Importando as bibliotecas

```
In [122]:
```

```
import matplotlib.pyplot as plt
import nltk
#nltk.download(). Download das stopwords palavras que não tem valores semânticos
nltk.download('stopwords')
from nltk.corpus import PlaintextCorpusReader
from nltk.corpus import stopwords
#bibliotecas para trabalhar com cores nas StopWords
from matplotlib.colors import ListedColormap
from wordcloud import WordCloud
import string
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

Criando os corpus

```
In [123]:
```

```
corpus = PlaintextCorpusReader('/content/drive/MyDrive/CursoMD/Arquivos', '.*', encoding =
```

Leitura dos arquivos do drive

```
In [124]:
```

```
arquivos = corpus.fileids()
#primeiro arquivo
arquivos[0]
```

```
Out[124]:
```

'1.txt'

In []:

```
#zero a 10
arquivos[0:10]
```

In []:

```
#imprime todos os nomes
for a in arquivos:
    print(a)
```

```
In [126]:
```

```
#Acesso ao texto do primeiro arquivo
texto = corpus.raw('1.txt')
texto
```

Out[126]:

"@relation 'Reuters-21578 Corn ModApte Train-weka.filters.unsupervised.attri
bute.NumericToBinary-weka.filters.unsupervised.instance.RemoveFolds-S0-N5-F
1'\r\n"

In [127]:

```
# Acesso a todos as palavras de todos os arquivos do corpus
todo_texto = corpus.raw()
#todo_texto
```

In [128]:

```
# Obtenção de todas as palavras do corpus e visualização da quantidade
palavras = corpus.words()
#acessando pelo indíce
palavras[171]
```

Out[128]:

'on'

In [129]:

```
#quantidade de palavras encontradas no nosso corpus
len(palavras)
```

Out[129]:

619424

In []:

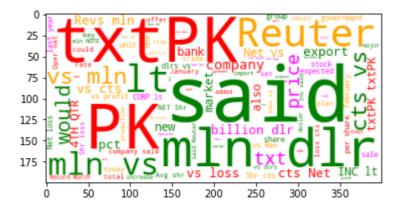
```
#Usando o NLTK, obtemos as stop word em inglês
stops = stopwords.words('english')
#stops = stopwords.words('portuguese')
stops
```

Criação da núvens de palavras

In [101]:

Out[101]:

<matplotlib.image.AxesImage at 0x7f072a048490>



In [92]:

Out[92]:

555712

In [102]:

#Vamos ver quantas palavras temos na stopwords
len(stops)

Out[102]:

179

In [103]:

```
#Adicionar novos stopwords
stops.append('txtPK')
stops.append('PK')
len(stops)
```

Out[103]:

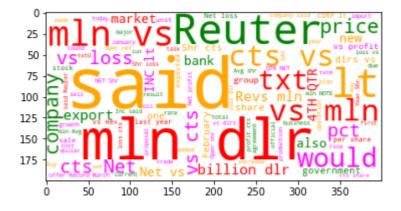
181

Vamos testar novamente e fazer um comparativo

In [104]:

Out[104]:

<matplotlib.image.AxesImage at 0x7f07265fda50>



Vamos gerar uma lista de palavras sem as stopWords e Pontuações

In [130]:

```
# Criação de nova lista de palavras, removendo stop words
palavras_semstop = [p for p in palavras if p not in stops]
len(palavras_semstop)
```

Out[130]:

552757

In [135]:

#Remoção da pontuação, gerando uma lista sem stop words e sem pontuação
palavras_sem_pontuacao = [p for p in palavras_semstop if p not in string.punctuation]
len(palavras_sem_pontuacao)

Out[135]:

489132

In []:

Cálculo da frequência das palavras e visualização das mais comuns frequencia = nltk.FreqDist(palavras_sem_pontuacao) frequencia

In [140]:

```
#mais comuns
mais_comuns = frequencia.most_common(100)
mais_comuns
```

```
Out[140]:
```

```
[(',', 3886),
 ('said', 3398),
 ('3', 2836),
 ('0', 2728),
 ('mln', 2724),
 ('1', 2321),
('vs', 2201),
 ('J', 2145),
 ('dlrs', 1946),
 ('000', 1641),
 ('2', 1584),
 ('&#', 1466)
 (";',", 1432),
 ('The', 1429),
 ('U', 1422),
 ('S', 1414),
 ('cts', 1374),
 ('\x00\x00\x00', 1318),
 ('4', 1296),
 ('lt', 1288),
 ('5', 1274),
 ('Reuter', 1214),
 ('\x10', 1181),
 ('pct', 1147),
 ('6', 1117),
 ('\x00', 1117),
 ('8', 1111),
 ('\x0f', 1087),
 ('7', 1080),
 ('\x14', 1072),
 ('\x91', 1048),
 ('\x08', 1041),
 ('\x83', 1041),
 ('9', 1039),
 ('\x03', 1039),
 ('\x9e', 995),
 ('\x92', 993),
 ('÷', 991),
 ('\x06', 975),
 ('\x05', 967),
 ('\x8e', 963),
 ('¶', 962),
 ('\x1b', 962),
 ('\x90', 960),
 ('\x93', 960),
 ('\x18', 949),
 ('\x94', 945),
 ('\x07', 945),
 ('\x8b', 942),
 ('\x8d', 941),
  '\x9a', 936),
 ('±', 933),
 ('\x0e', 929),
```

```
('ï', 924),
('A', 923),
('\x1a', 922),
('\x9d', 912),
(''', 911),
('\x16', 910),
('\x81', 910),
('»', 906),
('\x82', 906),
('\x9c', 898),
('\x9b', 897),
('¤', 897),
('°', 886),
('£', 886),
('\x02', 882),
('\x19', 880),
('year', 878),
('\x96', 878),
('\x8f', 877),
('\x8a', 877),
('¬', 871),
('\x01', 869),
('\x89', 866),
('8', 864),
('\x17', 864),
('¡', 859),
('"', 859),
('\x84', 859),
('.', 857),
('\x87', 857),
('\xad', 852),
('\x99', 849),
 '\x7f', 847),
('«', 844),
('¢', 844),
('©', 841),
('x', 834),
('\x95', 829),
('¦', 827),
('§', 826),
('\x04', 823),
('\', 821),
('\x13', 818),
('V', 815),
('\x86', 810),
('\x98', 808),
('I', 801)]
```