In [ ]:

```
from spark.spark_config import create_spark_session
spark = create_spark_session()
```

In [2]:

```
from data_loader import DataLoader
from preprocessor import Preprocessor
from nlp_processor import NLPProcessor
from feature_extractor import FeatureExtractor
from eda_analyser import EdaAnalyser
from repository.cassandra_provider import CassandraProvider
```

In [3]:

```
# Load file
data_loader = DataLoader(spark)
df = data_loader.load_raw_csv()
# df = df.limit(1000000) # Test
```

In [4]:

```
# EDA
analyser = EdaAnalyser(df)
analyser.run_full_eda_report(['title', 'lyrics', 'views'])
```

```
Columns and their datatypes present in the dataset:
root
 |-- title: string (nullable = true)
 |-- tag: string (nullable = true)
 |-- artist: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- views: integer (nullable = true)
 |-- features: string (nullable = true)
 |-- lyrics: string (nullable = true)
 |-- id: integer (nullable = true)
 |-- language_cld3: string (nullable = true)
 |-- language_ft: string (nullable = true)
 |-- language: string (nullable = true)

Sample 5 rows:
```

```
+-----------------+---+---------+----+------+--------------------+--------
------------+---+-------------+-----------+--------+
|            title|tag|   artist|year| views|            features|
lyrics| id|language_cld3|language_ft|language|
+-----------------+---+---------+----+------+--------------------+--------
------------+---+-------------+-----------+--------+
|        Killa Cam|rap|  Cam'ron|2004|173166|{"Cam\\'ron","Ope...|[Chorus:
Opera St...|  1|           en|         en|      en|
|       Can I Live|rap|    JAY-Z|1996|468624|                  {}|[Produce
d by Irv ...|  3|           en|         en|      en|
|Forgive Me Father|rap| Fabolous|2003|  4743|                  {}|Maybe ca
use I'm e...|  4|           en|         en|      en|
|     Down and Out|rap|  Cam'ron|2004|144404|{"Cam\\'ron","Kan...|[Produce
d by Kany...|  5|           en|         en|      en|
|           Fly In|rap|Lil Wayne|2005| 78271|                  {}|[Intro]
\nSo they ...|  6|           en|         en|      en|
+-----------------+---+---------+----+------+--------------------+--------
------------+---+-------------+-----------+--------+
only showing top 5 rows
```

```
Dimension of the Dataframe is: (5134856, 11)
Number of null values:
```

```
+-----+------+-----+
|title|lyrics|views|
+-----+------+-----+
|  165|     0|    0|
+-----+------+-----+

Top 5 most viewed pl songs:
```

```
+--------------------+---------------+-------+
|               title|         artist|  views|
+--------------------+---------------+-------+
|Pan Tadeusz - Inw...|Adam Mickiewicz|1865798|
|          Tamagotchi|      TACONAFIDE| 618358|
|           Half dead|     Quebonafide| 484043|
|          Patoreakcja|           Mata| 443703|
|             Nie nie|     Otsochodzi| 399099|
+--------------------+---------------+-------+

Top 5 most viewed en songs:
```

```
[Stage 10:>                                                         (0 +
1) / 1]
```

```
+------------+--------------+--------+
|       title|        artist|   views|
+------------+--------------+--------+
|     Rap God|        Eminem|17575634|
|         WAP|       Cardi B|16003444|
|Shape of You|    Ed Sheeran|14569727|
|     HUMBLE.|Kendrick Lamar|11181199|
|   The Hills|    The Weeknd| 9291775|
+------------+--------------+--------+
```

In [5]:

```
df = Preprocessor.run(df)
```

In [6]:

```
df_tokenized = NLPProcessor.run(df)
df_tokenized.cache()
```

Out[6]:

```
DataFrame[title: string, tag: string, artist: string, year: int, views:
int, features: string, lyrics: string, id: int, language_cld3: string, l
anguage_ft: string, language: string, lyrics_cleaned: string, words_lemm
atized: array<string>]
```

In [7]:

```
extractor = FeatureExtractor()
df_final = extractor.fit(df_tokenized).transform(df_tokenized)
```

In [8]:

```
cassandra_provider = CassandraProvider()
cassandra_provider.save(df_final)
```

In [9]:

```
spark.stop()
```