

프로그래머스 - 2차원으로 만들기

≡ 태그

answer[i/n][i%n] = num_list[i] splice(0 n)

자바스크립트

2차원으로 만들기

문제 설명

정수 배열 num_list와 정수 n이 매개변수로 주어집니다. num_list를 다음 설명과 같이 2차원 배열로 바꿔 return하도록 solution 함수를 완성해주세요.

num_list가 [1, 2, 3, 4, 5, 6, 7, 8] 로 길이가 8이고 n이 2이므로 num_list를 2 * 4 배열로 다음과 같이 변경합니다. 2차원으로 바꿀 때에는 num_list의 원소들을 앞에서부터 n개씩 나눠 2차원 배열로 변경합니다.

제한사항

num_list의 길이는 n의 배 수개입니다.

$0 \leq \text{num_list의 길이} \leq 150$

$2 \leq n < \text{num_list의 길이}$

입출력 예 #1

num_list: [1, 2, 3, 4, 5, 6, 7, 8] n:2 ,result:[[1, 2], [3, 4], [5, 6], [7, 8]]

num_list가 [1, 2, 3, 4, 5, 6, 7, 8] 로 길이가 8이고 n이 2이므로 2 * 4 배열로 변경한 [[1, 2], [3, 4], [5, 6], [7, 8]] 을 return합니다.

입출력 예 #2

num_list: [100, 95, 2, 4, 5, 6, 18, 33, 948] ,n:3 ,result:[[100, 95, 2], [4, 5, 6], [18, 33, 948]]

num_list가 [100, 95, 2, 4, 5, 6, 18, 33, 948] 로 길이가 9이고 n이 3이므로 3 * 3 배열로 변경한 [[100, 95, 2], [4, 5, 6], [18, 33, 948]] 을 return합니다.

```
function solution(num_list, n) {  
  // 빈 2차원 배열 만들기  
  let num_list_2d = [];  
  
  // num_list를 반복하여 한 번에 n개의 요소를 추출합니다.  
  while (num_list.length > 0) {  
    // splice()를 사용하여 num_list에서 n개의 요소를 추출합니다.  
    let sub_array = num_list.splice(0, n);  
  
    // 추출된 요소를 num_list_2d로 푸시합니다.  
    num_list_2d.push(sub_array);  
  }  
  
  // 결과 2차원 배열을 반환합니다.  
  return num_list_2d;  
}
```

splice() 메서드는 배열에서 요소를 제거하고 새 배열로 반환하는 데 사용할 수 있습니다. 이 경우 splice(0, n)은 num_list 배열의 시작 부분에서 n개의 요소를 제거하고 새 배열로 반환합니다.

예를 들어 num_list가 [1, 2, 3, 4, 5]이고 n이 3이면 splice(0, n)은 [1, 2, 3]을 반환하고 num_list에서 해당 요소를 제거하여 [4, 5].

```
class Solution {
    public int[][] solution(int[] num_list, int n) {
        int[][] answer = {};

        int length = num_list.length;

        answer = new int[length/n][n];

        for(int i=0; i<length; i++){
            answer[i/n][i%n]=num_list[i];
        }

        return answer;
    }
}
```

solution() 함수는 정수 num_list 배열과 정수 n을 매개변수로 받습니다.
2차원 배열 응답을 생성하고 적절한 수의 행과 열로 초기화합니다. 행 수는 length/n(길이는 num_list의 길이)이고 열 수는 n입니다.
그런 다음 num_list를 반복하고 공식 answer[i/n][i%n] = num_list[i]를 사용하여 응답 배열의 위치에 각 요소를 할당합니다. 이 수식은 원래 num_list 배열의 위치를 기준으로 각 요소의 행 및 열 인덱스를 결정합니다.

```
answer[i/n][i%n] = num_list[i]
```

i/n은 현재 반복 횟수(i)를 n으로 나누어 계산되는 행 인덱스입니다. 이렇게 하면 num_list 배열의 각 요소에 대한 행 인덱스가 제공되며, 첫 번째 행(인덱스 0)부터 시작하여 각 후속 행에 대해 1씩 증가합니다.

i%n은 현재 반복 횟수(i)를 n으로 나눈 나머지를 취하여 계산되는 열 인덱스입니다. 이렇게 하면 num_list 배열의 각 요소에 대한 열 인덱스가 제공되며, 첫 번째 열(인덱스 0)부터 시작하여 각 후속 열에 대해 1씩 증가합니다.

예를 들어 n이 2이고 i가 3이면 i/n은 1(두 번째 행)이고 i%n은 1(두 번째 열)입니다. 이는 num_list 배열의 인덱스 3에 있는 요소가 answer 배열의 두 번째 행과 두 번째 열에 할당됨을 의미합니다.