Transpose and Flatten ★

65/115 challenges solved

Rank: 32507 | Points: 835

Your Transpose and Flatten submission got 20.00 points. Share Tweet

Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial

Transpose

We can generate the transposition of an array using the tool `numpy.transpose`.

It will not affect the original array, but it will create a new array.

```
import numpy

my_array = numpy.array([[1,2,3],
                        [4,5,6]])
print numpy.transpose(my_array)

#Output
[[1 4]
 [2 5]
 [3 6]]
```

Flatten

The tool flatten creates a copy of the input array flattened to one dimension.

```
import numpy

my_array = numpy.array([[1,2,3],
                        [4,5,6]])
print my_array.flatten()

#Output
[1 2 3 4 5 6]
```

Task

You are given a NXM integer array matrix with space separated elements (N = rows and M = columns).

Your task is to print the transpose and flatten results.

Input Format

The first line contains the space separated values of N and M.

The next N lines contains the space separated elements of M columns.

Output Format

First, print the transpose array and then print the flatten.

Sample Input

```
2 2
1 2
3 4
```

Sample Output

```
[[1 3]
 [2 4]]
[1 2 3 4]
```

Change Theme | Python 3

```
1  import numpy as np
2
3  if __name__ == '__main__':
4      n, m = tuple(map(int, input().split()))
5
6      matriz = []
7      for _ in range(n):
8          matriz.append(list(map(int, input().split())))
9
10     matriz = np.array(matriz)
11
12     print(np.transpose(matriz))
13     print(matriz.flatten())
14
```

Line: 14 Col: 1

Upload Code as File | Test against custom input | Run Code | Submit Code

You have earned 20.00 points!

65/115 challenges solved.

57%

Congratulations

You solved this challenge. Would you like to challenge your friends? Next Challenge

Earn a certificate in Python

Kudos on your progress! Take the HackerRank Skills Certification test and enrich your profile Get Certified

Test case 0

Test case 1

Test case 2

Compiler Message

Success

Input (stdin) Download

```
2 2
1 2
3 4
```

Expected Output Download

```
[[1 3]
 [2 4]]
[1 2 3 4]
```