BEATLES LYRICS ANALİZİ

In [25]:
```
import pandas as pd
import re
from sklearn.feature_extraction.text import CountVectorizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
import matplotlib.pyplot as plt
from wordcloud import WordCloud
```

In [26]:
```
data = pd.read_csv("C:/Users/enesd/Downloads/Beatles-Lyrics.csv")
```

Verimizi çektik şimdi işlemlere başlayalım.

In [27]:
```
data
```

Out[27]:

	ALBUM	SONG	LYRICS	EARLY_LATE	COMPOSER	YEAR
0	A. Please Please Me	A Taste of Honey	A taste of honey... tasting much sweeter than ...	Early	McCartney	1963
1	A. Please Please Me	Anna (Go To Him)	Anna, You come and ask me, girl, To set you fr...	Early	Lennon	1963
2	A. Please Please Me	Ask Me Why	I love you Can't you tell me things I want to ...	Early	Lennon	1963
3	A. Please Please Me	Baby It's You	Sha la la la la la la Sha la la la la la la...	Early	Lennon	1963
4	A. Please Please Me	Boys	I been told when a boy kiss a girl, Take a tri...	Early	Starr	1963
...	...	...	...	...	...	...
181	M. Let It Be	Let It Be	When I find myself in time of trouble Mother M...	Late	McCartney	1970
182	M. Let It Be	Maggie Mae	Oh, dirty Maggie Mae they have taken her away ...	Late	Lennon/McCartney	1970
183	M. Let It Be	One After 909	My baby says she's trav'ling on the One after ...	Late	Lennon/McCartney	1970
184	M. Let It Be	The Long And Winding Road	The long and winding road that leads to your d...	Late	McCartney	1970
185	M. Let It Be	Two Of Us	Two of us riding nowhere Spending someone's ha...	Late	Lennon/McCartney	1970

186 rows × 6 columns

In [28]:
```
lyrics = data['LYRICS'].dropna().tolist()
```

İçerisinde söz bulunmayan şarkıları çıkarttık, Şarkı sözlerini aldık.

In [29]:
```
lyrics = [lyric.lower() for lyric in lyrics]
```

Küçük harflere dönüştürelim

In [30]:
```
lyrics = [re.sub(r'[^\w\s]', '', lyric) for lyric in lyrics]
```

Noktalama işaretlerini sildik.

In [31]:
```
lyrics = [re.sub(r'\d+', '', lyric) for lyric in lyrics]
```

Sayıları sildik. şimdi stopword kısmına geçelim. Burda kendi isteğimle analizin kalitesini arttırmak için internetten bulduğum stopwordleri ekledim.

In [32]:
```
stop_words = set(stopwords.words('english'))
additional_stop_words = ["0o", "0s", "3a", "3b", "3d", "6b", "6o", "a", "a1", "a2", "a3", "a4", "ab", "able", "
stop_words.update(additional_stop_words)
```

Stop word ifadelerini silme ve lemmatizasyon yapma.

In [33]:
```
lemmatizer = WordNetLemmatizer()
filtered_lyrics = []
for lyric in lyrics:
    tokenized_lyric = word_tokenize(lyric)
    filtered_lyric = [lemmatizer.lemmatize(word) for word in tokenized_lyric if word.lower() not in stop_words]
    filtered_lyrics.append(filtered_lyric)
```

Terim frekanslarını hesaplama.

In [34]:
```
vectorizer = CountVectorizer()
word_counts = vectorizer.fit_transform([' '.join(lyric) for lyric in filtered_lyrics])
words = vectorizer.get_feature_names()
word_counts_df = pd.DataFrame(word_counts.toarray(), columns=words)
```

```
C:\Users\enesd\anaconda3\lib\site-packages\sklearn\utils\deprecation.py:87: FutureWarning: Function get_feature
_names is deprecated; get_feature_names is deprecated in 1.0 and will be removed in 1.2. Please use get_feature
_names_out instead.
  warnings.warn(msg, category=FutureWarning)
```

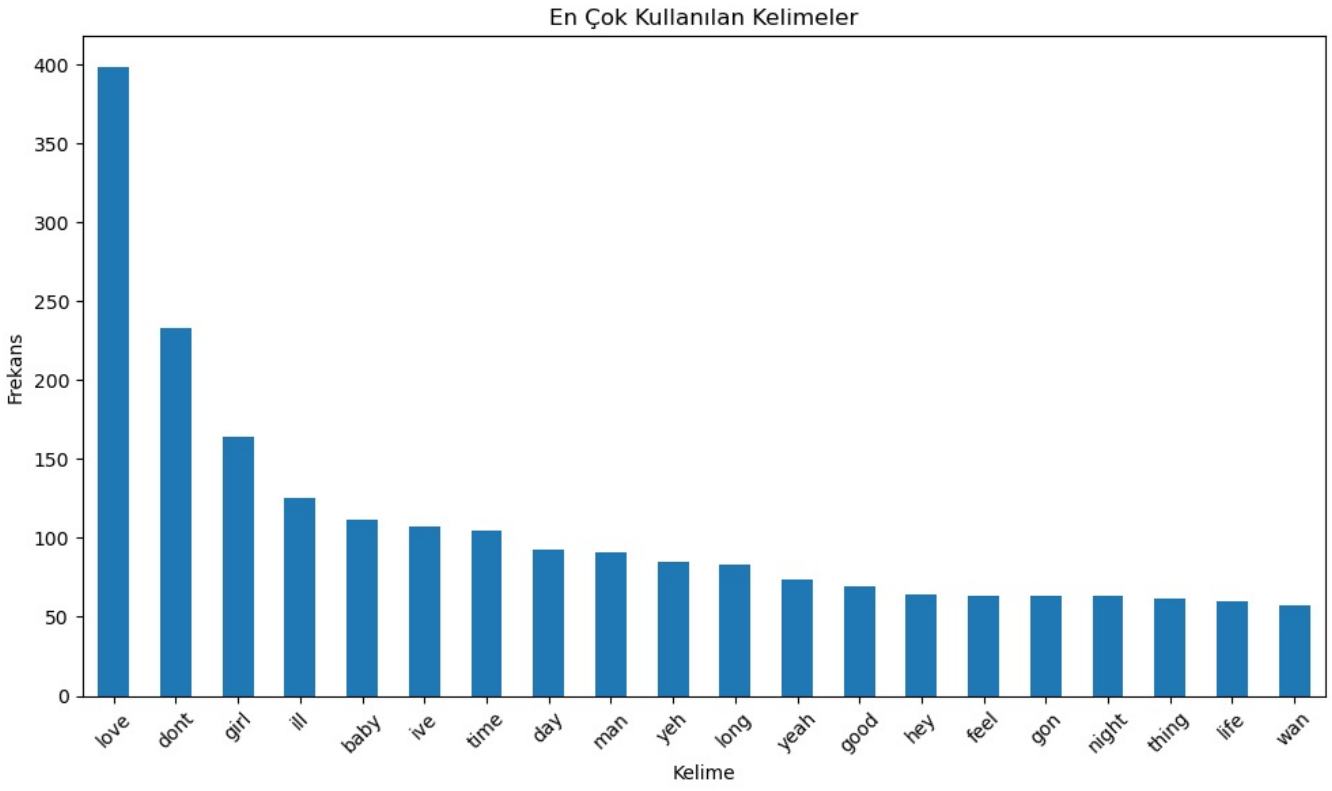
En çok kullanılan kelimeleri gösterelim

In [35]:
```
top_words = word_counts_df.sum().sort_values(ascending=False).head(20)
print(top_words)
```

```
love     398
dont     233
girl     164
ill      125
baby     112
ive      107
time     105
day       93
man       91
yeh       85
long      83
yeah      74
good      69
hey       64
feel      63
gon       63
night     63
thing     62
life      60
wan       57
dtype: int64
```

Barplot

In [36]:
```
plt.figure(figsize=(10, 6))
top_words.plot(kind='bar')
plt.title('En Çok Kullanılan Kelimeler')
plt.xlabel('Kelime')
plt.ylabel('Frekans')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



Word cloud

In [37]:
```
total_word_counts = word_counts_df.sum().sort_values(ascending=False)

wordcloud = WordCloud(width=800, height=400, background_color='white').generate_from_frequencies(total_word_cou
```

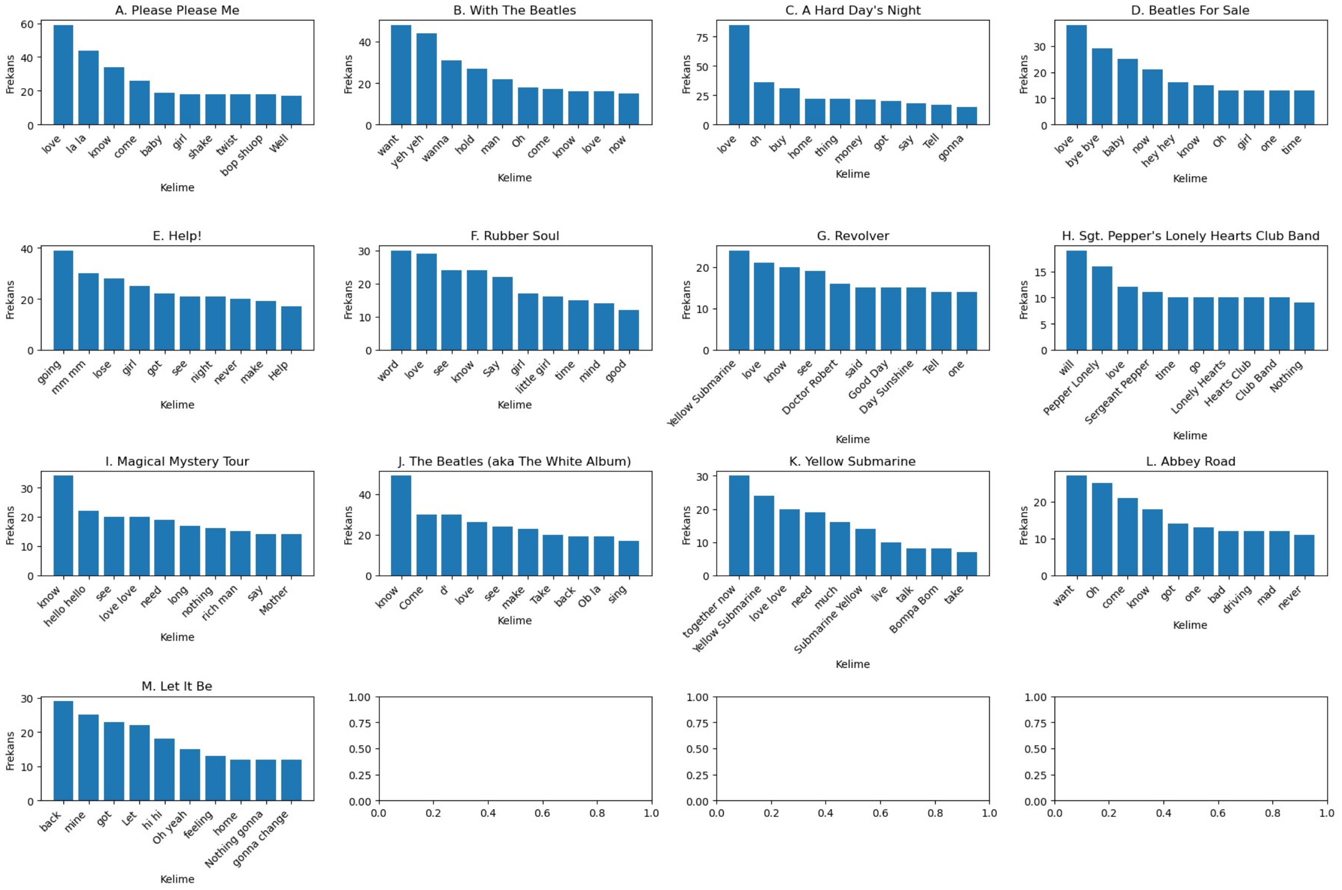
```
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

Albümlere göre en çok kullanılan şarkı sözleri

In [38]:

```
df.dropna(subset=['LYRICS', 'ALBUM'], inplace=True)

lyrics = df['LYRICS'].values
albums = df['ALBUM'].values

lyrics_by_album = {}
for album, lyric in zip(albums, lyrics):
    if album not in lyrics_by_album:
        lyrics_by_album[album] = ''
    lyrics_by_album[album] += lyric + ' '

top_words_by_album = {}
for album, lyric in lyrics_by_album.items():
    wordcloud = WordCloud(width=800, height=400, background_color='white')
    word_frequencies = wordcloud.process_text(lyric)
    top_words = sorted(word_frequencies.items(), key=lambda x: x[1], reverse=True)[:10]
    top_words_by_album[album] = top_words

fig, axes = plt.subplots(nrows=4, ncols=4 ,figsize=(18, 12))
axes = axes.flatten()
for i, (album, top_words) in enumerate(top_words_by_album.items()):
    ax = axes[i]
    words, frequencies = zip(*top_words)
    ax.bar(words, frequencies)
    ax.set_title(album)
    ax.set_xlabel('Kelime')
    ax.set_ylabel('Frekans')
    ax.set_xticklabels(words, rotation=45, ha='right')

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

```
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\3848984094.py:28: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
```

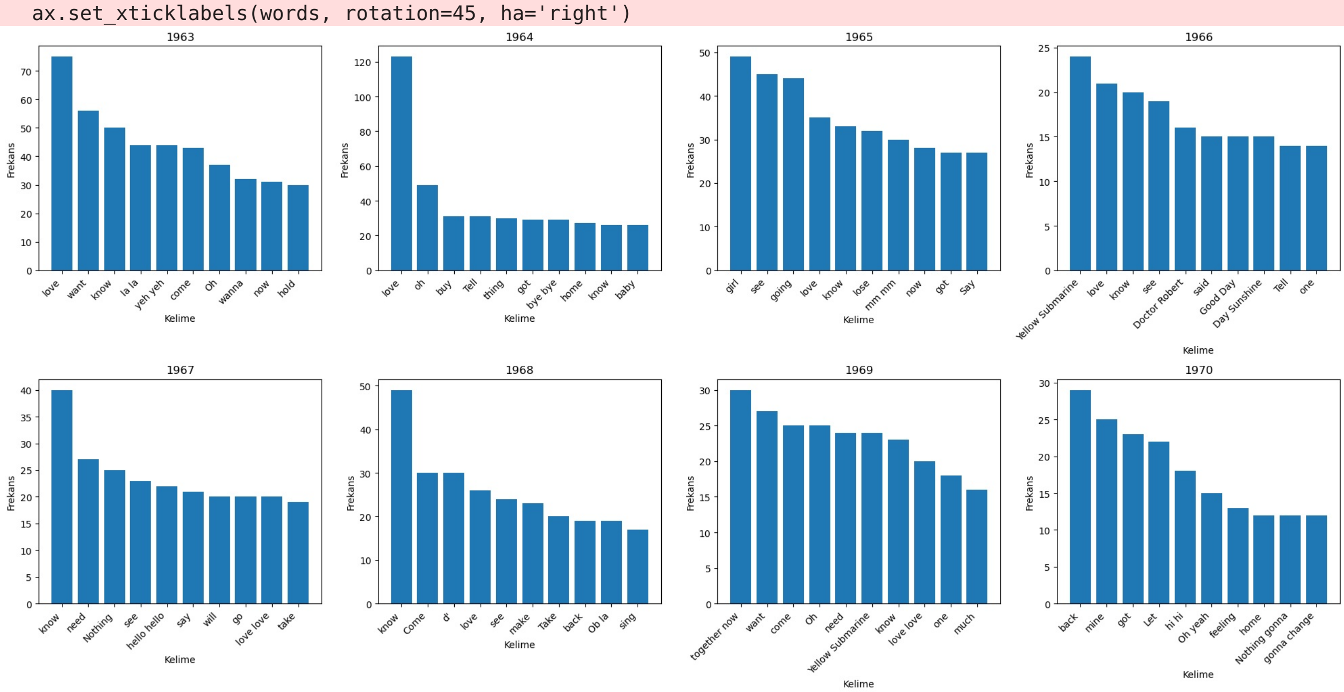


Yıllara göre en çok kullanılan şarkı sözleri

In [39]:
```python
df = data.copy()

df.dropna(subset=['LYRICS', 'YEAR'], inplace=True)

lyrics = df['LYRICS'].values
years = df['YEAR'].values

lyrics_by_year = {}
for year, lyric in zip(years, lyrics):
```

```
    if year not in lyrics_by_year:
        lyrics_by_year[year] = ''
    lyrics_by_year[year] += lyric + ' '

top_words_by_year = {}
for year, lyric in lyrics_by_year.items():
    wordcloud = WordCloud(width=800, height=400, background_color='white')
    word_frequencies = wordcloud.process_text(lyric)
    top_words = sorted(word_frequencies.items(), key=lambda x: x[1], reverse=True)[:10]
    top_words_by_year[year] = top_words

fig, axes = plt.subplots(nrows=2, ncols=4, figsize=(20, 10))
axes = axes.flatten()

for i, (year, top_words) in enumerate(top_words_by_year.items()):
    ax = axes[i]
    words, frequencies = zip(*top_words)
    ax.bar(words, frequencies)
    ax.set_title(str(year))
    ax.set_xlabel('Kelime')
    ax.set_ylabel('Frekans')
    ax.set_xticklabels(words, rotation=45, ha='right')

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

```
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
C:\Users\enesd\AppData\Local\Temp\ipykernel_12336\384367014.py:31: UserWarning: FixedFormatter should only be u
sed together with FixedLocator
  ax.set_xticklabels(words, rotation=45, ha='right')
```



Bestecinin kim olduğuna göre en çok kullanılan kelime dataframe'i

In [40]:
```
composers = df['COMPOSER'].values
lyrics = df['LYRICS'].values

lyrics_by_composer = {}
for composer, lyric in zip(composers, lyrics):
    if composer not in lyrics_by_composer:
        lyrics_by_composer[composer] = ''
    lyrics_by_composer[composer] += lyric + ' '
```

```
most_common_words_by_composer = {}
for composer, lyric in lyrics_by_composer.items():
    wordcloud = WordCloud(width=800, height=400, background_color='white')
    word_frequencies = wordcloud.process_text(lyric)
    most_common_word = sorted(word_frequencies.items(), key=lambda x: x[1], reverse=True)[0][0]
    most_common_words_by_composer[composer] = most_common_word

df_most_common_words = pd.DataFrame.from_dict(most_common_words_by_composer, orient='index', columns=['Most Com
df_most_common_words.index.name = 'Composer'

print(df_most_common_words)
```

```
                              Most Common Word
Composer
McCartney                                love
Lennon                                   know
Starr                         Yellow Submarine
Harrison                                 love
Lennon/McCartney                         love
Lennon/Harrison                          hold
Lennon/McCartney/Harrison/Starr   Sergeant Pepper
Lennon/McCartney/Harrison                   Ah
```

Son olarak da yıllara göre beatles'ın çıkarttığı şarkıları bir grafikle inceliyoruz ve projemizi tamamlıyoruz.

In [41]:
```
df['YEAR'] = pd.to_datetime(df['YEAR'], format='%Y')

song_count = df['YEAR'].dt.year.value_counts().sort_index()

plt.figure(figsize=(12, 6))
plt.plot(song_count.index, song_count.values, marker='o', linestyle='-', color='blue')
plt.xlabel('Yıl')
plt.ylabel('Şarkı Sayısı')
plt.title('Beatles Şarkı Sayısı - Yıllara Göre')
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```