In [9]:
```
import pandas as pd
import sklearn as sk
import math
```

In [10]:
```
first_sentence = "Jupiter is the largest Planet"
second_sentence = "Mars is the fourth planet from the Sun"
#split so each word have their own string
first_sentence = first_sentence.split(" ")
second_sentence = second_sentence.split(" ")#join them to remove common duplicate w
total= set(first_sentence).union(set(second_sentence))
print(total)
```

```
{'fourth', 'is', 'planet', 'Planet', 'the', 'Sun', 'Jupiter', 'largest', 'from',
'Mars'}
```

In [11]:
```
wordDictA = dict.fromkeys(total, 0)
wordDictB = dict.fromkeys(total, 0)
for word in first_sentence:
    wordDictA[word]+=1

for word in second_sentence:
    wordDictB[word]+=1
```

In [12]:
```
pd.DataFrame([wordDictA, wordDictB])
```

Out[12]:

	fourth	is	planet	Planet	the	Sun	Jupiter	largest	from	Mars
0	0	1	0	1	1	0	1	1	0	0
1	1	1	1	0	2	1	0	0	1	1

In [13]:
```
def computeTF(wordDict, doc):
    tfDict = {}
    corpusCount = len(doc)
    for word, count in wordDict.items():
        tfDict[word] = count/float(corpusCount)
    return(tfDict)
#running our sentences through the tf function:
tfFirst = computeTF(wordDictA, first_sentence)
tfSecond = computeTF(wordDictB, second_sentence)
#Converting to dataframe for visualization
tf = pd.DataFrame([tfFirst, tfSecond])
```

In [14]:
```
tf
```

Out[14]:

	fourth	is	planet	Planet	the	Sun	Jupiter	largest	from	Mars
0	0.000	0.200	0.000	0.2	0.20	0.000	0.2	0.2	0.000	0.000
1	0.125	0.125	0.125	0.0	0.25	0.125	0.0	0.0	0.125	0.125

In [15]:
```
def computeIDF(docList):
    idfDict = {}
    N = len(docList)

    idfDict = dict.fromkeys(docList[0].keys(), 0)
    for word, val in idfDict.items():
        idfDict[word] = math.log10(N / (float(val) + 1))

    return(idfDict)
```

```
#inputing our sentences in the log file
idfs = computeIDF([wordDictA, wordDictB])
```

In [16]:
```
idfs
```

Out[16]:
```
{'fourth': 0.3010299956639812,
 'is': 0.3010299956639812,
 'planet': 0.3010299956639812,
 'Planet': 0.3010299956639812,
 'the': 0.3010299956639812,
 'Sun': 0.3010299956639812,
 'Jupiter': 0.3010299956639812,
 'largest': 0.3010299956639812,
 'from': 0.3010299956639812,
 'Mars': 0.3010299956639812}
```

In [17]:
```
def computeTFIDF(tfBow, idfs):
    tfidf = {}
    for word, val in tfBow.items():
        tfidf[word] = val*idfs[word]
    return(tfidf)
#running our two sentences through the IDF:
idfFirst = computeTFIDF(tfFirst, idfs)
idfSecond = computeTFIDF(tfSecond, idfs)
#putting it in a dataframe
idf= pd.DataFrame([idfFirst, idfSecond])
print(idf)
```

```
     fourth        is    planet    Planet       the       Sun   Jupiter  \
0  0.000000  0.060206  0.000000  0.060206  0.060206  0.000000  0.060206
1  0.037629  0.037629  0.037629  0.000000  0.075257  0.037629  0.000000

    largest      from      Mars
0  0.060206  0.000000  0.000000
1  0.000000  0.037629  0.037629
```

In [ ]:
```

```