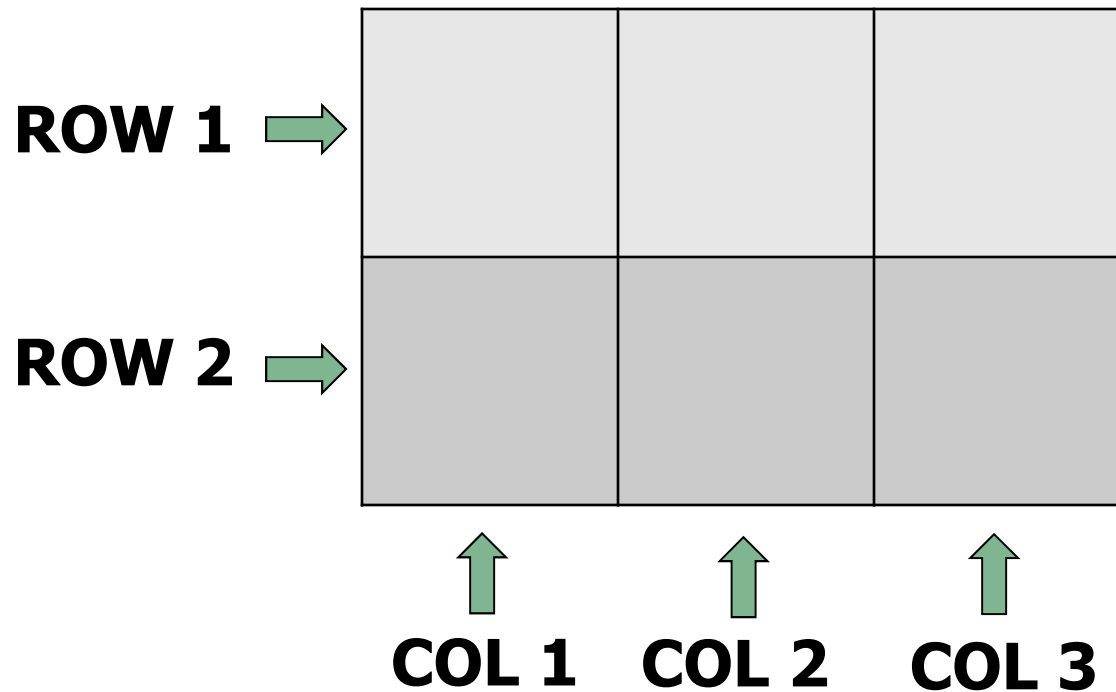


Arrays

Collection of data values organized into rows and columns.



Arrays

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

```
a = [1 2 3  
4 5 6  
7 8 9]  
a = [1 2 3; 4 5 6; 7 8 9]  
a = [1,2,3;4,5,6;7,8,9]
```

Individual data values within an array are accessed by including the name of the array followed by subscripts in parenthesis that identify the row and column of the particular value.

$a(2,3) = 6$ $a(3,2) = ?$ $a(1,3) = ?$

Arrays

Arrays can be classified as either *vectors* or *matrices*.

Vector: one dimensional array

Matrix: Array with two or more dimensions

$$CV = [3;5;9]$$

$$CV = \begin{bmatrix} 3 \\ 5 \\ 9 \end{bmatrix}$$

3x1 array – column vector

$$RV = [8 \ 7 \ 6]$$

$$RV = [8 \ 7 \ 6]$$

1x3 array – row vector

Arrays

Clear all your variables and command window, then generate the following arrays in Matlab:

$$A = \begin{bmatrix} 3 & 1 & 1 \\ 2 & 6 & 1 \\ 2 & 3 & 9 \end{bmatrix} \quad B = [3 \quad 2 \quad 1] \quad C = \begin{bmatrix} 1 \\ 5 \\ 3 \end{bmatrix}$$

$$D = \begin{bmatrix} B(1,3) & C(1,1) \\ A(2,1) & A(1,2) + B(1,2) \\ A(3) & A(7) \end{bmatrix} = \begin{bmatrix} ? & ? \\ ? & ? \\ ? & ? \end{bmatrix}$$

Arrays

$$A = \begin{bmatrix} 3 & 1 & 1 \\ 2 & 6 & 1 \\ 2 & 3 & 9 \end{bmatrix} \quad B = [3 \quad 2 \quad 1] \quad C = \begin{bmatrix} 1 \\ 5 \\ 3 \end{bmatrix}$$

E = A(:,1) = ? (All the values on column 1)

F = B' = ? (The transpose operator)

G = A(2:3,2:3) = ?

H = [A(:,1) C] = ?

I = [(A(:,1))' B] = ?

J = [(A(:,1))'; B] = ?

Initializing with shortcut expressions

Arr = *first: increment: last*

The list stops when the next value in the series is greater than the value of last

K = 1:2:14	= 1	3	5	7	9	11	13
L = 1:10	= ?						
M = 10:1	= ?						
N = 10:-1:1	= ?						
O = (0:0.25:1)*pi	= ?						
P = (3:3:21)'	= ?						

Initializing with built-in functions

size: return the size of the array (#files #columns)

size (A) = 3 3 size(P) = 7 1

[nr,nc]=size(P) → nr =7 nc =1

size (B) = ?

size (G) = ?

[nrh, nch] = size(H) =?

zeros: Create an array of all zeros

zeros(3)=

0 0 0

0 0 0

0 0 0

Initializing with built-in functions

```
zeros(2,3)=
```

```
0 0 0
```

```
0 0 0
```

```
zeros(size(A))=
```

```
0 0 0
```

```
0 0 0
```

```
0 0 0
```

```
Q = zeros(1,10)    =?
```

```
R = zeros(size(H)) =?
```

```
S = zeros(5,1)     =?
```


Initializing with built-in functions

ones: Create an array of all ones

```
ones(1,3)=  
1 1 1
```

eye: generates identity matrices

```
eye(2)=  
1 0  
0 1
```

```
U = ones(4,3)    =?
```

```
V = ones(size(H)) =?
```

```
W = eye(2)      =?
```

length: returns the length of a vector or the longest dimension of a 2 dimensional array

```
length(P) = 7
```

```
X = ones(length(P)) =?
```

```
Y = ones(length(P),1) =?
```

```
Z = eye(length(P)) =?
```