is a reply is moderately correlated with ham emails. Other features I've selected are shown as being relatively week correlators with spam, but may be benefical when incorporated together.

1 Question 4

Describe the process of improving your model. You should use at least 2-3 sentences each to address the following questions:

1. How did you find better features for your model?
2. What did you try that worked or didn't work?
3. What was surprising in your search for good features?

Look at my correlation heatmap, I only selected features that showed at least a .15 correlation with spam emails. I found it somewhat surprising that features like frequency of the word 'win' and whether or not the email was a forward weren't stronger indicators of spam. Mostly, I just did a bit of trial and error adding and removing features and seeing how that affected the accuracy of my training set's ability to predict the validation set.

2 Question 5: ROC Curve

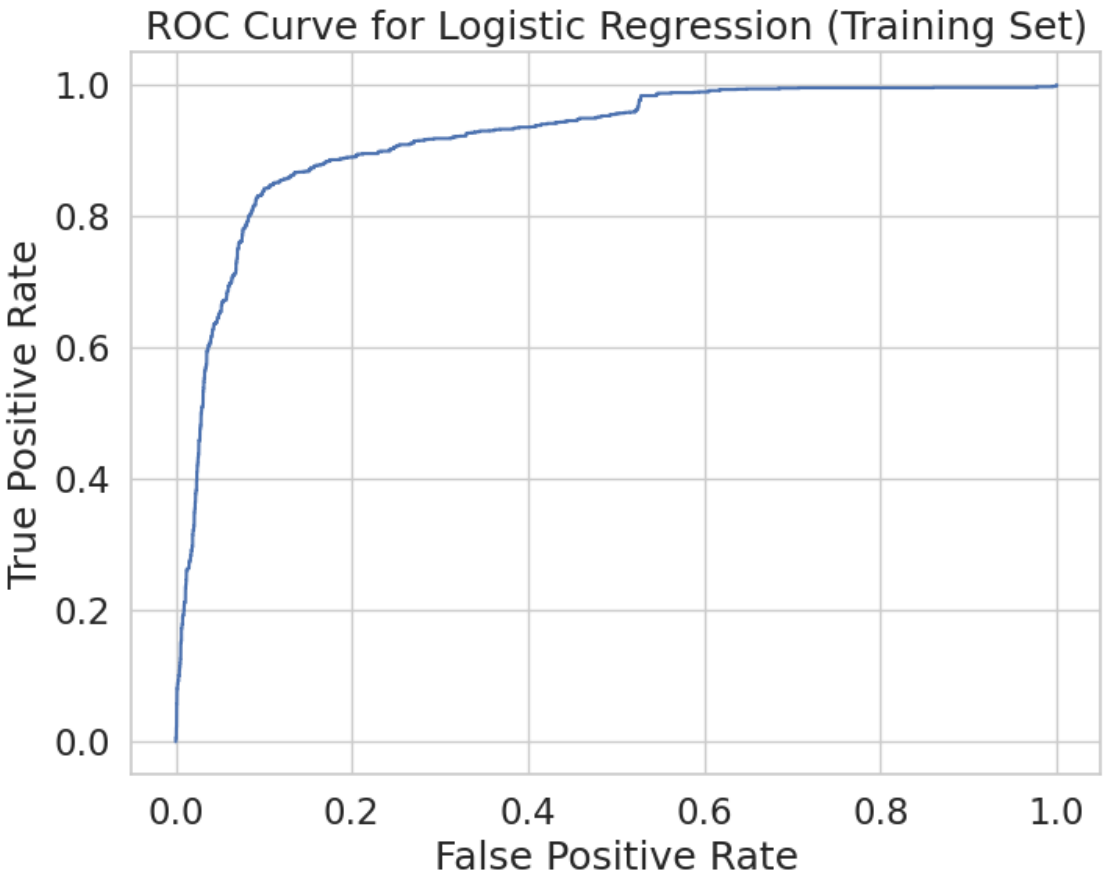
In most cases, we won't be able to get 0 false positives and 0 false negatives, so we have to compromise. For example, in the case of cancer screenings, false negatives are comparatively worse than false positives — a false negative means that a patient might not discover that they have cancer until it's too late. In contrast, a patient can receive another screening for a false positive.

Recall that logistic regression calculates the probability that an example belongs to a particular class. To classify an example, we say that an email is spam if our classifier gives it ≥ 0.5 probability of being spam. However, **we can adjust that cutoff threshold**. We can say that an email is spam only if our classifier gives it ≥ 0.7 probability of being spam, for example. This is how we can trade off false positives and false negatives.

The Receiver Operating Characteristic (ROC) curve shows this trade-off for each possible cutoff probability. In the cell below, plot an ROC curve for your final classifier (the one you use to make predictions for Gradescope) on the training data. Lecture 23 may be helpful.

Hint: You'll want to use the `.predict_proba` method (documentation) for your classifier instead of `.predict` to get probabilities instead of binary predictions.

```
In [67]: train_probabilities = model.predict_proba(X_train)[:, 1]

         fpr, tpr, thresholds = roc_curve(y_train, train_probabilities)

         plt.figure(figsize=(8, 6))
         plt.plot(fpr, tpr, label="ROC Curve")
         plt.xlabel("False Positive Rate")
         plt.ylabel("True Positive Rate")
         plt.title("ROC Curve for Logistic Regression (Training Set)")
         plt.grid(True)
         plt.show()
```


ROC Curve for Logistic Regression (Training Set)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

2.0.1 Question 6a

Pick at least **one** of the emails provided above to comment on. How would you classify the email (e.g., spam or ham), and does this align with the classification provided in the training data? What could be a reason someone would disagree with *your* classification of the email? In 2-3 sentences, explain your perspective and potential reasons for disagreement.

I will pick example 1. I would classify this email as ham, which does not align with the classification provided in the training data. A likely reason someone would disagree with my classification of the email is either due to the excessive use of '!' as well as having a mailing list footer and link. I would consider this to be ham due to the personalized content of the email indicating that the sender has an actual relation with the recipient.

2.0.2 Question 6b

As data scientists, we sometimes take the data to be a fixed "ground truth," establishing the "correct" classification of emails. However, as you might have seen above, some emails can be ambiguous; people may disagree about whether an email is actually spam or ham. How does the ambiguity in our labeled data (spam or ham) affect our understanding of the model's predictions and the way we measure/evaluate our model's performance?

Ambiguity in labeled data challenges the assumption that there is a fixed "ground truth". It introduces noise in the form of false positives or false negatives in the training data, which will result in a less accurate model but at no fault of the methods or features selected to create that model. In other words, ambiguously labeled data affects our understanding of the model's predictions because we may change features to accomodate lower than expected accuracy, but the error introduced by the ambiguous data cannot be corrected for in any way other than establishing a more clear definition of what it means to be spam.

Part ii Please provide below the index of the email that you flipped classes (`email_idx`). Additionally, in 2-3 sentences, explain why you think the feature you chose to remove changed how your email was classified.

The index of the email I flipped classes was 352. I think the feature I chose, 'private', changed how the email was classified because it was one of 3 total features present in email 352 that contributed to it being labeled as spam. With 5 total features in the original model, it makes sense that an email with 3/5 of those features would be labeled as spam, while it's more likely that an email with 2/4 of the remaining features is now less confidently classified as ham.

Part i In this context, do you think you could easily find a feature that could change an email's classification as you did in part a)? Why or why not?

In this context, no I do not think I could find a feature that could change an email's classification like in part a). This would be because of the fact that in a model containing 1000 features, each feature likely contributes an average of a fraction of a percent of certainty that an email is spam or ham, so the removal of any given feature would likely only change the probability of a correct classification by <1%, which is usually not enough to change the classification of most emails. This however is not to say it's impossible. Depending on the threshold of certainty you would like to have before classifying as spam or ham, if a particular email is within .1% of that threshold, then there is a decent chance the change of a single feature would result in a change of that email's classification.

Part ii Would you expect this new model to be more or less interpretable than `simple_model`?

Note: A model is considered interpretable if you can easily understand the reasoning behind its predictions and classifications. For example, the model we saw in part a), `simple_model`, is considered interpretable as we can identify which features contribute to an email's classification.

I would expect the new model to be less interpretable than simple model, since it's harder to understand each feature's contribution to classifying a variable when there are just so many of them. With 1000 features, the effect of any single feature is usually small and spread out, making it difficult to pinpoint which features are driving the prediction.

2.0.3 Question 7c

Now, imagine you're a data scientist at Meta, developing a text classification model to decide whether to remove certain posts / comments on Facebook. In particular, you're primarily working on moderating the following categories of content: * Hate speech * Misinformation * Violence and incitement

Pick one of these types of content to focus on (or if you have another type you'd like to focus on, feel free to comment on that!). What content would fall under the category you've chosen? Refer to Facebook's Community Standards, which outline what is and isn't allowed on Facebook.

I would focus on misinformation. From Facebook's community standards, I would say that Fraud, Scams, and Deceptive Practices, Authentic Identity Representation, Inauthentic Behavior, and Misinformation to fall under the category of misinformation.

2.0.4 Question 7d

What are the stakes of misclassifying a post in the context of a social media platform? Comment on what a false positive and false negative means for the category of content you've chosen (hate speech, misinformation, or violence and incitement).

The stakes of misclassifying misinformation would result in false positives which removes factually correct information from the platform disenfranchising users from benefiting from the sharing of knowledge, as well as false negatives which open opportunities for users to be swayed or convinced of untrue facts and beliefs by not accurately removing posts containing misinformation.

2.0.5 Question 7e

As a data scientist, why might having an interpretable model be useful when moderating content online?

As a data scientist, an interpretable model would be useful when moderating content online for a couple main reasons. Firstly, anyone who wants to fine tune or make adjustments to the model for any number of reasons which may include the changing of content moderating policies on the platform as a whole, can do so more effeciently and easily. Second, an interpretable model is important in being able to explain to both users and moderators why a certain post or piece of content was flagged or removed, without having to run a complex case by case analysis.