In [11]:

```
import numpy as np
import pandas as pd
import sklearn
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import roc_auc_score
```

In [12]:

```
train = pd.read_csv('linear_train.csv', names=['Last Name', 'Label'])
test = pd.read_csv('linear_test.csv', names=['Last Name'])
ans_example = pd.read_csv('linear_ans_example.csv')
```

In [17]:

```
#  Создадим новые признаки - n-граммы
#  Кроме этих признаков, у нас больше ничего не будет
#  Для этого создадим CountVectorizer, обучим его на словах из train
vect = CountVectorizer(ngram_range=(1, 7), analyzer='char_wb', lowercase=False)
fitted = vect.fit(train['Last Name'])
train_n = fitted.transform(train['Last Name'])
test_n = fitted.transform(test['Last Name'])
```

In [14]:

```
#  Далее используем логистическую регрессию
lg_regr = LogisticRegression(random_state=12, solver='lbfgs', warm_start=True);
lg_regr.fit(train_n, train['Label'])
prediction = lg2.predict_proba(test_n)
```

In [15]:

```
with open('output.txt', 'w') as file_out:
    file_out.write('Id,Answer\n')
    i = 0
    for item in prediction[:, 1]:
        file_out.write(str(i) + ',' + str(item) + '\n')
        i += 1
```