FOL to CNF

Create a list of skolem constants

```
import re
def fol-to-cnf(fol):
    statement = fol.replace ("<=>", "_")
    while '_' in statement:
        i = statement.index('_')
        new-statement = '[' + statement [:i] + '=>' +
                        statement [i+1:] + ']&[' +
                        statement[i+1:] + '=>' +
                        statement[:i] + ']'

    statement= statement.replace ("=>", "-")
    expr = '\[([^]]+)\]'
    statements = re.findall (expr, statement)
    for i, s in enumerate (statements):
        if '[' in s and ']' not in s:
            statements [i] + = ']'

    for s in statements:
        statement = statement.replace(s, fol-to-CNF(s))

    while '-' in statements:
        statement = statement.replace (s,
        i = statement.index('-')
        br = -1 ______ ('[') if '[' in statement
        new-statement = '~' + statement(br:i] + '|' + statement[i+1:]
        statement = statement [:br] + new-statement if br>0 else
                        new-statement

    while '~∀' in statement :
        i = statement index ('~∀')
        statement = list (statement)
        statement [i], statement [i+1], statement [i+2] = '∃', statement[i+2],
                                                              '~'
        statement = ''. join (s)
    statement = statement . replace ('~[∀', '[~∀')
    statement = statement.replace ('~[∃', '[~∃')
    expr = '([∀|∃].)'
    statements = re.findall (expr, statement)
```

```
for s in statements:
    statement = statement.replace(s, fol_to_cnf(s))
exps = '~\[[^]]+\]'
statements = re.findall(exps, statement)
for s in statements:
    statement = statement.replace(s, DeMorgan(s))
return statement

print(Skolemization(fol_to_cnf("animal(y) <=> loves(x,y)")))
```

("∀x [∀y [animal(y) => loves(x,y)] => [∃z [loves(z,x)]]")

("∀[american(x) & weapon(y) & sells(x,y,z) & hostile(z)] => criminal(x)")

Output:

[~animal(y) | loves(x,y)] & [~loves(x,y) | animal(y)]

[animal(G(x)) & ~loves(x, G(x))] | [loves(F(x), x)]

[~american(x) | ~weapon(y) | ~sells(x,y,z) | ~hostile(z)] | ~~hostile(z)~~ criminal(x).

Explanation:

$\forall x \; King(x) \wedge Greedy(x) \Rightarrow Evil(x)$

$King(Richard) \wedge Greedy(Richard) \Rightarrow Evil(Richard)$.

$A \Leftrightarrow B$

↳ Replace with $(A \Rightarrow B) \wedge (B \Rightarrow A)$

$A \Rightarrow B$

↳ $\sim A \vee B$

$\sim[A] \rightarrow$ demorgan

$\sim[King(Richard) \wedge Greedy(Richard)] \vee Evil(Richard)$

$\sim King(Richard) \vee \sim Greedy(Richard) \vee Evil(Richard)$

19/11/24