**Problem #1380: Lucky Number in a Matrix (Easy)**

<https://leetcode.com/problems/lucky-numbers-in-a-matrix/description/>

**My Solution:**

1. Let m be the length of matrix, i.e., number of rows.
2. Let n be the length of matrix row 0 i.e., number of columns.
3. Outer for loop to iterate through each of the rows of matrix.

Initialize minEle to be the first element in the row and minEleCol to be 0.

Inner for loop to iterate through the columns in row I to find minEle and minEleCol.

If there is an element smaller than minEle, update minEle to be that element and its column to be minEleCol.

Now we need to check if minEle is the maximum in its column.

Initialize a Boolean found to True.

Use a for loop with index k to iterate through the rows of the column minEleCol.

If there is an element which is greater than minEle, then set Found to False and break out of the loop.

If Found is true (i.e., we did not break out of the for loop with index k), then append minEle to res.

1. Return res.

class Solution:

def luckyNumbers (self, matrix: List[List[int]]) -> List[int]:

m = len(matrix)

n = len(matrix[0])

res = []

for i in range(m):

minEle = matrix[i][0]

minEleCol = 0

for j in range(n):

if matrix[i][j] < minEle:

minEle = matrix[i][j]

minEleCol = j

found = True

for k in range(m): # Check if minEle is the maximum in minEleCol

if matrix[k][minEleCol] > minEle:

found = False

break

if found:

res.append(minEle)

return res