

CONSTRAIN SATISFACTION PROBLEM

NAME: RISHAL RAMESH

EXP NO: 3

REG NO: RA1911030010084

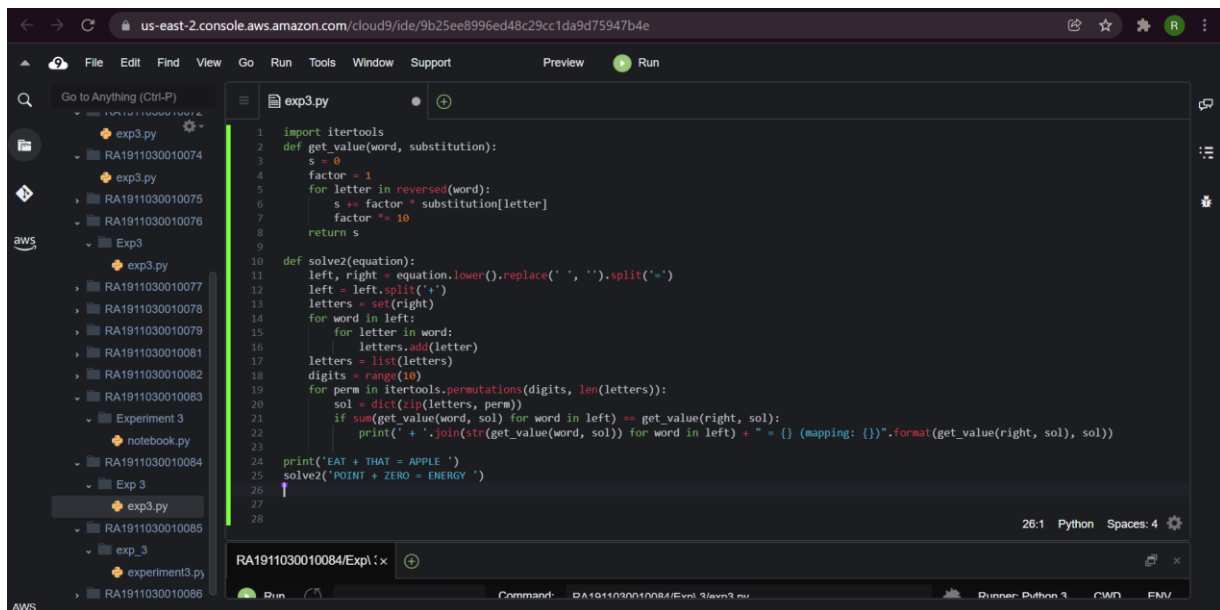
CODE:

```
import itertools

def get_value(word, substitution):
    s = 0
    factor = 1
    for letter in reversed(word):
        s += factor * substitution[letter]
        factor *= 10
    return s

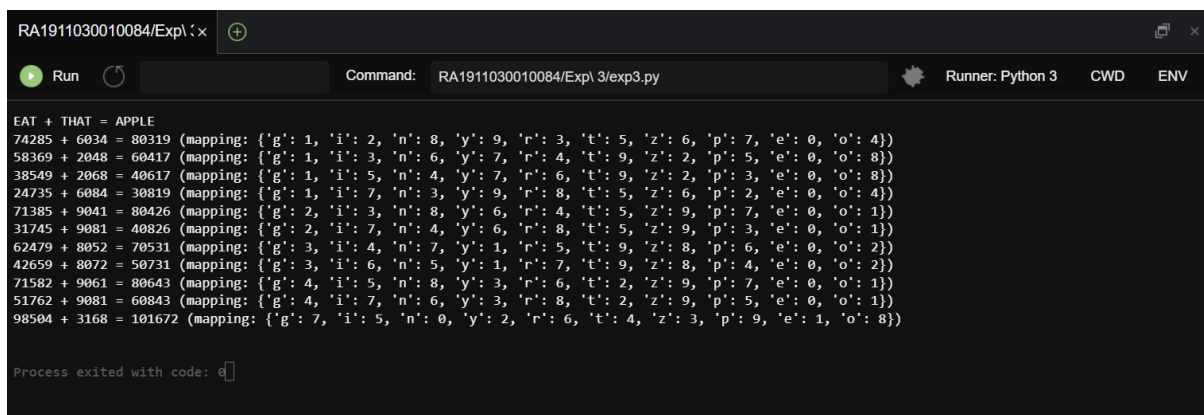
def solve2(equation):
    left, right = equation.lower().replace(' ', '').split('=')
    left = left.split('+')
    letters = set(right)
    for word in left:
        for letter in word:
            letters.add(letter)
    letters = list(letters)
    digits = range(10)
    for perm in itertools.permutations(digits, len(letters)):
        sol = dict(zip(letters, perm))
        if sum(get_value(word, sol) for word in left) == get_value(right, sol):
            print(' + '.join(str(get_value(word, sol)) for word in left) + " = {} (mapping: {})"
                  .format(get_value(right, sol), sol))
```

```
print('EAT + THAT = APPLE ')
solve2('POINT + ZERO = ENERGY ')
```



```
1 import itertools
2 def get_value(word, substitution):
3     s = 0
4     factor = 1
5     for letter in reversed(word):
6         s += factor * substitution[letter]
7         factor *= 10
8     return s
9
10 def solve2(equation):
11     left, right = equation.lower().replace(' ', '').split('=')
12     left = left.split('+')
13     letters = set(right)
14     for word in left:
15         for letter in word:
16             letters.add(letter)
17     letters = list(letters)
18     digits = range(10)
19     for perm in itertools.permutations(digits, len(letters)):
20         sol = dict(zip(letters, perm))
21         if sum(get_value(word, sol) for word in left) == get_value(right, sol):
22             print(" + ".join(str(get_value(word, sol)) for word in left) + " = {} (mapping: {})".format(get_value(right, sol), sol))
23
24 print('EAT + THAT = APPLE ')
25 solve2('POINT + ZERO = ENERGY ')
26
27
28
```

OUTPUT:



```
RA1911030010084/Expl:\x
Run Command: RA1911030010084/Expl 3/exp3.py Runner: Python 3 CWD ENV

EAT + THAT = APPLE
74285 + 6034 = 80319 (mapping: {'g': 1, 'i': 2, 'n': 8, 'y': 9, 'r': 3, 't': 5, 'z': 6, 'p': 7, 'e': 0, 'o': 4})
58369 + 2048 = 60417 (mapping: {'g': 1, 'i': 3, 'n': 6, 'y': 7, 'r': 4, 't': 9, 'z': 2, 'p': 5, 'e': 0, 'o': 8})
38549 + 2068 = 40617 (mapping: {'g': 1, 'i': 5, 'n': 4, 'y': 7, 'r': 6, 't': 9, 'z': 2, 'p': 3, 'e': 0, 'o': 8})
24735 + 6084 = 30819 (mapping: {'g': 1, 'i': 7, 'n': 3, 'y': 9, 'r': 8, 't': 5, 'z': 6, 'p': 2, 'e': 0, 'o': 4})
71385 + 9041 = 80426 (mapping: {'g': 2, 'i': 3, 'n': 8, 'y': 6, 'r': 4, 't': 5, 'z': 9, 'p': 7, 'e': 0, 'o': 1})
31745 + 9081 = 40826 (mapping: {'g': 2, 'i': 7, 'n': 4, 'y': 6, 'r': 8, 't': 5, 'z': 9, 'p': 3, 'e': 0, 'o': 1})
62479 + 8052 = 70531 (mapping: {'g': 3, 'i': 4, 'n': 7, 'y': 1, 'r': 5, 't': 9, 'z': 8, 'p': 6, 'e': 0, 'o': 2})
42659 + 8072 = 50731 (mapping: {'g': 3, 'i': 6, 'n': 5, 'y': 1, 'r': 7, 't': 9, 'z': 8, 'p': 4, 'e': 0, 'o': 2})
71582 + 9061 = 80643 (mapping: {'g': 4, 'i': 5, 'n': 8, 'y': 3, 'r': 6, 't': 2, 'z': 9, 'p': 7, 'e': 0, 'o': 1})
51762 + 9081 = 60843 (mapping: {'g': 4, 'i': 7, 'n': 6, 'y': 3, 'r': 8, 't': 2, 'z': 9, 'p': 5, 'e': 0, 'o': 1})
98504 + 3168 = 101672 (mapping: {'g': 7, 'i': 5, 'n': 0, 'y': 2, 'r': 6, 't': 4, 'z': 3, 'p': 9, 'e': 1, 'o': 8})

Process exited with code: 0
```

RESULT:

The constraint satisfying problem $EAT + THAT = APPLE$ solved using the carry over technique and values for the alphabets obtained successfully.