

How to Create a Matrix, Vector, and Cell Array in MATLAB

This article covers how to create matrices, vectors, and cell arrays with the programming software MATLAB. Since MATLAB is a program offering endless possibilities, being able to understand the basics will lead to the ability to write more...

Part 1 of 4:

Creating Matrices

1. **Create a matrix.** A matrix is a rectangular set of numbers that are arranged in multiple rows and columns. Code a matrix by listing a series numbers separated by spaces or commas per row within the command window. For a new row, a semi colon is placed between the values of the old row and new row. The matrix is finished by surrounding it by brackets. Hit enter when finished to display the matrix.

1. Example: M=[2,3,4,5;6,7,8,9] or M=[2 3 4 5;6 7 8 9]

```
>> M=[2,3,4,5;6,7,8,9]
```

```
M =
```

```
     2     3     4     5
     6     7     8     9
```

```
>> M=[2 3 4 5;6 7 8 9]
```

```
M =
```

```
     2     3     4     5
     6     7     8     9
```

2. **Create a one value matrix.** Type commands such as zeros or ones to create a matrix with only one value within it. Follow these commands by the number of rows then the number of columns separated by a comma and surrounded by parenthesis.

1. Example: zeros(# of rows, # of columns)

```
>> zeros(3,5)
```

```
ans =
```

```
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
```

2. Example: ones(# of rows, # of columns)

```
>> ones(3,5)
```

```
ans =
```

```
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
```

3. Edit a ones matrix. You can multiply a ones matrix by any number to create a new matrix in which all the elements are the value which the ones matrix was multiplied by.

```
>> 2*ones(3,5)
```

```
ans =
```

```
     2     2     2     2     2
     2     2     2     2     2
     2     2     2     2     2
```

Part 2 of 4:

Creating Vectors

1. **Create a horizontal vector.** A vector is a set of numbers that are arranged in a single row or single column. Code a horizontal vector by listing a series of numbers in the vector that are separated by spaces or commas and surrounded by brackets. Hit enter once completed the code to display the vector.
 1. Example: `V=[2,3,4,5,6,7,8,9]` or `V=[2 3 4 5 6 7 8 9]`

```
>> V=[2,3,4,5,6,7,8,9]
```

```
V =
```

```
     2     3     4     5     6     7     8     9
```

```
>> V=[2 3 4 5 6 7 8 9]
```

```
V =
```

```
     2     3     4     5     6     7     8     9
```

2. **Create a vertical vector.** Start by listing a series of numbers that are separated by semicolons and surrounded by brackets. Hit enter once completed the code to display the vector.
 1. Example: `V=[2;3;4;5;6;7;8;9]`

```
>> V=[2;3;4;5;6;7;8;9]
```

```
V =
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

3. **Create a one value vector.** Type commands such as zeros or ones to create a horizontal or vertical vector with only one value within it. Follow these commands by the number of rows then the number of columns separated by a comma and surrounded by parenthesis.

1. Example: zeros(# of rows, # of columns)

```
>> zeros(1,5)
```

```
ans =
```

```
0 0 0 0 0
```

2. Example: ones(# of rows, # of columns)

```
>> ones(5,1)
```

```
ans =
```

```
1
```

```
1
```

```
1
```

```
1
```

```
1
```

3. Edit a ones vector. You can multiply a ones vector by any number to create a new vector in which all the elements are the value which the ones vector was multiplied by.

```
>> 7*ones(5,1)
```

```
ans =
```

```
7
```

```
7
```

```
7
```

```
7
```

```
7
```

4. **Create a vector with a pattern.** A pattern is a list of numbers that follow a certain rule or sequence. Type the colon operator or the linspace command. A colon operator focuses on a specific increment that the pattern follows while the linspace command focuses on the number of variables within the pattern that are have the same increment between them.

1. Use a colon operator. Type the first number of pattern, increment between the values, and last number of the pattern all separated by colons. Hit enter once code is completed.

```
>> 2:4:34  
  
ans =  
  
    2     6    10    14    18    22    26    30    34
```

2. Use linspace command. Type linspace then first number in pattern, last number in the pattern, total number of equally spaced values separated by commas and surrounded by parenthesis. Hit enter once the code is completed.

```
>> linspace(2,34,9)  
  
ans =  
  
    2     6    10    14    18    22    26    30    34
```

Part 3 of 4:

Creating Cell Arrays

1. **Create a cell array.** A cell array is a rectangular set of data similar to a matrix but it can hold any type of data such as text, numbers, and/or vector. Code a cell array by listing a series of numbers, vectors, or characters in the same format as a matrix while characters are in quotation marks and vectors are in brackets. The cell array is surrounded by curly brackets { }. Hit enter once the code is completed.

1. Example: C={1, 'h', 5, 7; [1:2:11], 'm', 8, 25} or C={1 'h' 5 7; [1:2:11] 'm' 8 25}

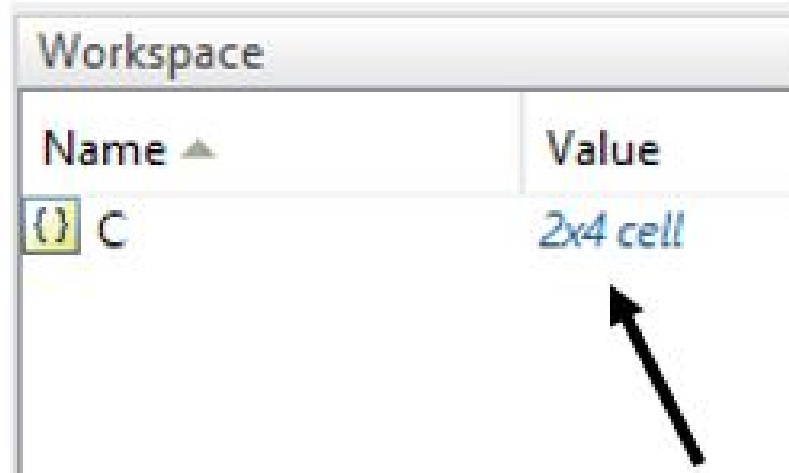
```
>> C={1, 'h', 5, 7; [1:2:11], 'm', 8, 25}  
  
C =  
  
    [         1]    'h'    [5]    [ 7]  
    [1x6 double]    'm'    [8]    [25]
```

```
>> C={1 'h' 5 7;[1:2:11] 'm' 8 25}
```

```
C =
```

```
      [      1]      'h'      [5]      [ 7]  
 [1x6 double]      'm'      [8]      [25]
```

2. **Double-click on the cell array's value in the workspace to view the cell array entirely.** A new window will open to so you can confirm your vector entry in the cell array.



The screenshot shows the MATLAB Variables - C window. The variable 'C' is displayed as a 2x4 cell array. The contents are as follows:

	1	2	3	4	5
1	1	'h'	5	7	
2	[1,3,5,7,9,11]	'm'	8	25	
3					
4					
5					
6					
7					

Part 4 of 4:

Avoiding Common Errors

1. **Ensure that all commands are in lowercase letters.**

2. Ensure that the Matrix has the same number of variables in each column.
3. Ensure that the Matrix has the same number of variables in each row.

```
>> M=[2 3 4 5;6 7 8]
```

1. Dimensions of matrices being concatenated are not consistent.

4. Ensure that the Cell Array has curly brackets { } enclosing it.
5. Ensure that the Matrix and/or Vector has brackets [] enclosing it

```
>> M=(2,3,4,5;6,7,8,9)
```

```
M=(2,3,4,5;6,7,8,9)
```

```
↑
```

1. Error: Expression or statement is incorrect--possibly unbalanced (, {, or [.

6. Ensure that the words or letters within a cell array are in quotation marks.

```
>> C={1 h 5 7;[1:2:11] 'm' 8 25}
```

```
Undefined function or variable 'h'.
```

You finished reading the article "**How to Create a Matrix, Vector, and Cell Array in MATLAB**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.