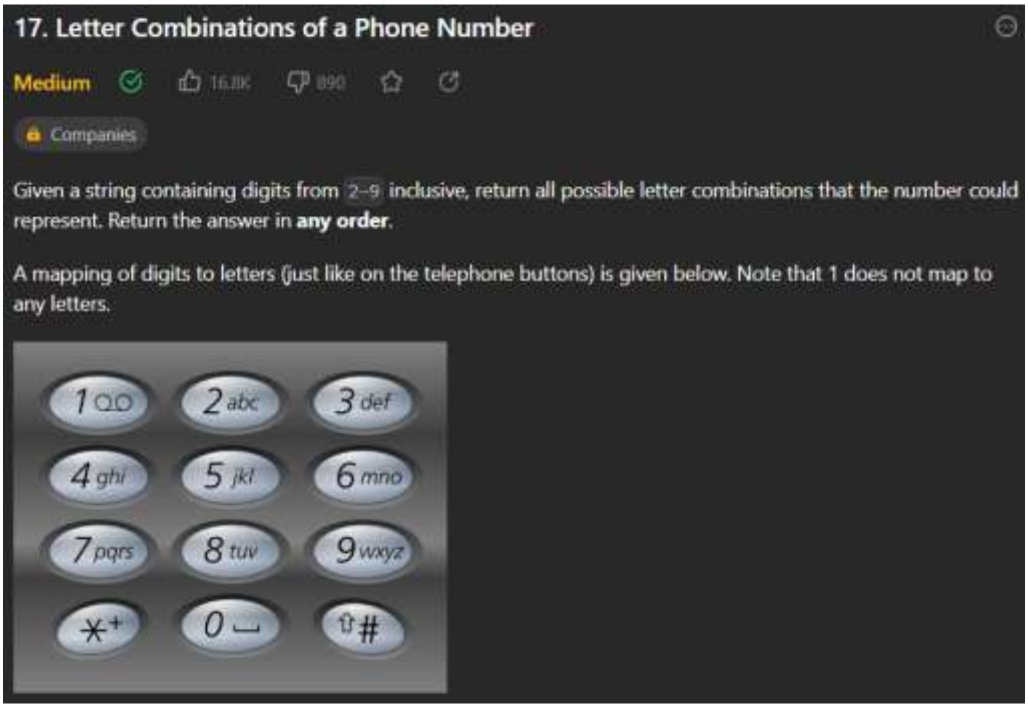
LEETCODE SOLUTIONS - YOGESH MUNEES

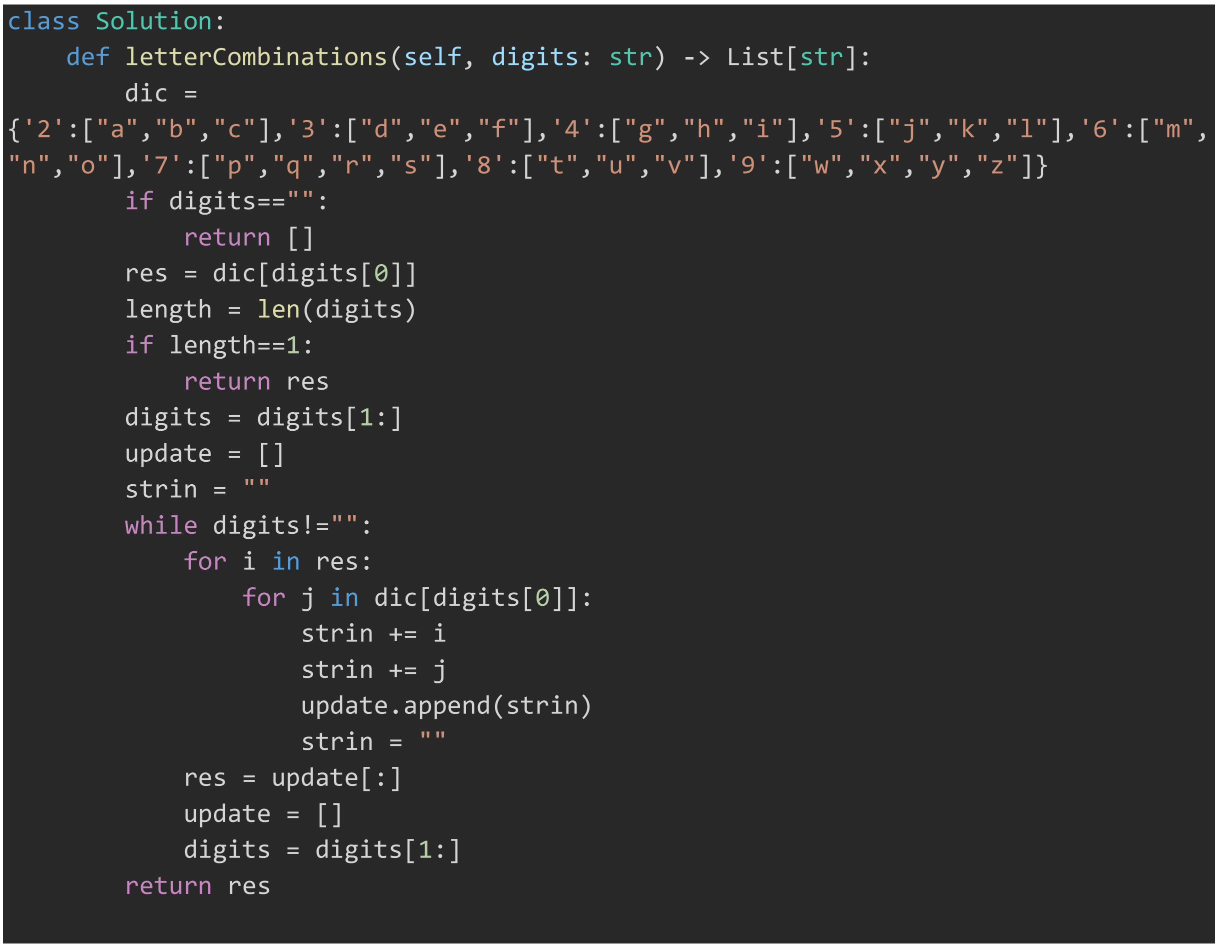


17. Letter Combinations of a Phone Number

Given a string containing digits from `2-9` inclusive, return all possible letter combinations that the number could represent. Return the answer in **any order**.

A mapping of digits to letters (just like on the telephone buttons) is given below. Note that 1 does not map to any letters.

Solution:



```
class Solution:
    def letterCombinations(self, digits: str) -> List[str]:
        dic = 
{'2':["a","b","c"],'3':["d","e","f"],'4':["g","h","i"],'5':["j","k","l"],'6':["m",
"n","o"],'7':["p","q","r","s"],'8':["t","u","v"],'9':["w","x","y","z"]}
        if digits=="":
            return []
        res = dic[digits[0]]
        length = len(digits)
        if length==1:
            return res
        digits = digits[1:]
        update = []
        strin = ""
        while digits!="":
            for i in res:
                for j in dic[digits[0]]:
                    strin += i
                    strin += j
                    update.append(strin)
                    strin = ""
            res = update[:]
            update = []
            digits = digits[1:]
        return res
```