```
Start coding or generate with AI.
```

```
import pandas as pd

# Load the Excel file into a pandas DataFrame
df = pd.read_excel("/content/Natural Hair in The Bahamas (Females) (Responses).xlsx")

# Display the first few rows of the DataFrame
display(df.head())

# Display information about the DataFrame
display(df.info())
```

	What is your hair type? (See above photo for reference)	What's your race?	What's your occupation?	Have you ever been bullied/judged for your hair?	If your answer was yes to the previous question, what was the reason?	Explain a situation in which you were bullied/judged for your hair?	How many times have you been bullied/judged for your hair?	How did you feel when you were bullied/judged for your hair?	What do you think was the reason behind you being bullied/judged?	Do y belie t treatme of a natur hair T Baham is equ and fai Expla yo answe
0	4a	Black	College student	Yes	Hair type	I'd say probably when I was young, (don't real...	1-2 times	Terrible	Social norms - shared standards of acceptable ...	Not rea I feel thou femal with 4b
1	4c	Black	College student	Yes	Hair type, Length, Style	While in a hair salon, the hair stylist called...	Most of my life	It made me very insecure about my hair and I f...	Harrasment - the act of systematic and/or cont...	Not at Girls w loos curls often
2	4c	Black	College student	Yes	Style	I was being serve at my high school by the lun...	Not able to count	Annoyed	Social norms - shared standards of acceptable ...	Not at Baham pre loos cu becaus
3	4c	Black	College student	No	NaN	NaN	1-2 times	NaN	jealousy and wickedddness	N
4	4c	Black	College student	Yes	Length	I did the big chop (cutting off all of my perm...	1-2 times	I felt less confident in myself.	Colorism - prejudice or discrimination against...	becau althou we are bla major

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16 entries, 0 to 15
Data columns (total 11 columns):
 #   Column
---  ------
 0   What is your hair type? (See above photo for reference)
 1   What's your race?
 2   What's your occupation?
 3   Have you ever been bullied/judged for your hair?
```

```
import re

# Select the relevant text columns
text_columns = [
```

```
    'If your answer was yes to the previous question, what was the reason?',
    'Explain a situation in which you were bullied/judged for your hair?',
    'How did you feel when you were bullied/judged for your hair?',
    'What do you think was the reason behind you being bullied/judged?',
    'Do you believe the treatment of all natural hair in The Bahamas is equal and fair? Explain your answer.',
    'Do you think society in The Bahamas is changing its view of natural hair for the better or worst? Based on your answer, how is it visib
]

df_text = df[text_columns].copy()

# Handle missing values by filling with an empty string
df_text = df_text.fillna('')

# Combine the text from the selected columns into a single column
df_text['combined_text'] = df_text.apply(lambda row: ' '.join(row.values.astype(str)), axis=1)

# Clean the combined text: remove special characters and convert to lowercase
df_text['cleaned_text'] = df_text['combined_text'].apply(lambda x: re.sub(r'[^a-zA-Z0-9\s]', '', x.lower()))

# Display the first few rows of the DataFrame with the new columns
display(df_text.head())
```

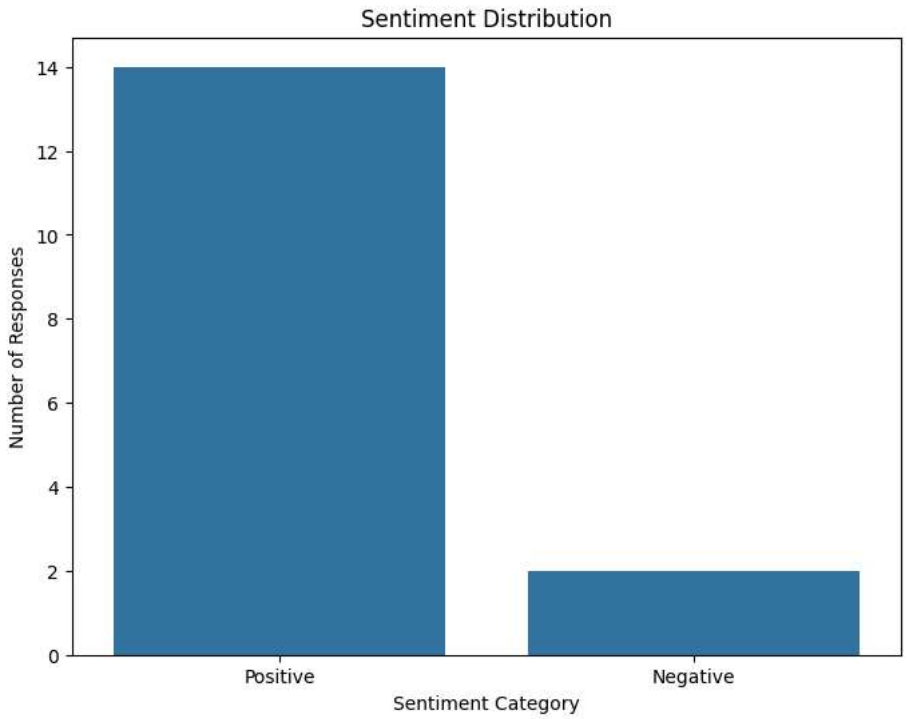
	If your answer was yes to the previous question, what was the reason?	Explain a situation in which you were bullied/judged for your hair?	How did you feel when you were bullied/judged for your hair?	What do you think was the reason behind you being bullied/judged?	Do you believe the treatment of all natural hair in The Bahamas is equal and fair? Explain your answer.	Do you think society in The Bahamas is changing its view of natural hair for the better or worst? Based on your answer, how is it visible in everyday life?	combined_text	cleaned_text
0	Hair type	I'd say probably when I was young, (don't real...	Terrible	Social norms - shared standards of acceptable ...	Not really. I feel as though females with 4b 4...	Honestly, I think they are changing for the be...	Hair type I'd say probably when I was young, (...	hair type id say probably when i was young don...
1	Hair type, Length, Style	While in a hair salon, the hair stylist called...	It made me very insecure about my hair and I f...	Harrasment - the act of systematic and/or cont...	Not at all. Girls with looser curls are often ...	I believe natural hair is becoming more accept...	Hair type, Length, Style While in a hair salon...	hair type length style while in a hair salon t...
2	Style	I was being serve at my high school by the lun...	Annoyed	Social norms - shared standards of acceptable ...	Not at all. Bahamas prefer looser curls becaus...	They probably are but you cannot see it as the...	Style I was being serve at my high school by t...	style i was being serve at my high school by t...

```
# Analyze the distribution of sentiment categories
sentiment_distribution = df_text['sentiment'].value_counts()

# Display the distribution
display(sentiment_distribution)
```

sentiment	count
Positive	14
Negative	2

dtype: int64

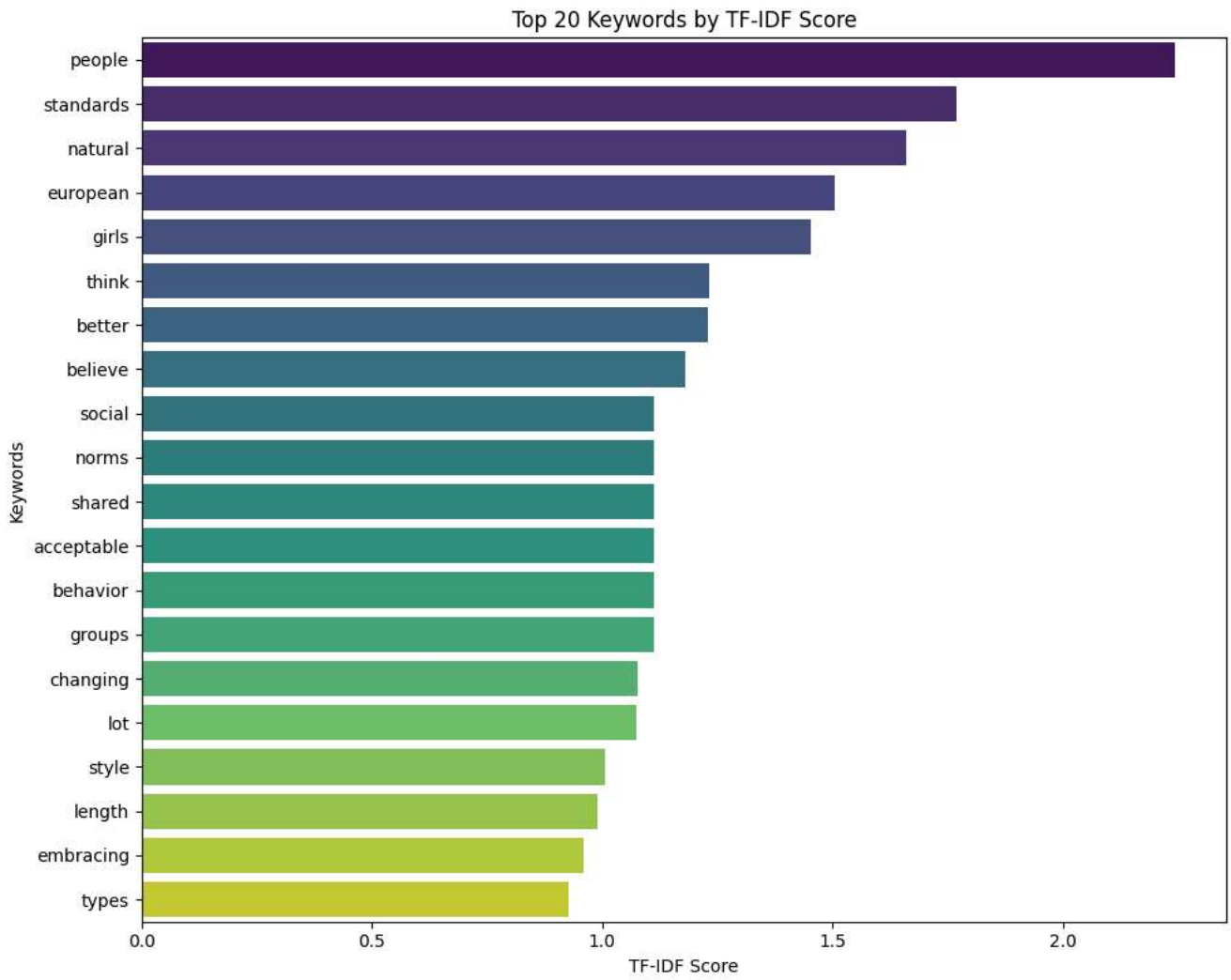
```
import matplotlib.pyplot as plt
import seaborn as sns

# Visualize the sentiment distribution
plt.figure(figsize=(8, 6))
sns.barplot(x=sentiment_distribution.index, y=sentiment_distribution.values)
plt.title('Sentiment Distribution')
plt.xlabel('Sentiment Category')
plt.ylabel('Number of Responses')
plt.show()
```




```
from sklearn.feature_extraction.text import TfidfVectorizer

# Initialize TF-IDF Vectorizer
# Consider only words that appear in at least 2 documents and in no more than 85% of the documents
tfidf_vectorizer = TfidfVectorizer(max_df=0.85, min_df=2, stop_words='english')

# Fit and transform the cleaned text data
tfidf_matrix = tfidf_vectorizer.fit_transform(df_text['cleaned_text'])

# Get feature names (words)
feature_names = tfidf_vectorizer.get_feature_names_out()

# Sum TF-IDF scores for each word across all documents
sum_tfidf = tfidf_matrix.sum(axis=0)

# Create a dictionary of words and their summed TF-IDF scores
tfidf_scores = [(word, sum_tfidf[0, idx]) for word, idx in tfidf_vectorizer.vocabulary_.items()]

# Sort the words by their TF-IDF scores in descending order
tfidf_scores = sorted(tfidf_scores, key=lambda x: x[1], reverse=True)

# Display the top 20 keywords
print("Top 20 Keywords:")
for word, score in tfidf_scores[:20]:
    print(f"{word}: {score:.4f}")
```

```
Top 20 Keywords:
people: 2.2425
standards: 1.7675
natural: 1.6595
european: 1.5053
girls: 1.4527
think: 1.2318
better: 1.2297
believe: 1.1806
social: 1.1114
norms: 1.1114
shared: 1.1114
acceptable: 1.1114
behavior: 1.1114
groups: 1.1114
changing: 1.0773
lot: 1.0740
style: 1.0055
length: 0.9908
embracing: 0.9587
types: 0.9257
```

```
# Get the top 20 keywords and their scores for visualization
top_keywords = tfidf_scores[:20]
words = [item[0] for item in top_keywords]
scores = [item[1] for item in top_keywords]

# Create a bar chart of the top keywords
plt.figure(figsize=(10, 8))
sns.barplot(x=scores, y=words, palette='viridis', hue=words, legend=False)
plt.title('Top 20 Keywords by TF-IDF Score')
plt.xlabel('TF-IDF Score')
plt.ylabel('Keywords')
plt.tight_layout()
plt.show()
```




```
# Generate a summary report
summary = f"""
## Analysis Summary

**Sentiment Analysis:**
The sentiment analysis using VADER shows that out of {len(df_text)} responses, {sentiment_distribution.get('Positive', 0)} were classified a

**Keyword Analysis:**
The top 20 keywords identified through TF-IDF analysis are:
"""

for word, score in tfidf_scores[:20]:
    summary += f"- {word}: {score:.4f}\n"

summary += """

**Observations:**
Based on the sentiment distribution, the majority of the responses were classified as positive by the VADER model. However, it is important
"""
```

```
print(summary)
```

```
## Analysis Summary

**Sentiment Analysis:**
The sentiment analysis using VADER shows that out of 16 responses, 14 were classified as Positive and 2 as Negative. There were 0 Neutra

**Keyword Analysis:**
The top 20 keywords identified through TF-IDF analysis are:
- people: 2.2425
- standards: 1.7675
- natural: 1.6595
- european: 1.5053
- girls: 1.4527
- think: 1.2318
- better: 1.2297
- believe: 1.1806
- social: 1.1114
- norms: 1.1114
- shared: 1.1114
- acceptable: 1.1114
- behavior: 1.1114
- groups: 1.1114
- changing: 1.0773
- lot: 1.0740
- style: 1.0055
- length: 0.9908
- embracing: 0.9587
- types: 0.9257


**Observations:**
Based on the sentiment distribution, the majority of the responses were classified as positive by the VADER model. However, it is import
```

```
# Generate a summary report
summary = f"""
## Analysis Summary

**Sentiment Analysis:**
The sentiment analysis using VADER shows that out of {len(df_text)} responses, {sentiment_distribution.get('Positive', 0)} were classified a

**Keyword Analysis:**
The top 20 keywords identified through TF-IDF analysis are:
"""

for word, score in tfidf_scores[:20]:
    summary += f"- {word}: {score:.4f}\n"

summary += """

**Observations:**
Based on the sentiment distribution, the majority of the responses were classified as positive by the VADER model. However, it is important
"""

print(summary)
```

```
## Analysis Summary

**Sentiment Analysis:**
The sentiment analysis using VADER shows that out of 16 responses, 14 were classified as Positive and 2 as Negative. There were 0 Neutra

**Keyword Analysis:**
The top 20 keywords identified through TF-IDF analysis are:
- people: 2.2425
- standards: 1.7675
- natural: 1.6595
- european: 1.5053
- girls: 1.4527
- think: 1.2318
- better: 1.2297
- believe: 1.1806
- social: 1.1114
- norms: 1.1114
- shared: 1.1114
- acceptable: 1.1114
- behavior: 1.1114
```

- groups: 1.1114
- changing: 1.0773
- lot: 1.0740
- style: 1.0055
- length: 0.9908
- embracing: 0.9587
- types: 0.9257

Observations:
Based on the sentiment distribution, the majority of the responses were classified as positive by the VADER model. However, it is import