In [4]:
```
import nltk
from nltk.tokenize import word_tokenize
from nltk.stem import PorterStemmer
from nltk.corpus import wordnet

nltk.download('punkt')
nltk.download('wordnet')
nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\pronn\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\pronn\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data]     C:\Users\pronn\AppData\Roaming\nltk_data...
```

Out[4]: True

Morphological Analysis

In [5]:
```
def analyze_word(word) :
    stemmer = PorterStemmer()
    stem = stemmer.stem(word)

    lemma = None
    synsets = wordnet.synsets(word)
    if synsets :
        lemma = synsets[0].lemmas()[0].name()

    plural = "plural" if lemma and stem != lemma else "singular"

    tense = "present"
    pos = None
    for synset in synsets :
        pos = synset.pos()
        if "past" in synset.name() :
            tense = "past"
            break
    return {
        "word" : word,
        "root" : stem,
        "singular/plural" : plural,
        "tense" : tense,
        "POS" : pos
    }

input_text = input("Enter a sentence: ")
tokens = word_tokenize(input_text)

print("{:<15} {:<15} {:<15} {:<15} {:<15}".format("word", "root", "singular/plural"
print("-"*75)
```

```
for token in tokens :
    analysis = analyze_word(token)
    print("{:<15} {:<15} {:<15} {:<15} {:<15}".format(
        analysis["word"],
        analysis["root"],
        analysis["singular/plural"],
        analysis["tense"],
        analysis["POS"]
    ))
```

```
Enter a sentence: Colorful skies looked pretty more importantly loved
word            root            singular/plural tense           POS
--------------------------------------------------------------------------
Colorful        color           plural          present         a
skies           sky             singular        present         v
looked          look            singular        present         v
pretty          pretti          plural          present         r
more            more            plural          present         r
importantly     importantli     plural          present         r
loved           love            singular        present         a
```

Word Generation

In [6]:

```
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
from nltk.corpus import wordnet

def generate_word_forms(root) :
    # Singular and plural forms
    singular = root
    if root.endswith("s") or root.endswith("x") or root.endswith("z") or root.endsw
        plural = root + "es"
    elif root.endswith("y") and len(root) > 1 and root[-2] not in "aeiou" :
        plural = root[:-1] + "ies"
    else :
        plural = root + "s"

    # Comparative and superlative forms
    if root.endswith("e") :
        comparative = root + "r"
        superlative = root + "st"
    elif len(root) >= 2 and root[-1] not in "aeiou" and root[-2] not in "aeiou" :
        comparative = root + root[-1] + "er"
        superlative = root + root[-1] + "est"
    else :
        comparative = root + "er"
        superlative = root + "est"

    return {
        "singular" : singular,
        "plural" : plural,
        "comparative" : comparative,
        "superlative" : superlative
    }
```

```
input_text = input("Enter a sentence: ")
tokens = word_tokenize(input_text)

print("{:<15} {:<15} {:<15} {:<15} {:<15}".format("word", "singular", "plural", "co
print("-"*75)

for token in tokens :
    word_forms = generate_word_forms(token)
    print("{:<15} {:<15} {:<15} {:<15} {:<15}".format(
        token,
        word_forms["singular"],
        word_forms["plural"],
        word_forms["comparative"],
        word_forms["superlative"]
    ))
```

```
Enter a sentence: Short koala bears larger frank
word            singular        plural          comparative     superlative
---------------------------------------------------------------------------
Short           Short           Shorts          Shortter        Shorttest
koala           koala           koalas          koalaer         koalaest
bears           bears           bearses         bearsser        bearssest
larger          larger          largers         largerer        largerest
frank           frank           franks          frankker        frankkest
```