Problem #2138: Divide a String into Groups of Size k (Easy)

<https://leetcode.com/problems/divide-a-string-into-groups-of-size-k/description/>

My Solution:

1. If the length of s modulo k is greater than 0 (i.e. length of s when divided by k has a remainder greater than 0), then using fill character for length of (k – remainder) make the string s evenly divisible by k.
2. Initialize I to 0
3. Initialize res to an empty list to hold the result.
4. While loop with condition while I is less than the length of s, append to res a segment from index I to index (I + k).
5. Then return res.

class Solution:

def divideString(self, s: str, k: int, fill: str) -> List[str]:

if len(s) % k > 0:

s += (fill)\*(k - (len(s) % k))

i = 0

res = []

while i < len(s):

res.append(s[i : i + k])

i += k

return res