Eye and Identity ★

76/115 challenges solved

Rank: 22679 | Points: 1055

Your Eye and Identity submission got 20.00 points. Share Tweet

Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial

identity

The identity tool returns an identity array. An identity array is a square matrix with all the main diagonal elements as 1 and the rest as 0. The default type of elements is float.

```
import numpy
print numpy.identity(3) #3 is for  dimension 3 X 3

#Output
[[ 1.  0.  0.]
 [ 0.  1.  0.]
 [ 0.  0.  1.]]
```

eye

The eye tool returns a 2-D array with 1's as the diagonal and 0's elsewhere. The diagonal can be main, upper or lower depending on the optional parameter k. A positive k is for the upper diagonal, a negative k is for the lower, and a 0 k (default) is for the main diagonal.

```
import numpy
print numpy.eye(8, 7, k = 1)    # 8 X 7 Dimensional array with first upper diagonal 1.

#Output
[[ 0.  1.  0.  0.  0.  0.  0.]
 [ 0.  0.  1.  0.  0.  0.  0.]
 [ 0.  0.  0.  1.  0.  0.  0.]
 [ 0.  0.  0.  0.  1.  0.  0.]
 [ 0.  0.  0.  0.  0.  1.  0.]
 [ 0.  0.  0.  0.  0.  0.  1.]
 [ 0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.]]

print numpy.eye(8, 7, k = -2)   # 8 X 7 Dimensional array with second lower diagonal 1.
```

Task

Your task is to print an array of size NXM with its main diagonal elements as 1's and 0's everywhere else.

Note

In order to get alignment correct, please insert the line **numpy.set_printoptions(legacy='1.13')** below the numpy import.

Input Format

A single line containing the space separated values of N and M.

N denotes the rows.

M denotes the columns.

Output Format

Print the desired NXM array.

Sample Input

```
3 3
```

Sample Output

```
[[ 1.  0.  0.]
 [ 0.  1.  0.]
 [ 0.  0.  1.]]
```

Change Theme | Python 3

```
1  import numpy as np
2
3  np.set_printoptions(legacy='1.13')
4
5  if __name__ == '__main__':
6      n, m = tuple(map(int, input().split()))
7
8      matrix = np.eye(n, m, k=0)
9
10     print(matrix)
11
```

Line: 10 Col: 18

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

You have earned 20.00 points!

76/115 challenges solved.

66%

Python ★★★★★

Congratulations

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

Next Challenge

Test case 0

Test case 1

Test case 2

Compiler Message

Success

Input (stdin) Download

```
3 3
```

Expected Output Download

```
[[ 1.  0.  0.]
 [ 0.  1.  0.]
 [ 0.  0.  1.]]
```